

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Локтионова Оксана Геннадьевна  
Должность: проректор по учебной работе  
Дата подписания: 10.10.2024 11:26:44  
Уникальный программный ключ:  
0b817ca911e6668abb13a5d426d79e5f1c11eabbf73e943df4a4851fda56d089

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра вычислительной техники



**Технологии обработки текстовой информации**

Методические указания  
по выполнению практической работы по дисциплине  
«Информатика» для студентов специальности  
10.05.02 Информационная безопасность  
телекоммуникационных систем

Курск 2021 г.

УДК 681.3

Составители: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина

Рецензент

Доцент кафедры программной инженерии,  
кандидат технических наук

*Ю.А. Халин*

**Технологии обработки текстовой информации:** методические указания по выполнению практической работы / Юго-Зап. гос. ун-т; сост.: С.В. Дегтярев, Е.Н. Иванова. – Курск, 2021. – 27 с.: ил. 11., табл. 6. – Библиограф.: с. 22.

Рассматриваются приемы использования различных инструментов, предоставляемых текстовым процессором, при преобразовании текстовой информации в соответствии с требуемыми параметрами. Теоретический материал сопровождается примерами.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.  
Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## **Цель работы**

Получить знания о методах и средствах текстового процессора, позволяющих обеспечить представление текстовой информации в заданном виде; научиться осуществлять выбор встроенного средства текстового процессора для получения документа, удовлетворяющего заданным параметрам; овладеть навыками использования различных инструментов, предоставляемых текстовым процессором, при преобразовании текстовой информации в соответствии с требуемыми параметрами.

## **Краткая теоретическая информация**

### **Некоторые приемы автоматизации создания и редактирования текстовых документов (на примере MS Word)**

#### **Использование автоматического Оглавления**

Текстовый документ удобно и приятно читать, если он структурирован, т.е. выделены смысловые части документа, каждая из которых имеет свой заголовок. В случае, если в текстовом документе заголовки оформлены с использованием стилей, Word позволяет автоматически генерировать оглавление. Оглавление содержит заголовки и номера страниц, на которых они расположены.

Для добавления оглавления необходимо:

- поместить курсор на то место где должно появиться оглавление;
- на ленте меню Ссылки в разделе Оглавление выбрать раскрывающееся меню Оглавление;
- в открывшемся списке выбрать вариант Оглавление.

Откроется диалоговое окно «Оглавление» (рисунок 1). В данном окне необходимо указать количество уровней, которые будут отображаться в оглавлении, выбрать формат оглавления. Вводимые изменения в формат оглавления автоматически отображаются в зоне «Образец печатного документа». Кнопка «Параметры» служит для указания стилей, которые будут добавляться в оглавление.

Используя команду «Изменить», можно изменить выравнивание и параметры шрифта для задания отображения названий заголовков документа в Оглавлении.

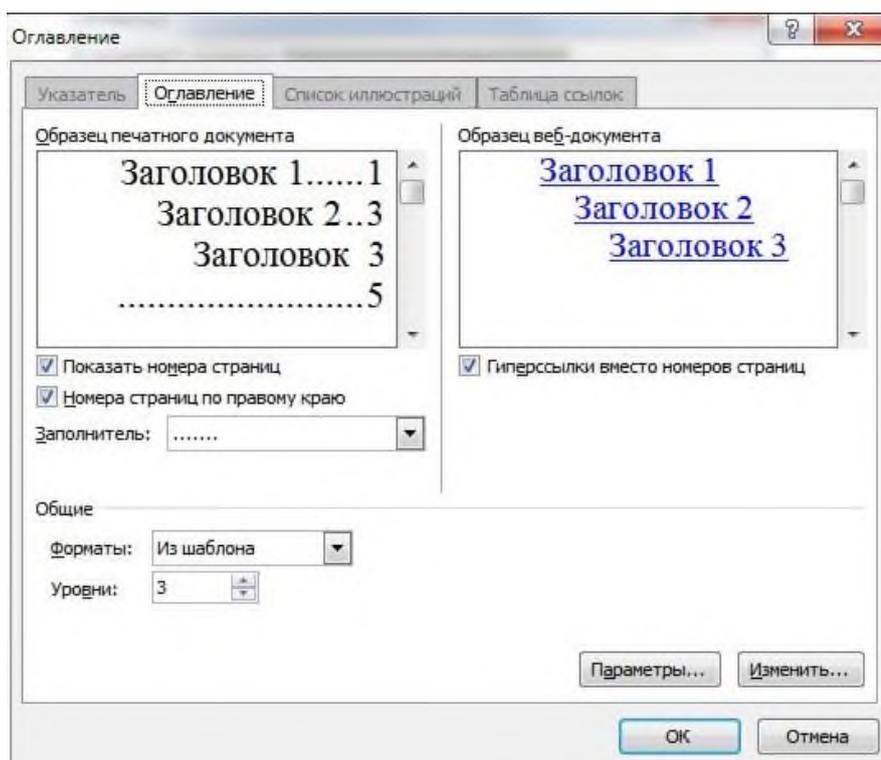


Рисунок 1 – Диалоговое окно «Оглавление»

После нажатия на кнопку «ОК» в место, куда указывал курсор, добавится оглавление.

Если в процессе работы над документом расположение заголовков поменялось или добавились новые, то можно обновить оглавление. Для этого необходимо нажать правой кнопкой на построенном оглавлении, выбрать в контекстном меню команду «Обновить поле» и в появившемся диалоге поставить флажок «Обновить целиком». После этого MS Word автоматически изменит Оглавление в соответствии с произошедшими изменениями.

Для добавления нового пункта в Оглавление требуется добавить новый заголовок в тексте. Для удаления пункта нужно – изменить стиль соответствующего заголовка в тексте на «Обычный».

Иногда при добавлении стороннего текста в оглавление попадают: целые абзацы, рисунки, пустые строки и пр. Для устранения этой проблемы необходимо изменить формат текста, который соответствует этим пунктам. Для этого необходимо воспользоваться командой «Очистить формат» (соответствующий инструмент можно найти на панели инструментов раздела Шрифт ленты меню Главная, рисунок 2) и назначить требуемый стиль проблемному фрагменту документа.

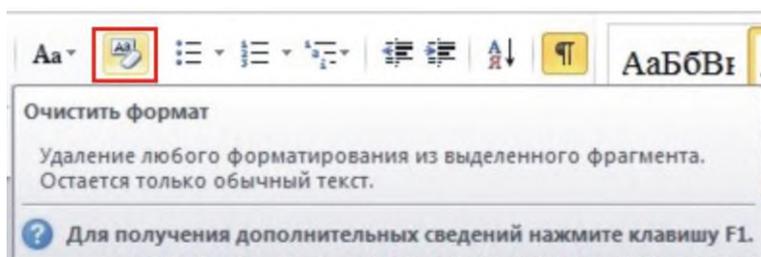


Рисунок 2 – Инструмент «Очистить формат»

Для быстрого перехода на соответствующий абзац с использованием оглавления необходимо щелкнуть по пункту оглавления с нажатой клавишей Ctrl.

### **Использование автоматической нумерации объектов**

В любом текстовом документе помимо текста и заголовков встречается множество различных объектов: списки, рисунки, таблицы, формулы. Для форматирования таких объектов также рекомендуется применять стили. В большинстве случаев для таких объектов подойдут встроенные стили, но иногда следует создавать и собственные.

Для создания списков удобно создать стиль, основанный на стиле «Обычный», настроенный для работы с многоуровневым списком. Так, например, можно создать стиль «Мой многоуровневый список», который содержит нужное форматирование списка до 4-го уровня. И если необходим одноуровневый список, можно применять этот стиль, используя только первый уровень, для многоуровневых списков использовать стиль в полной мере.

Для рисунков следует создать собственный стиль и применять его после каждой соответствующей вставки, для подписей под вставными объектами, можно использовать стиль «Название объекта» или создать свой. Преимущество встроенного стиля в том, что он автоматически применяется при использовании функции «Вставить название...».

Графические объекты могут добавляться из файлов, созданных другими приложениями или создаваться средствами MS Word.

Над такими объектами можно производить следующие операции:

- создание;
- вставка объекта в документ;
- форматирование положения объекта в тексте;
- редактирование объекта;

– нумерация объекта.

Создание объектов – это уникальное для каждого объекта действие. Оно производится с помощью соответствующего приложения. Например, рисунки можно создавать с помощью MS Paint, схемы – с помощью MS Visio, диаграммы – с помощью MS Excel, формулы и таблицы – с помощью MS Word.

У графических объектов, также как у текста, есть свои особенности форматирования, причем среди этих особенностей есть как общие для всех, так и уникальные для каждого типа. Общие принципы в основном относятся к добавлению объекта и его положению в тексте (они будут рассмотрены в начале этой главы). Уникальные особенности зависят от типа объекта и способа его создания.

Общие принципы работа с графическими объектами.

Вставка

Существует множество способов добавления графических объектов в документ. Например, при вставке графического объекта, хранящегося в файле, выполняют следующие действия.

- установить курсор в место вставки рисунка;
- выбрать инструмент Рисунок из раздела Иллюстрации ленты меню Вставка.

Форматирование графических объектов

Чаще всего в текстовых документах графические объекты – это прямоугольные области с изображением. Большая часть форматирования заключается в указании положения этого прямоугольника относительно основного текста.

Большинство параметров форматирования графических объектов задается с помощью диалога «Формат объекта». Его можно вызвать через контекстное меню выбрав «Формат объекта...». Некоторая часть настроек из этого диалога вынесена на ленту (меню Разметка страницы, раздел Упорядочить).

На Рисунок 3 представлен графический объект. Он имеет множество свойств (например, размер и положение в тексте), значение которых можно менять. Когда объект выделен, вокруг него появляются маркеры (рисунок 3). При наведении указателя мыши на один из маркеров указатель меняет форму и превращается в двунаправленную  $\leftrightarrow$ ,  $\updownarrow$ ,  $\nearrow$ ,  $\nwarrow$  или в круговую  $\curvearrowright$  стрелку. Потянув за один из этих маркеров можно изменить вид объекта:

- если тянуть за верхний или нижний маркеры, изменяется высота

объекта;

– если тянуть за левый или правый – ширина;

– если тянуть за угловые, пропорционально меняется ширина и высота (отношение сторон сохраняется);

– если тянуть за маркер управления углом поворота – объект начинает вращаться (рисунок 3, маркер управления углом поворота доступен не во всех объектах).

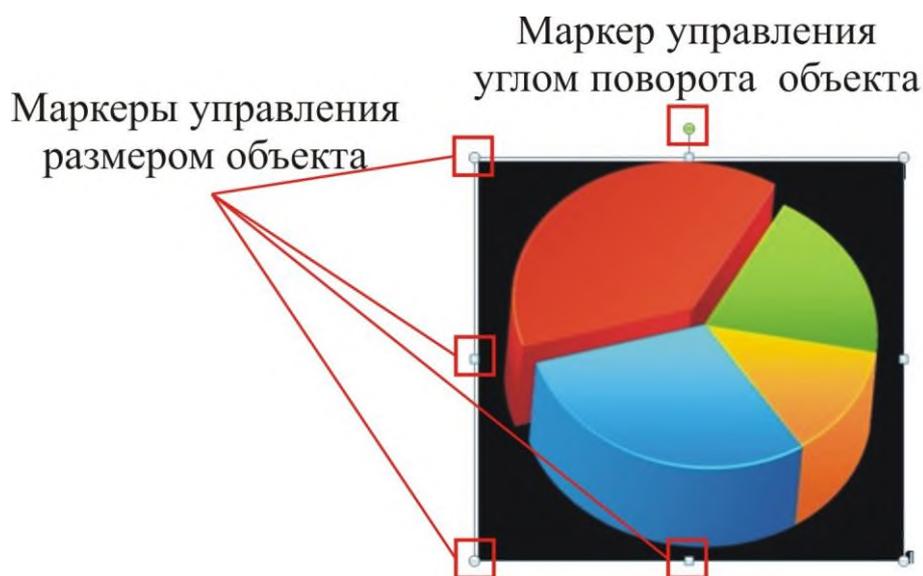


Рисунок 3 – Рисунок графического объекта

При наведении указателя мыши на объект указатель превращается в четырехнаправленную стрелку  $\leftrightarrow$ . В таком состоянии объект можно перетаскивать с помощью мыши по полю документа. Он займет новое положение в тот момент, когда левая кнопка мыши будет отпущена после перетаскивания.

Изменять размер и угол поворота объекта можно также через диалог «Формат объекта». Если объект – рисунок, в контекстном меню нужно выбрать «Размер...». С помощью этого диалога размер можно указывать не только в абсолютных значениях, но и в относительных (в процентах от исходного размера).

После вставки объекта следует задать характер его взаимодействия с текстом. Средства для этого представлены в диалоговом окне «Формат объекта» на вкладке «Положение». Более тонкая настройка взаимодействия объектов с текстом осуществляется через диалог «Дополнительная разметка», который можно вызвать, нажав на кнопку «Дополнительно ...».

При выборе положения рисунка в тексте следует уделить вни-

мание роли рисунка. Рисунок можно рассматривать как встроенный в текст, так и как элемент оформления страницы. Разница заключается в том, что происходит с объектом во время редактирования текста: он перемещается вместе с текстом при добавлении или удалении текста или он неподвижен относительно страницы, а текст перемещается, обтекая объект по заданным правилам. В первом случае объект надо закрепить относительно абзаца, а во втором случае – относительно страницы и соответствующим образом задать вертикальное и горизонтальное положение объекта.

Чаще всего в документах используют положение «В тексте», при этом объект становится элементом текста – одним большим «символом». Это положение удобно как для вставки объектов внутрь абзаца (например, небольшие по размеру рисунки), так и для отдельно стоящих объектов. В первом случае объект добавляется внутрь абзаца с текстом, во втором как отдельный абзац.

В положении «В тексте» рисунки перемещаются вместе с текстом. Графические объекты ведут себя более предсказуемо: не появляются лишних пустых пространств, более удобное расположение названия и пр. Форматирование рисунка сводится к форматированию абзаца, в котором он расположен: выравнивание (по левому краю, по правому краю и пр.), отступы и т.д. Это позволяет использовать стили абзацев для оформления рисунков и прочих графических объектов.

Для специфических задач можно использовать и остальные положения текста.

Вариант «Вокруг рамки». В этом случае текст располагается вокруг воображаемой прямоугольной рамки, охватывающей весь контур объекта.

Вариант «По контуру» отличается от предыдущего тем, что воображаемая прямоугольная рамка не проводится, и текст плавно обтекает контур объекта (если он криволинейный).

Вариант «Перед текстом» – это прием вставки объекта без обтекания. Текст и объект лежат на разных слоях, причем объект лежит выше и загораживает часть текста. Этим приемом пользуются, когда оформление важнее содержания.

Вариант «За текстом» – это еще один прием вставки объекта без обтекания. Текст и объект тоже лежат на разных слоях, но в данном случае объект лежит на нижнем слое и загораживается текстом. Этот вариант используют для размещения текста на темати-

ческом художественном фоне.

Вариант «Сквозное» – это прием обтекания, аналогичный обтеканию «По контуру», но в данном случае текст обтекает объект не только снаружи, но и изнутри.

Там же, в диалоговом окне «Дополнительная разметка» можно выбрать вариант обтекания «Сверху и снизу». Этот прием используют наиболее часто – его считают основным для объектов, ширина которых составляет более половины ширины страницы.

В некоторых положениях можно задать дополнительные параметры (диалог «Дополнительная разметка»):

Иногда возникают задачи, связанные с взаимодействием двух расположенных рядом объектов, созданных инструментами MS Word. Объекты могут находиться один над одним – перекрывать друг друга. Для настройки различных вариантов расположения объектов используются команды вкладки «Формат», которая появляется при выделении объекта созданного средствами Word.

Наиболее распространенные команды, которые используются для настройки относительного положения фигур, показаны на Рисунке 4:

- группирование;
- управление порядком следования;
- выравнивание;
- распределение.

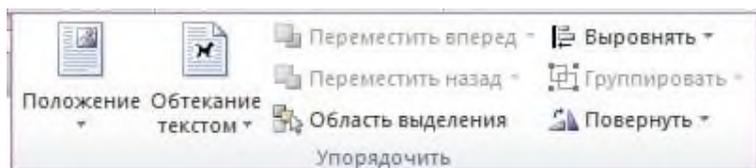


Рисунок 4 – Команды управления положением изображений на экране

Большинство команд доступно и в контекстном меню графического объекта. Для выделения нескольких объектов необходимо нажать кнопку SHIFT и, удерживая её, выбрать несколько фигур.

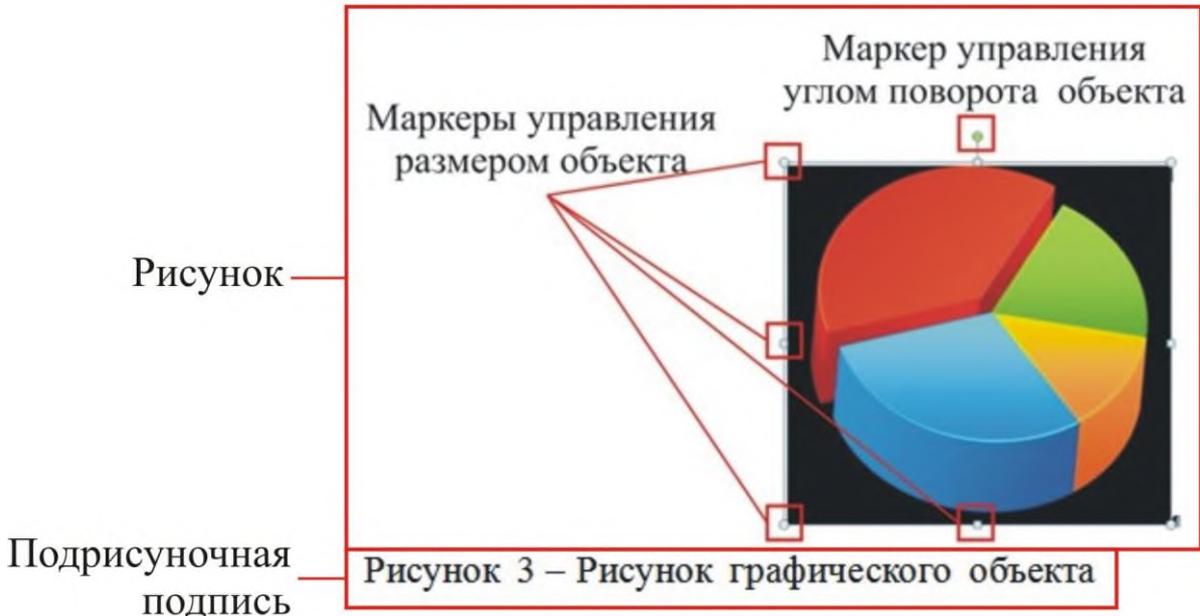
### **Добавление названий, нумерации и ссылок в тексте**

В случаях, когда в текстовом документе использовано множество рисунков им следует давать номера, названия, а так же обеспечивать ссылку в тексте на соответствующий рисунок (рисунок 5).

MS Word предоставляет средства для автоматизации этих действий.

### Ссылка на рисунок в тексте

ли тянуть за маркер управления углом поворота  
 нает вращаться (рисунок 3, маркер управления угл  
 упен не во всех объектах).



При наведении указателя мыши на объект  
 Рисунок 5 – Оформление ссылки на рисунок

Подрисуночная подпись обычно выполняется под рисунком и состоит из нескольких частей (рисунок б):

«Рисунок» – постоянная часть названия (подпись соответствует типу объекта: у рисунка – «Рисунок», у таблицы – «Таблица» и т.д.);

«3» – номер рисунка в тексте;

«Рисунок графического объекта» – название рисунка.

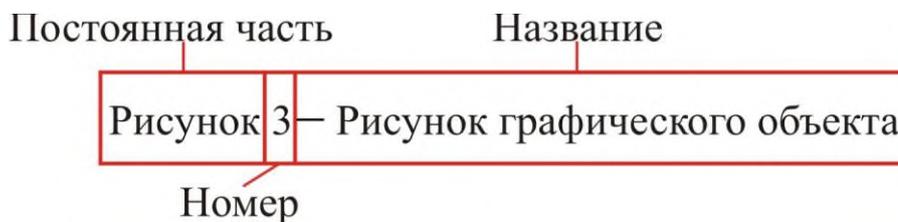


Рисунок 6 – Принцип формирования названия рисунка

В тексте ссылка на рисунок может выглядеть, например, так: «На Рисунок 3 представлен графический объект», или так: «...объект начинает вращаться (рисунок 3, маркер ...». Таким обра-

зом ссылка на рисунок состоит из двух частей: подписи и номера рисунка (

рисунок 7). К сожалению, нет средств, позволяющих изменять постоянную часть названия автоматически, это вы можете сделать сами.

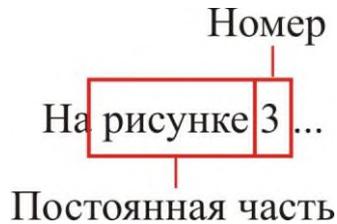


Рисунок 7 – Оформление ссылки на рисунок

Для оформления названий рисунков и ссылок на них средствами MS Word необходимо создать и изменить 2 стиля (рисунок 8):  
 – создать новый стиль «Объект» (основан на стили «Обычный», поля абзаца 0 см, отступ первой строки – 1,25, выравнивание – по ширине, интервалы и отступы – 0, междустрочный интервал – 1,5 строки);

– изменить встроенный стиль «Название объекта» (поля абзаца 0 см, отступ первой строки – 1,25, выравнивание – по ширине, шрифт 14 пт, обычное начертание, интервалы и отступы – 0, междустрочный интервал – 1,5 строки)

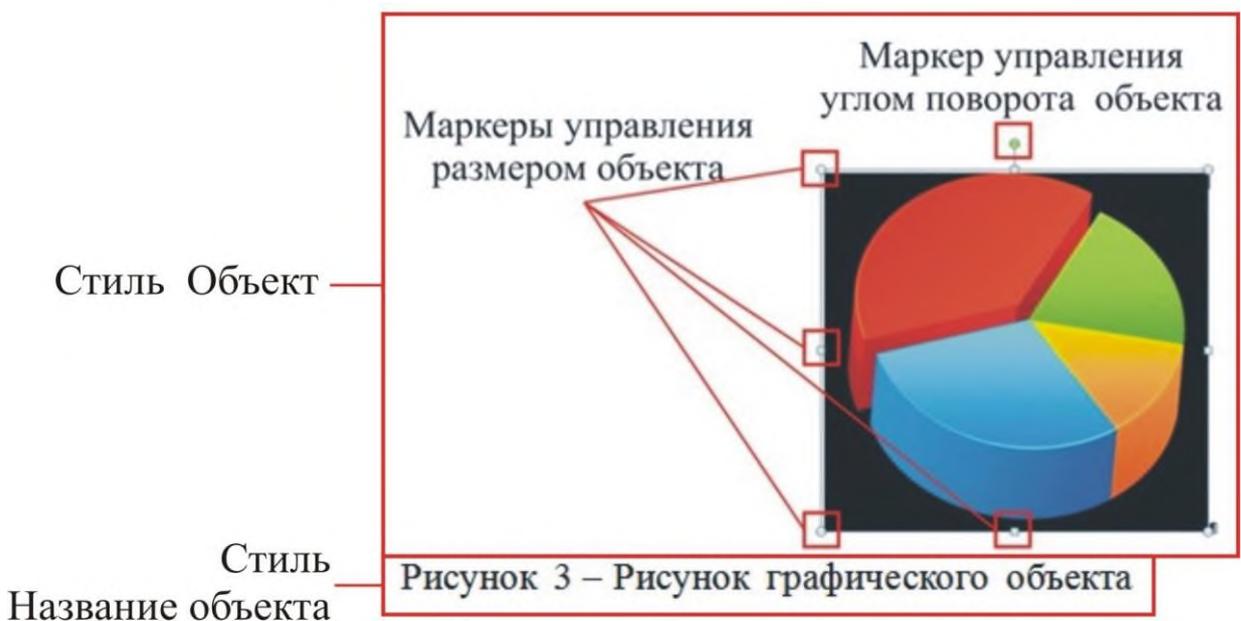


Рисунок 8 – Стили для рисунков и их названий

После создания стилей для добавления рисунка в текст необ-

ходимо выполнить следующие действия:

- в документ в новый абзац добавить рисунок (положение «В тексте»);
- к рисунку (к абзацу рисунка) применить стиль «Объект»;
- создать название рисунка:
- в контекстном меню рисунка выбрать «Вставить название»;
- в поле «подпись» выбрать «Рисунок» (рисунок 9). Если такого пункта нет, создать подпись, нажав кнопку «Создать ...» и набрав там «Рисунок»;
- указать положение «Под выделенным объектом»;
- нажать кнопку «ОК»;
- после этого под рисунком появится название «Рисунок <номер>», к которому уже применен стиль «Название объекта»;
- введите после номера пробел, тире, пробел и с большой буквы напишите название рисунка.

Для добавления к номеру рисунка номера главы в диалоговом окне «Название» следует нажать кнопку «Нумерация» и установить соответствующую опцию.

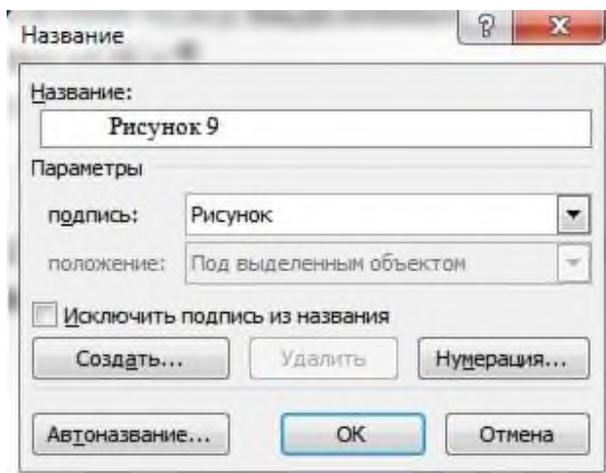


Рисунок 9 – Диалоговое окно «Название»

Для вставки в текст ссылки на рисунок необходимо вызвать команду «Перекрестная ссылка», размещенную в разделе Названия ленты меню Ссылки. При этом в тексте документа уже должен быть размещен рисунок и создана подрисуночная подпись. В появившемся диалоговом окне (см. рисунок 10) выбрать тип ссылки «Рисунок», в выпадающем списке «Вставить ссылку на:» выбрать «Постоянная часть и номер», в области «Для какого названия» выбрать строку с нужным названием рисунка.

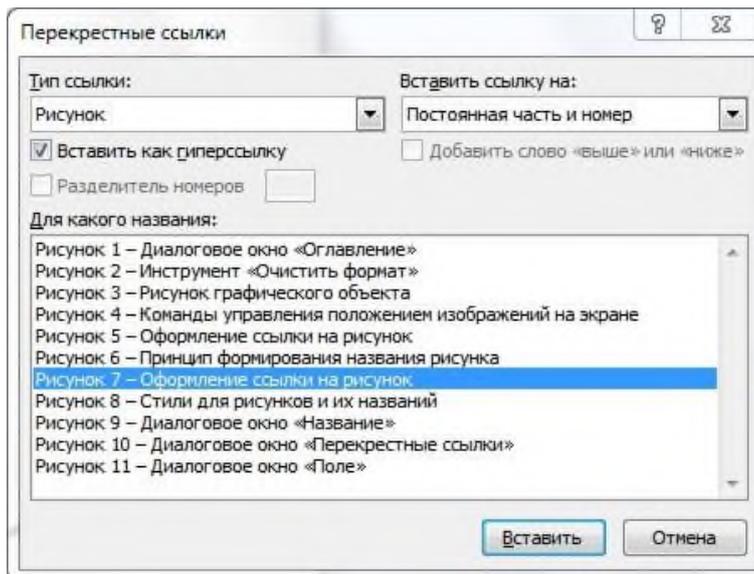


Рисунок 10 – Диалоговое окно «Перекрестные ссылки»

Ссылка на рисунок в тексте начинается с прописной (заглавной) буквы, редактирование регистра обычным способом не даст желаемого результата, так как при обновлении ссылки первая буква снова станет прописной. Для исключения этой ситуации после вставки в текст ссылки следует щелкнуть по ней правой кнопкой, выбрать в контекстном меню пункт «Изменить поле» и в появившемся диалоговом окне «Поле» (рисунок 11) в выпадающем списке «Формат» выбрать строку «все строчные», после чего активировать опцию «Сохранять формат при обновлении».

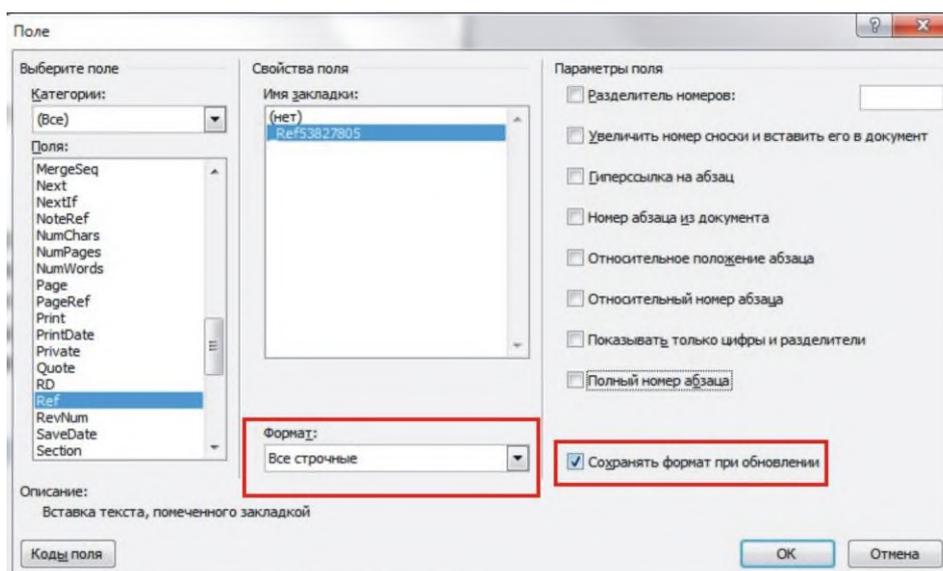


Рисунок 11 – Диалоговое окно «Поле»

Иногда возникает необходимость в ссылке скрыть постоянную

часть (т.е. слово «рисунок»), а номер оставить. Для выполнения такого действия следует выделить ту часть ссылки, которую необходимо убрать и нажать сочетание клавиш Ctrl+Shift+N или вызвать диалоговое окно Шрифт и активировать опцию Скрытый. Здесь следует помнить, что при включенном режиме отображения непечатаемых символов выделенный фрагмент не исчезнет с экрана, при отключении этого режима фрагмент текста будет скрыт.

### **Задание**

Составить отчет по результатам выполнения лабораторной работы. Отчет должен содержать:

- титульный лист (Приложение А);
- Содержание (Приложение Б);
- Цель работы;
- содержательная часть;
- Выводы по работе.

Содержательная часть отчета состоит из краткого ответа на вопрос, текста сказки и текста научной статьи. Выбор текста и параметров его форматирования осуществляется согласно варианту (табл. 1 – табл. 6). Между Целью работы и Первым разделом разрыва страниц (разделов) делать не нужно.

Стили Заголовков сказки и Заголовков статьи необходимо создать самостоятельно согласно заданным параметрам (табл. 1, табл. 4). Новый стиль должен основываться на стиле Заголовков 1. Параметры стиля Заголовков 1:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 12 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Номера страниц проставляются внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первой страницей является титульный лист, нумерация проставляется, начиная с Цели работы.

Макет Содержания приведен в Приложении Б.

Для оформления текста Цели работы, ответа на вопрос, Выво-

дов по работе необходимо использовать стиль «Обычный». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

В качестве ответа на теоретический вопрос нужно в форме многоуровневого списка написать, какие положительные и отрицательные стороны использования встроенных инструментов текстового процессора (о тех инструментах, которыми Вы пользуетесь) Вы видите (Приложение В).

Оформления объектов в тексте. Все объекты в тексте (рисунки, формулы) должны быть пронумерованы. На каждый объект должна быть ссылка в тексте. Для нумерации и ссылок должны использоваться средства автоматической нумерации, предоставляемые MS Word. Формулы нумеруются арабскими цифрами, начиная с «1». Для обозначения рисунков следует использовать слово «Иллюстрация».

Рисунки должны иметь расположение «в тексте» и выровнены по контексту, без отступа первой строки. Необходимо использовать различные виды выравнивания для рисунков без повторения. У каждого рисунка снизу должна быть подрисуночная подпись вида «Иллюстрация <номер> – <Название>» того же формата, что и текст.

Формулы должны быть выровнены по центру. Справа должен быть указан номер. Формулы набираются только с использованием MS Equation 3.0 или MathType equation.

Проверка на автоматическую нумерацию: при добавлении заголовка, рисунка или формулы и обновления полей (выделить весь текст и нажать F9) остальные объекты и ссылки на них должны корректно перенумеровываться. При обновлении полей не должен изменяться формат текста.

Параметры стиля Заголовков для названия сказки

вариант	шрифт	начертание	размер	цвет	видоизменение	выравнивание	интервал перед/после	междустрочный
1	BatangChe	К	16	красный	с тенью	Ц	12/12	1,5 строки
2	Book Antiqua	П	17	желтый	контур	ЛК	12/16	одинарный
3	Bookman Old Style	ПК	18	оранжевый	с тенью + контур	Ш	16/16	двойной
4	Cambria	К	19	синий	приподнятый	Ц	16/12	точно 12 пт
5	Century Gothic	П	20	зеленый	утопленный	ЛК	12/24	точно 14 пт
6	Comic Sans MS	ПК	19	бордовый	с тенью	Ш	16/24	точно 16 пт
7	Constantia	К	18	коричневый	контур	Ц	24/12	1,5 строки
8	Dejavu Serif	П	17	салатовый	с тенью + контур	ЛК	24/16	одинарный
9	Gabriola	ПК	16	красный	приподнятый	Ш	24/24	двойной
10	Palatino Linotype	К	17	фиолетовый	утопленный	Ц	12/12	точно 12 пт
11	Segoe Print	П	18	желтый	с тенью	ЛК	12/16	точно 14 пт
12	Trebuchet MS	ПК	19	оранжевый	контур	Ш	16/16	точно 16 пт
13	MS UI Gothic	К	20	синий	с тенью + контур	Ц	16/12	1,5 строки
14	Segoe Script	П	19	фиолетовый	приподнятый	ЛК	12/24	одинарный
15	MS Gothic	ПК	18	зеленый	утопленный	Ш	16/24	двойной

Отступы слева: 0; справа: 0; первая строка: нет для всех вариантов.

О – обычное; К – курсив; П – полужирный; ПК – полужирный курсив.

Ц – по центру; ЛК – по левому краю; Ш – по ширине.

Параметры для текста сказки

вариант	шрифт	начертание	размер	междустрочный интервал
1	Book Antiqua	О	14	1,5 строки
2	Bookman Old Style	К	16	двойной
3	Century	О	14	точно 16 пт
4	Book Antiqua	К	16	точно 14 пт
5	Bookman Old Style	О	14	1,5 строки
6	Century	К	16	двойной
7	Book Antiqua	О	14	точно 16 пт
8	Bookman Old Style	К	16	точно 14 пт
9	Century	О	14	1,5 строки
10	Book Antiqua	К	16	двойной
11	Bookman Old Style	О	14	точно 16 пт
12	Century	К	16	точно 14 пт
13	Book Antiqua	О	14	1,5 строки
14	Bookman Old Style	К	16	двойной
15	Century	О	14	точно 16 пт
Цвет: черный; видоизменений нет; выравнивание: по ширине; отступы слева: 0; справа: 0; первой строки: 1,25 см; интервал перед: 0; после: 0 для всех вариантов				

Параметры страницы для текста сказки

вариант	переплет	ориентация	Поля			
			верхнее	нижнее	левое	правое
1	1 см	альбомная	2 см	2,5 см	1,5 см	1,75 см
2	1,25 см	альбомная	2,2 см	2,4 см	1,6 см	1,6 см
3	0,75 см	альбомная	2,3 см	2,3 см	1,7 см	1,5 см
4	0,8 см	альбомная	2,4 см	2,2 см	1,8 см	1,4 см
5	0,9 см	альбомная	2,5 см	2,1 см	1,9 см	1,8 см
6	1,1 см	альбомная	2,1 см	2,0 см	2,0 см	1,9 см
7	1,15 см	альбомная	1,9 см	1,9 см	2,1 см	2,0 см
8	1,2 см	альбомная	1,8 см	1,8 см	2,2 см	2,1 см
			слева	справа	верхнее	нижнее
9	1 см	книжная	2 см	2,5 см	1,5 см	1,75 см
10	1,25 см	книжная	2,2 см	2,4 см	1,6 см	1,6 см
11	0,75 см	книжная	2,3 см	2,3 см	1,7 см	1,5 см
12	0,8 см	книжная	2,4 см	2,2 см	1,8 см	1,4 см
13	0,9 см	книжная	2,5 см	2,1 см	1,9 см	1,8 см
14	1,1 см	книжная	2,1 см	2,0 см	2,0 см	1,9 см
15	1,15 см	книжная	1,9 см	1,9 см	2,1 см	2,0 см
Размер бумаги: А5						

Параметры стиля Заголовков для названия статьи

вариант	шрифт	начертание	размер	цвет	отступ первая строка	выравнивание	интервал перед/после	междустрочный
1	Arial	О	16	черный	нет	Ц	12/12	1,5 строки
2	Arial Black	П	17	черный	1,25	ЛК	12/16	одинарный
3	Arimo	О	18	черный	1,25	Ш	16/16	двойной
4	Calibri	П	19	черный	нет	Ц	16/12	точно 12 пт
5	Candara	О	20	черный	1,25	ЛК	12/24	точно 14 пт
6	Centure	П	19	черный	1,25	Ш	16/24	точно 16 пт
7	Consolas	О	18	черный	нет	Ц	24/12	1,5 строки
8	Corbel	П	17	черный	1,25	ЛК	24/16	одинарный
9	Garamond	О	16	черный	1,25	Ш	24/24	двойной
10	Georgia	П	17	черный	нет	Ц	12/12	точно 12 пт
11	Impact	О	18	черный	1,25	ЛК	12/16	точно 14 пт
12	Meiryo	П	19	черный	1,25	Ш	16/16	точно 16 пт
13	Segoe UI	О	20	черный	нет	Ц	16/12	1,5 строки
14	Tahoma	П	19	черный	1,25	ЛК	12/24	одинарный
15	Verdana	О	18	черный	1,25	Ш	16/24	двойной

Отступы слева: 0; справа: 0; для всех вариантов.  
О – обычное; К – курсив; П – полужирный; ПК – полужирный курсив.  
Ц – по центру; ЛК – по левому краю; Ш – по ширине.

### Параметры для текста статьи

вариант	шрифт	начертание	размер	междустрочный интервал	ширина колонок
1	Times New Roman	О	14	1,5 строки	7,75
2	Arial	О	16	двойной	7,85
3	Cambria	О	14	точно 16 пт	7,9
4	Times New Roman	О	16	точно 14 пт	8,0
5	Arial	О	14	1,5 строки	8,1
6	Cambria	О	16	двойной	8,2
7	Times New Roman	О	14	точно 16 пт	8,3
8	Arial	О	16	точно 14 пт	8,4
9	Cambria	О	14	1,5 строки	8,25
10	Times New Roman	О	16	двойной	8,15
11	Arial	О	14	точно 16 пт	8,05
12	Cambria	О	16	точно 14 пт	7,8
13	Times New Roman	О	14	1,5 строки	7,9
14	Arial	О	16	двойной	8,0
15	Cambria	О	14	точно 16 пт	8,15

Цвет: черный; видоизменений нет; выравнивание: по ширине; отступы слева: 0; справа: 0; первой строки: 1,25 см; интервал перед: 0; после: 0; две колонки равной ширины для всех вариантов

Параметры страницы для текста статьи

вариант	ориентация	Поля			
		верхнее	нижнее	левое	правое
1	книжная	1 см	2,5 см	1,5 см	1,75 см
2	книжная	1,2 см	2,4 см	1,6 см	1,6 см
3	книжная	1,3 см	2,3 см	1,7 см	1,5 см
4	книжная	1,4 см	2,2 см	1,8 см	1,4 см
5	книжная	1,5 см	2,1 см	1,9 см	1,8 см
6	книжная	1,1 см	2,0 см	1,0 см	1,9 см
7	книжная	1,9 см	1,9 см	1,1 см	1,0 см
8	книжная	1,8 см	1,8 см	1,2 см	1,1 см
9	книжная	1 см	2,5 см	1,5 см	1,75 см
10	книжная	1,2 см	2,4 см	1,6 см	1,6 см
11	книжная	1,3 см	2,3 см	1,7 см	1,5 см
12	книжная	1,4 см	2,2 см	1,8 см	1,4 см
13	книжная	1,5 см	2,1 см	1,9 см	1,8 см
14	книжная	1,1 см	2,0 см	1,0 см	1,9 см
15	книжная	1,9 см	1,9 см	1,1 см	1,0 см

На страницах с текстом сказки в нижнем колонтитуле должна быть информация <Название сказки, автор>. На страницах с текстом статьи в верхнем колонтитуле должна быть информация <Фамилия Имя Отчество студента, группа>.

В разделе Выводы по работе необходимо написать, чему Вы научились, что узнали нового, какой инструмент текстового процессора, по Вашему мнению, является полезным, а какой – не следует использовать.

### **Список использованных источников**

1. Колокольникова, А.И. Информатика [Текст] : учебное пособие / А.И. Колокольникова. – Москва ; Берлин : Директ-Медиа, 2020. – 289 с. : ил., табл. – Режим доступа: URL: <https://biblioclub.ru/>

2. Информатика: лабораторные работы и тесты [Текст]: / П.В. Балакшин, В.В. Соснин, И.В. Калинин [и др.]. Санкт-Петербург : Университет ИТМО, 2019. – 59 с. : ил., схем. – Режим доступа: URL: <https://biblioclub.ru/>

## Оформление титульного листа отчета по лабораторной работе

Минобрнауки России  
ЮЗГУ  
ФФиПИ  
кафедра наименование кафедры

ЛАБОРАТОРНАЯ РАБОТА №1  
Технологии обработки текстовой информации

Выполнил(и):  
Проверил:

Курск 2020 г.

ст. гр. ХХ-016  
Фамилия И.О.  
Фамилия И.О.  
Иванова Е.Н.

Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 16 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

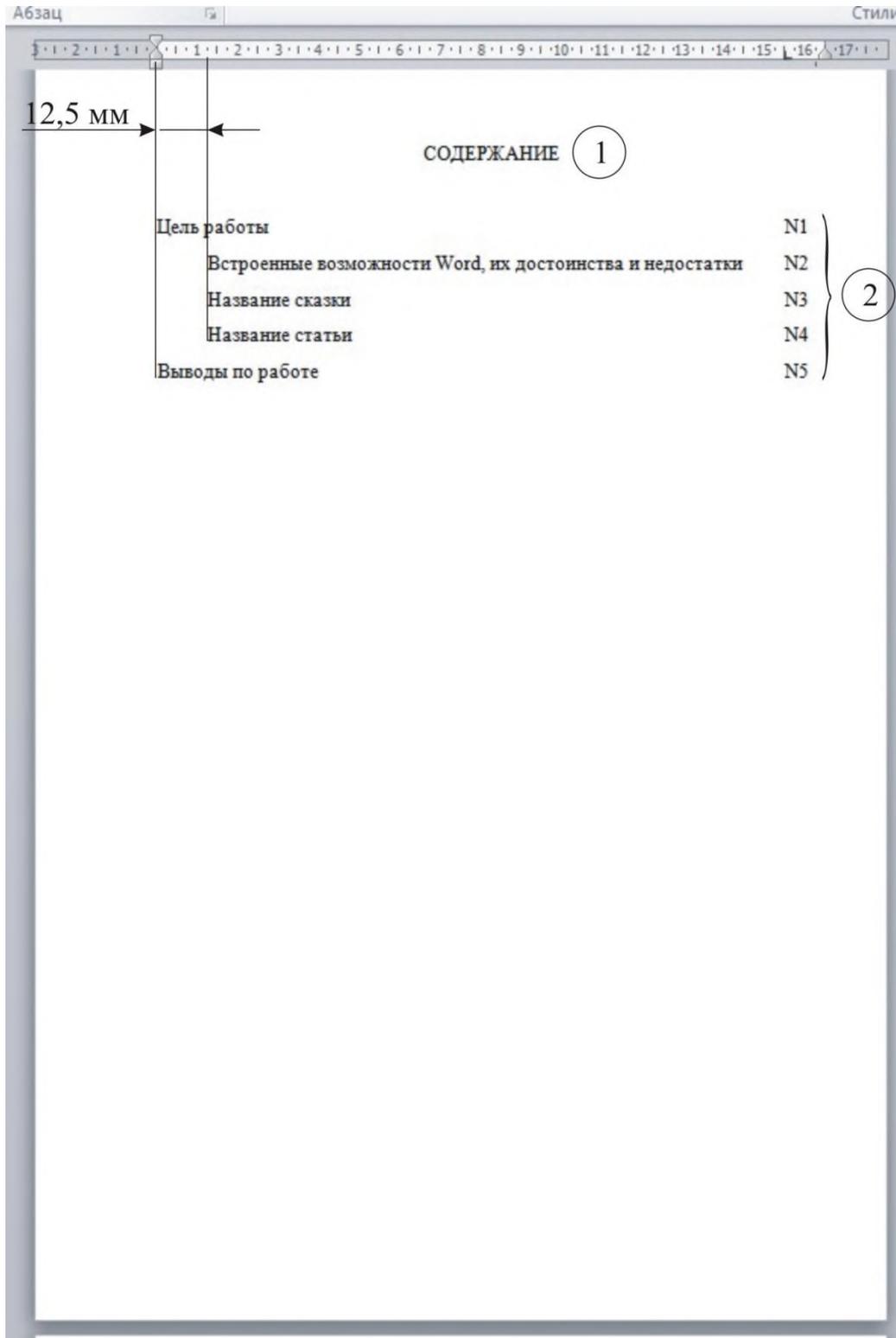
3: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

4: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

5: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по ширине, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки, позиция табуляции 12 см.

6: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

## Оформление Содержания отчета по лабораторной работе



Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по левому краю, отступы слева 0, справа 0, интервалы перед 0, после 0, междустрочный 1,5 строки.

## Пример ответа на теоретический вопрос и его оформления

Текстовый процессор обладает рядом полезных инструментов, помогающих ...

...

1. Полезные инструменты

1) Автособираемое оглавление. Этот инструмент позволяет быстро находить...

2) ...

...

5) ...

2. Беспольные инструменты

1) ...

2) ...

...

12,5 мм

18 мм

18 мм

25 мм

The image shows a screenshot of a text editor window with a ruler at the top. The text is as follows:

Текстовый процессор обладает рядом полезных инструментов, помогающих ...

...

1. Полезные инструменты

1) Автособираемое оглавление. Этот инструмент позволяет быстро находить...

2) ...

...

5) ...

2. Беспольные инструменты

1) ...

2) ...

...

Vertical red lines are drawn at approximately 12.5 mm, 18 mm, and 25 mm from the left margin. Horizontal arrows indicate these distances: 12,5 мм from the left edge to the first red line; 18 мм from the left edge to the second red line; 18 мм from the second red line to the third red line; and 25 мм from the second red line to the third red line.

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)  
Кафедра вычислительной техники

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Локтионова  
« 16 » 05 2023 г.



**Технологии обработки текстовой информации LaTeX**

Методические указания по выполнению практических и лабораторных работ  
по дисциплине «Информатика» для направлений подготовки 09.03.00,  
10.03.00, 11.03.00, 12.03.00.

УДК 004.91  
Составитель Е.А. Коломиец

Рецензент  
Кандидат технических наук, доцент Л.В. Стародубцева

**Технологии обработки текстовой информации LaTeX:** методические указания по выполнению практических и лабораторных работ по дисциплине «Информатика» / Юго-Зап. гос. ун-т; сост. Е.А. Коломиец. Курск, 2023. 19 с. Библиогр.: с. 17.

Рассматриваются методы и средства использования языка разметки и системы подготовки документов, использующая в качестве механизма для верстки – LaTeX.

Методические рекомендации предназначены для студентов, обучающихся по направлениям подготовки 09.03.00, 10.03.00, 11.03.00, 12.03.00.

Текст печатается в авторской редакции.

Подписано в печать . Формат 60x84 1/16.  
Усл. печ. л. 0,93 п.л. Уч.-изд. л. 0,8. Тираж 100 экз. Заказ. 415 Бесплатно.  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## Цель работы

Получить знания о методах и средствах использования языка разметки и системы подготовки документов, использующая в качестве механизма для верстки LaTeX, позволяющих обеспечить представление текстовой информации в заданном виде. Научится автоматизировать многие аспекты верстки документов, такие как автоматическая нумерация страниц, таблиц, иллюстраций, формул, перекрестные ссылки, колонтитулы, предметный указатель, оглавление и список литературы.

## Краткая теоретическая информация

### Структура LАТЕХ-файла

Структура LaTeX-файла состоит из преамбулы документа, основного текста, включенных и выключенных формул. Рассмотрим основные специальные знаки:

- % – начало комментария;
- { } – начало и конец блока;
- \ – специальная команда LaTeX;
- \$ – выделение формул;
- ~ – неразрывный пробел;
- &, #, \_, ^ – прочие специальные символы.

Рассмотрим создание простого документа на языке разметки LaTeX в редакторе TeXworks (рис.1).

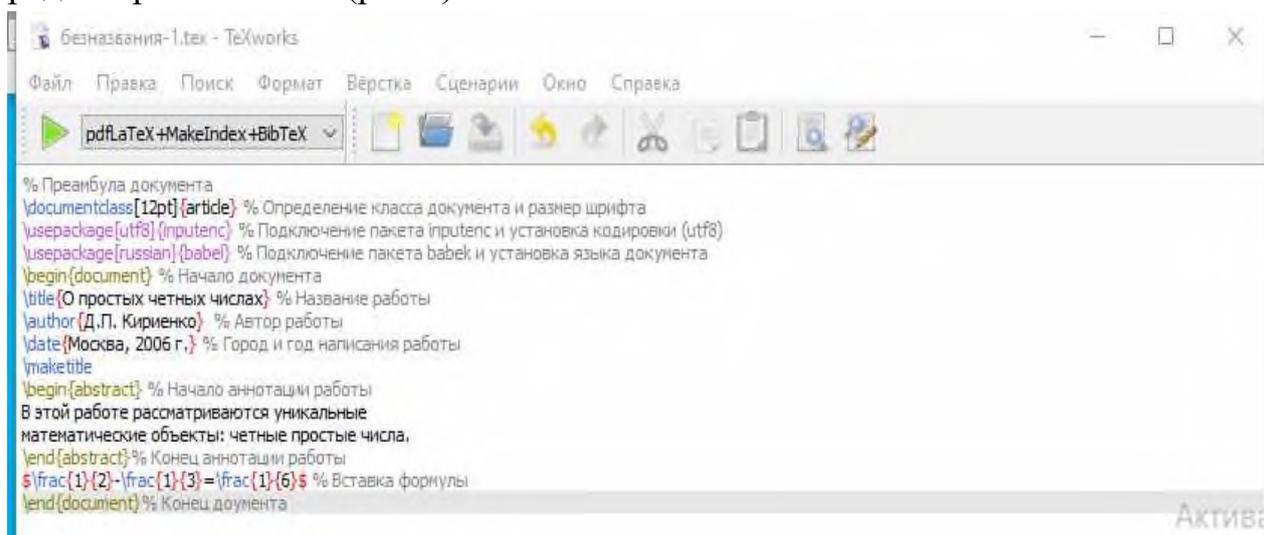


Рисунок 1 – Создание простого документа

После запуска программы, подгружаются все используемые пакеты и отображается готовый вид документа в формате PDF (рис.2).

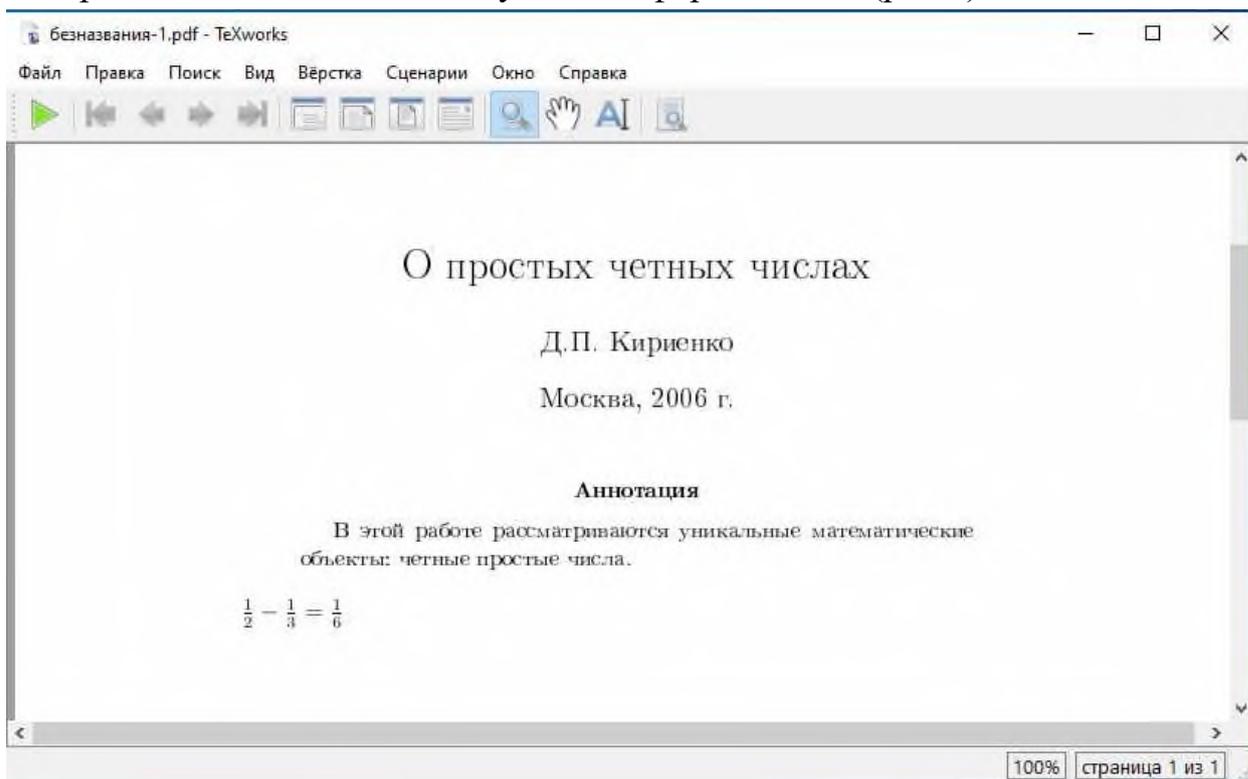


Рисунок 2 – Отображение созданного документа

Разбивка текста на абзацы происходит добавлением пустой строки после блока текста. Внутри блока текста слова можно разделять одним или несколькими пробелами / разрывом строки. Для задания короткого тире нужно написать два знака минус подряд «--», для задания длинного тире – «---».

В тексте могут использоваться специальные команды, начинающиеся с символа «\» (`\documentclass`). Обязательные аргументы команд заключаются в фигурные скобки, а не обязательные в квадратные скобки и они ставятся перед обязательными (`\documentclass[12pt]{article}`).

### Классы в LaTeX

С начала существования LaTeX использовалось 6 классов документов в зависимости от их применения:

- `article` – статья;
- `book` – книга;
- `report` – отчет;
- `proc` – доклад;
- `letter` – письмо;
- `slides` – слайды (презентация).

Каждый класс содержит определенный набор специальных команд и окружений. Так класс `article` используется для написания научных статей и содержит команды секционирования и часто используемое окружение `abstract` (аннотация или предисловие). В класс `book` уже добавлена команда `chapter` (глава) и определен набор команд для оформления титульного листа, предисловия и оглавления. Класс `proc`, используемый в основном для написания тезисов докладов, отличается от класса `article` в основном только обязательной двухколоночной печатью. Для написания писем используется класс `letter`, он существенно отличается от выше перечисленных классов, в нем без надобности использовать, например, команды секционирования.

В `LaTeX` предусмотрен класс `minimal`, который не содержит специальных команд и служит основой для создания других классов.

В настоящее время на основе шаблонного класса создано много других классов, используемых для обработки текстовой информации. Среди них можно выделить:

1) `Koma-Script` – набор классов, позволяющих использовать размеры шрифта `9pt`, `14pt`, `17pt` и со значительно меньшими по размеру полями относительно стандартных классов. К ним относятся: `scrartcl` (статья), `scrbook` (книга), `scrletter` (письмо), `scrreprt` (отчет);

2) `Eskdx` – набор классов и стилей, предназначенный для обработки текстовой информации в соответствии с требованиями ЕСКД. В основу входят три класса: `eskdtext` (текстовая документация), `eskdtab` (чертеж, схема), `eskdgraph` (документы, разбитые на графы).

Рассмотрим пример использования класса и его аргументы. Например: `\documentclass[12pt, a4paper, twocolumn]{article}`. Наиболее часто используемые параметры:

- `10pt` – размер текста 10 пунктов;
- `a4paper` – используется бумага формата А4;
- `landscape` – текст располагается вдоль широкого края бумаги;
- `twoside` – разные поля четных и нечетных страниц;
- `twocolumn` – текст разбивается на две колонки;
- `titlepage` – название, автор и дата документа отображаются на отдельной странице;
- `flng` – выключенные формулы располагаются слева, а не по центру строки.

## Обработка формул

При наборе формул включенных и выключенных игнорируются все пробелы. LaTeX расставит пробелы автоматически в зависимости от различных математических символов. Включенные формулы автоматически переносятся на следующую строку и прописываются « $\$x\$$ ». Выключенные формулы всегда располагаются на одной строке и прописываются « $\$\$x\$\$$ ». Рассмотрим написание некоторых переменных:

- $\$abc\$$  – произведение переменных;
- $\$a^2\$$  – возведение переменной в степень 2;
- $\$a^{145}\$$  – возведение переменной в степень, имеющие несколько символов;
- $\$y_1^2+\dots+y_n^2\$$  – возведение переменной степень, имеющей нижний индекс.

Дроби модно набирать как с помощью косой черты ( $\$3/y\$$ ), так и с помощью команды « $\backslashfrac$ » (рис.1,2).

Математические знаки такие как  $+$ ,  $-$ ,  $=$ ,  $<$ ,  $>$ ,  $:$  на языке верстки с помощью следующих команд (рис.3).

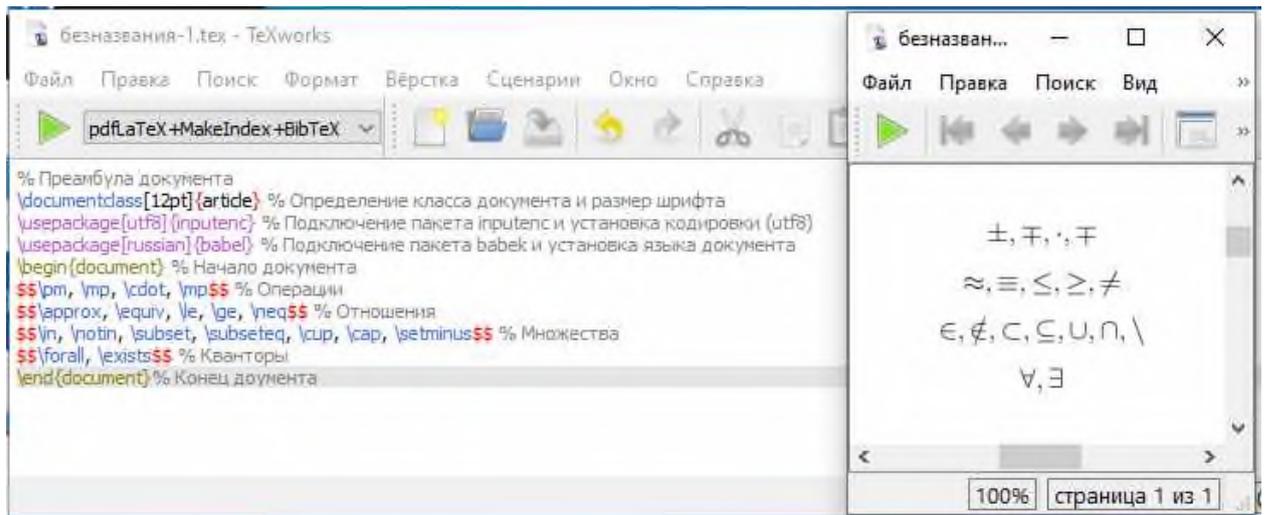


Рисунок 3 – Запись и отображение математических знаков

Использование символов греческого алфавита и назначение специальных команд представлены в таблице 1.

Таблица 1  
Греческий алфавит

Альфа	$\alpha$	<code>\alpha</code>	Бета	$\beta$	<code>\beta</code>	Гамма	$\gamma$	<code>\gamma</code>	Дельта	$\delta$	<code>\delta</code>
Эпсилон	$\epsilon$	<code>\epsilon</code>	Дзета	$\zeta$	<code>\zeta</code>	Эта	$\eta$	<code>\eta</code>	Тэта	$\theta$	<code>\theta</code>
Иота	$\iota$	<code>\iota</code>	Каппа	$\kappa$	<code>\kappa</code>	Ламбда	$\lambda$	<code>\lambda</code>	Мю	$\mu$	<code>\mu</code>
Ню	$\nu$	<code>\nu</code>	Кси	$\xi$	<code>\xi</code>	Омикрон	$\omicron$	<code>\omicron</code>	Пи	$\pi$	<code>\pi</code>
Ро	$\rho$	<code>\rho</code>	Сигма	$\sigma$	<code>\sigma</code>	Тау	$\tau$	<code>\tau</code>	Ипсилон	$\upsilon$	<code>\upsilon</code>
Фи	$\phi$	<code>\phi</code>	Хи	$\chi$	<code>\chi</code>	Пси	$\psi$	<code>\psi</code>	Омега	$\omega$	<code>\omega</code>

Кроме этих команд существуют альтернативные начертания некоторых букв, часто используемых в русских текстах, например, `\varepsilon`, `\varphi` и т.д.

Для обозначения заглавных греческих букв необходимо начинать написание специальной команды с заглавной буквы, например, `\Delta`, `\Omega` и т.д. Однако не следует использовать заглавные греческие буквы  $\Sigma$  и  $\Pi$  для обозначения сумм и произведений, так как есть специальные команды.

Тригонометрические и другие формулы записываются в виде следующих команд, аналогично их написанию, например, `\sin`, `\cos`, `\arcsin` и т.д. Примеры некоторых функций приведены на рисунке 4.

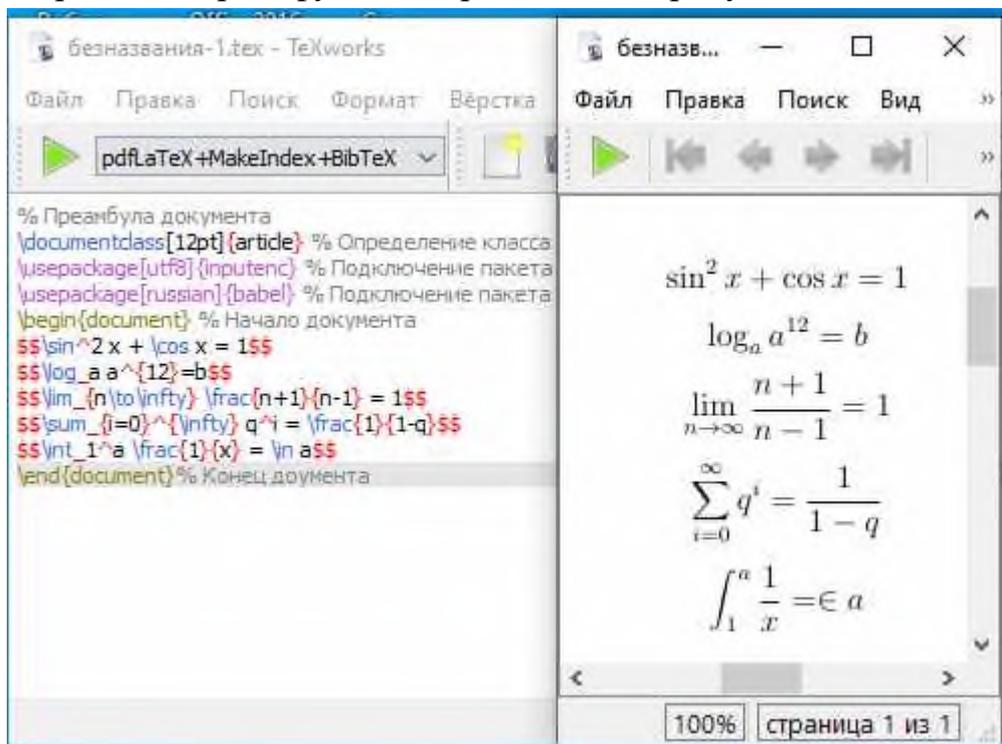


Рисунок 4 – Тригонометрические функции

Чтобы представить модуль числа  $x$  необходимо прописать следующую команду «`\$|x|\$`». Для набора больших скобок необходимо прописать команду

« $\left(1+\frac{1}{n}\right)^n = \left(1+\frac{1}{n}\right)^n$ ». Получим следующее отображение формулы:

$$\left(1 + \frac{1}{n}\right)^n = \left(1 + \frac{1}{n}\right)^n$$

Для отображения квадратных корней используется команда « $\sqrt{4}$ », если необходимо указать степень корня, то она указывается как необязательный параметр « $\sqrt[3]{4}$ ». Штрихи прописываются как « $x'$ », а факториал – « $3!$ ».

Для написания многострочных формул или систем уравнений необходимо подключить пакет `amsmath`, указав его преамбуле документа.

Одним из способов набора формул в несколько строк является специальная команда `{multiline}`. Многострочная формула заключается в блок `\begin{multiline}` формула `\end{multiline}` и не использует обозначения  $\$$ . Для обозначения многоточия используется специальная команда `\ldots`, а разрыв строки обозначается « $\backslash$ ». Пример записи и отображения верстки представлен на рисунке 5. Как видно из рисунка формуле сразу присвоен порядковый номер 1, если вам не нужна нумерация формулы, то необходимо использовать специальную команду `multiline*`.

$$1 + 2 + 3 + 4 + \dots + 50 + 34 + 56 \dots + 45 + 34 = 229 \quad (1)$$

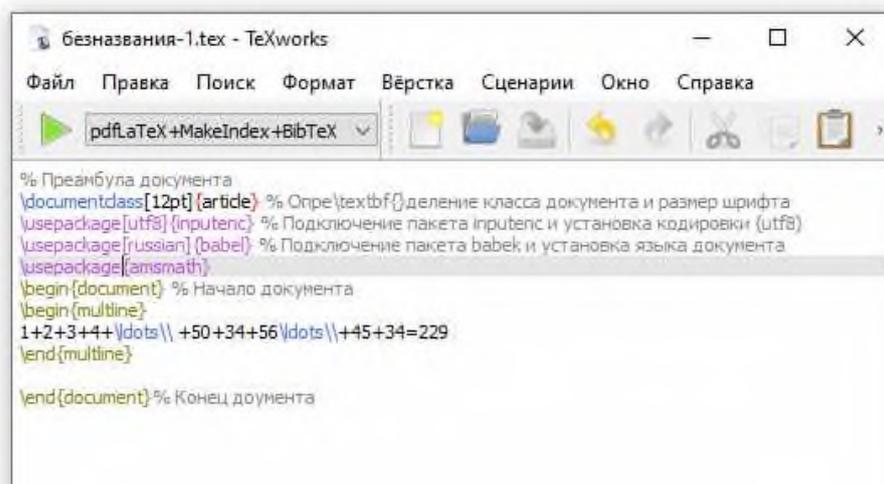


Рисунок 5 – Многострочная формула

Если необходимо внести в текст несколько формул подряд используется команда `gather`.

Верстка системы уравнений осуществляется с помощью команды `aligned`. Данное окружение оформляется внутри `$$`. Пример использования окружения представлен на рисунке 6. Формула начинается со специальной команды `\left\{`, которая создает большую скобку и заканчивается командой `\right\}`. Как видно из рисунка 6 в формуле присутствует знак `&`, обозначающий разделение и выравнивание частей уравнений от знака «`=`» вправо и влево.

Одна выключенная формула может состоять из нескольких окружений `aligned` для записи в одной строке нескольких систем уравнений, которые разделяются командой `\Leftrightarrow` - знак равносильности.

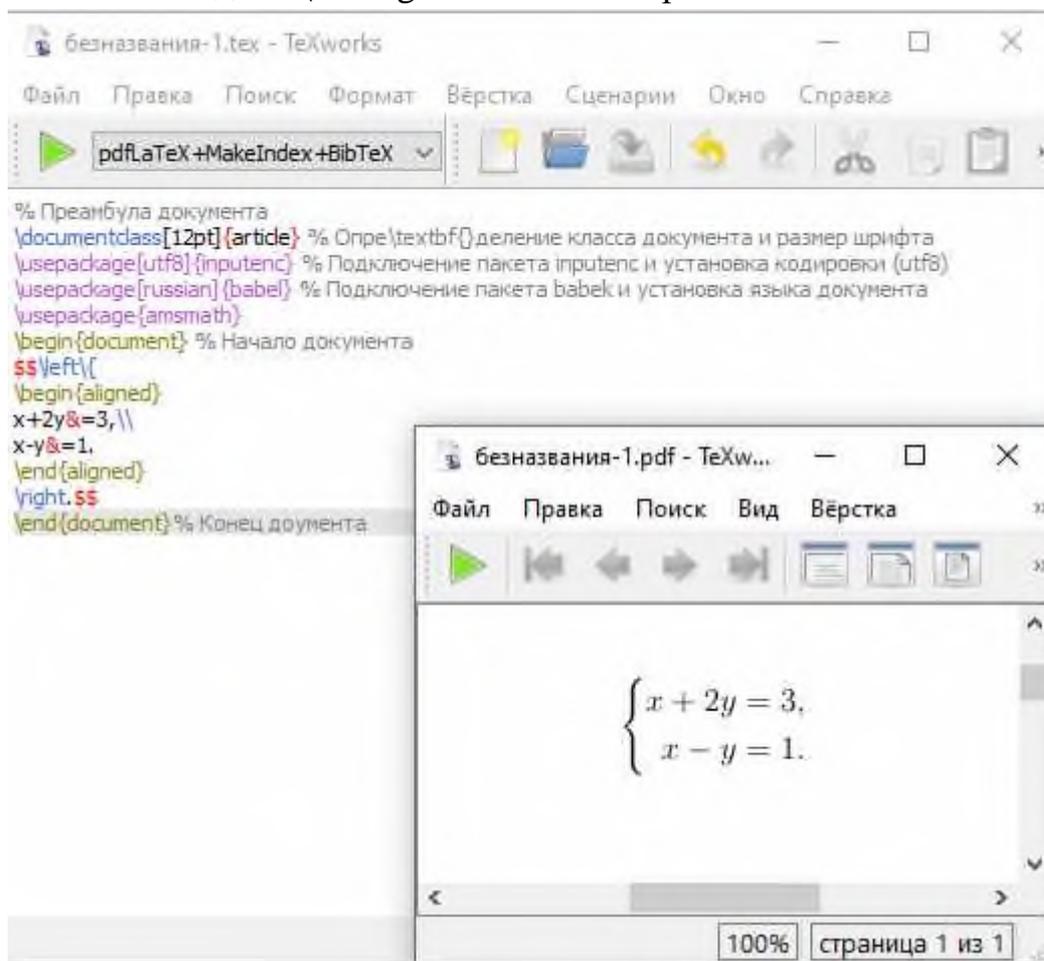


Рисунок 6 – Система уравнений

В верстке LaTeX предусмотрена автоматическая нумерация формул. В любом научном математическом тексте возникает необходимость нумерации формул с право от формулы в конце странице для ссылки на сами формулы в тексте (1), (2) и т.д. Если необходимо добавить какую-нибудь формулы в середину текста, то нумерация формул сбивается и приходится заново ее проставлять. LaTeX автоматически исправляет данную проблему.

Для ссылки на формулу в тексте она должна быть «помечена», для этого предусмотрено окружение `equation`. Также, как и в окружениях, рассмотренных ранее начало формулы обозначается командой `\begin{equation}`, а конец формулы командой `\end{equation}`. Данное окружение не нужно заключать внутрь знаков `$$`.

Для того, чтобы сделать ссылку в тексте на формулу, необходимо в окружение `equation` добавить команду `\label {название метки}`. В самом тексте ссылка на формулу оформляется в виде специальной команды `\ref{название метки}`. Если необходимо указать на какой странице находится формула, то добавляется префикс `page`. Команда будет выглядеть следующим образом «`\pageref{название метки}`».

### Обработка шрифтов, абзацы и окружения

Для переключения размеров шрифта используются специальные команды, представленные на рисунке 7. Если Вы хотите, чтобы для отдельных частей текста использовался свой размер шрифта, то необходимо данную часть текста заключить в фигурные скобки и прописать специальную команду «`{\large Текст}`».

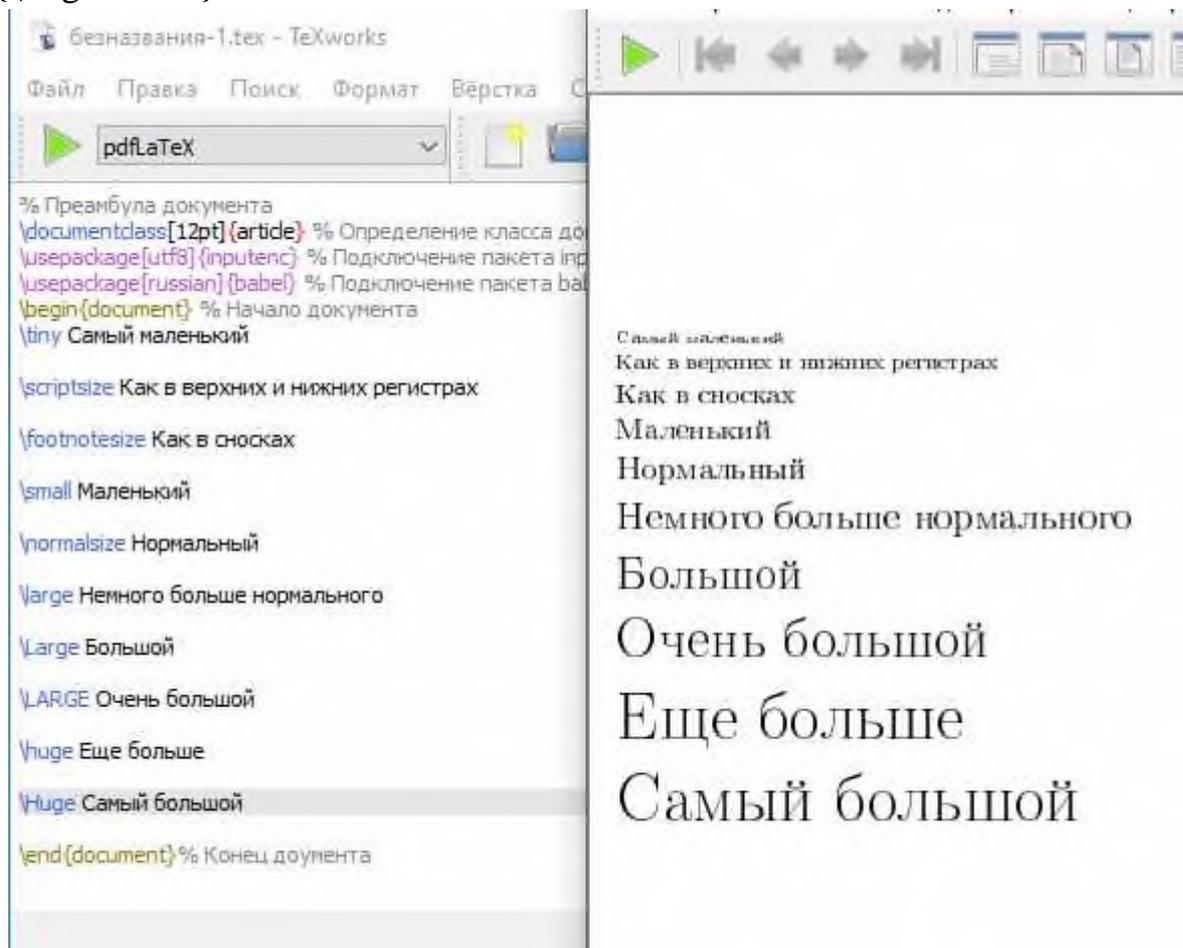


Рисунок 7 – Размер шрифта

Также каждый доступный шрифт имеет свое несколько атрибутов таких как:

- Гарнитура шрифта (family): `rmfamily` (шифт с засечками), `sffamily` (шифт без засечек), `ttfamily` (моноширинный шрифт).
- Насыщенность (series): `mdseries` (обычная), `bfseries` (**полужирная**).
- Начертание (shape): `upshape` (прямое), `itshape` (курсив), `slshape` (наклонное), `scshape` (капитель).

Различные атрибуты шрифта можно комбинировать и записывать их последовательно в фигурных скобках перед основным текстом. Если есть необходимость отмены всех начертаний в документе, то прописывается специальная команда «`\normalfont`» или «`\textnormal`».

Для того, чтобы убрать отступ абзаца (красной строки) необходимо использовать специальную команду `\noindent`. Данная команда очень удобна при оформлении описания формул, где необходимо указать расшифровку переменных. Для изменения величины отступа абзаца используется команда `\parindent=1cm` в преамбуле документа. Величина может указываться в различных единицах измерения.

Для установки интервала между абзацами используется команда `\parskip=1mm` в преамбуле документа. Если необходимо вставить дополнительный интервал между конкретными абзацами используется одна из стандартных команд `\smallskip`, `\medskip`, `\largeskip`, создающая соответственно маленький, средний и большой интервал. С помощью команды `\vspace{1mm}` устанавливается конкретный интервал между абзацами. Разрыв страницы обозначается командой `\newpage`.

Рассмотрим часто используемые окружения для оформления текста. Для выравнивания текста используются команды `center` (выравнивание по центру), `flushleft` (выравнивание по левому краю) и `flushright` (выравнивание по правому краю). Для оформления цитаты используется команда `quote`.

Если вы хотите, чтобы оформление текста отображалось в явном виде, то используйте команду `verbatim`, при этом текст выводится моноширинным шрифтом (текст программ). Для оформления нумерованных списков – `enumerate`, для ненумерованных списков – `itemize`. Для задания элемента списков используется команда `\item`. Использование окружения начинается с команды `\begin{имя_окружения}` и завершается командой `\end{имя_окружения}` (рис.8).

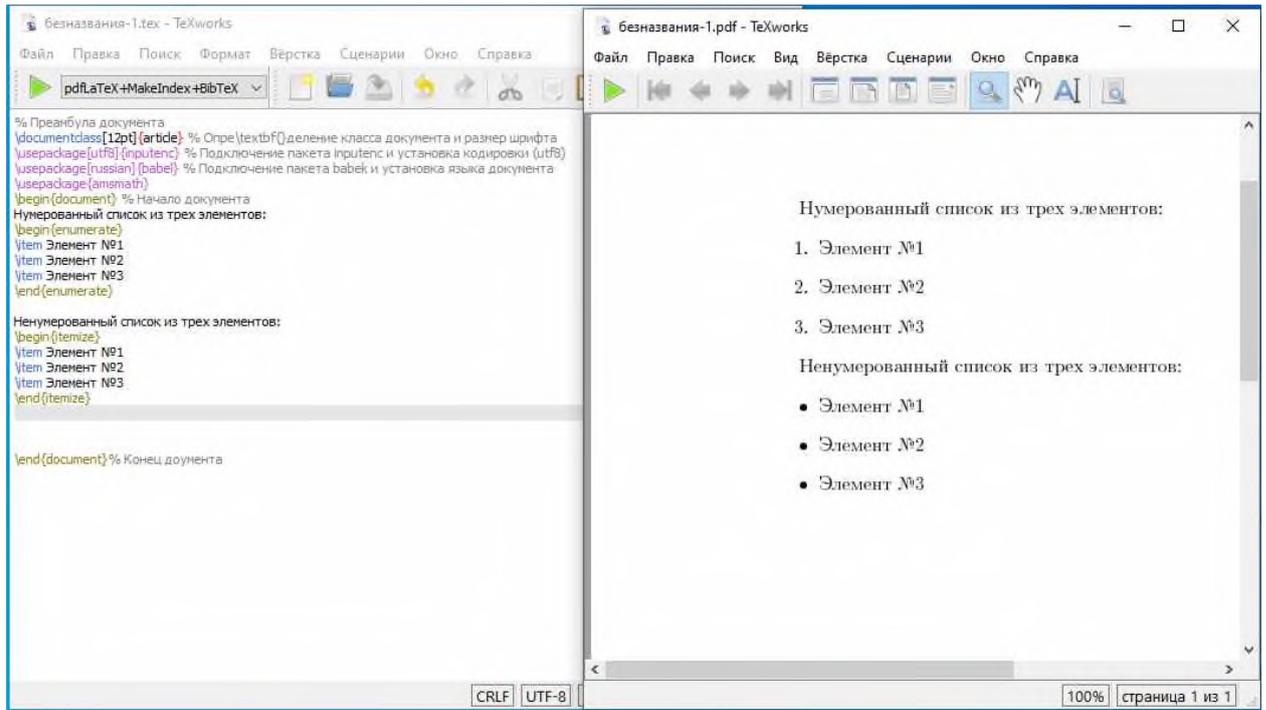


Рисунок 8 – Список элементов

## Оформление содержания и его элементов, нумерация страниц

Обычно весь документ разбивается на разделы, подразделы, подподразделы и т.д., которые нумеруются автоматически. Для это существуют специальные команды:

- `\section{Название раздела}`
- `\subsection{Название подраздела}`
- `\subsubsection{Название под подраздела}` и т.д.

Существует более мелкое разделение тексты на параграфы и т.д., которые не нумеруются: `\paragraph`, `\subparagraph`, `\subsubparagraph`.

Для создания оглавления используется специальная команда `\tableofcontents`. В оглавление включаются все разделы, имеющие нумерацию. Для корректного отображения оглавления необходимо скомпилировать документ несколько раз. Все данные о разделах хранятся в файле с расширением «.aux».

В LaTeX нумерация страниц ставится автоматически. Часто при оформлении титульных листов первые страницы не должны быть пронумерованы. Для ручной нумерации существует специальная команда `\pagestyle`, имеющая обязательные аргументы:

- `{empty}` – номера страниц отсутствуют;
- `{plain}` – номера страниц внизу;
- `{headings}` – номера страниц и колонтитулы с названием разделов;
- `{myheadings}` – собственный стиль оформления колонтитулов.

При использовании этой команды стиль колонтитулов указывается в преамбуле документа и распространяется на весь текст. Если необходимо изменить колонтитул на определенной странице используется специальная команда `\thispagestyle` с обязательным аргументом.

Сноски оформляются с помощью команды `\footnote{текст_сноски}`.

## Оформление рисунков и таблиц

Для добавления рисунков или изображений в текст документа необходимо в преамбуле документа прописать специальную команду `\usepackage{graphics}`.

В текст документа необходимо добавить специальную команду `\includegraphics{Путь к файлу/имя рисунка.eps}` `\{\small Рисунок 1 – Название рисунка}`. Часто с этой командой используем необязательный аргумент `[scale=0.6]` – уменьшение размера рисунка на 60% от оригинального. Рисунок в основном располагается по центру, поэтому одним из вариантов будет заключить его в окружении `center` вместе с подписью. Для ссылки на рисунок в тексте нужно добавить метку `\label` (рис.9). Файл с изображением должен находиться в папке с проектом.

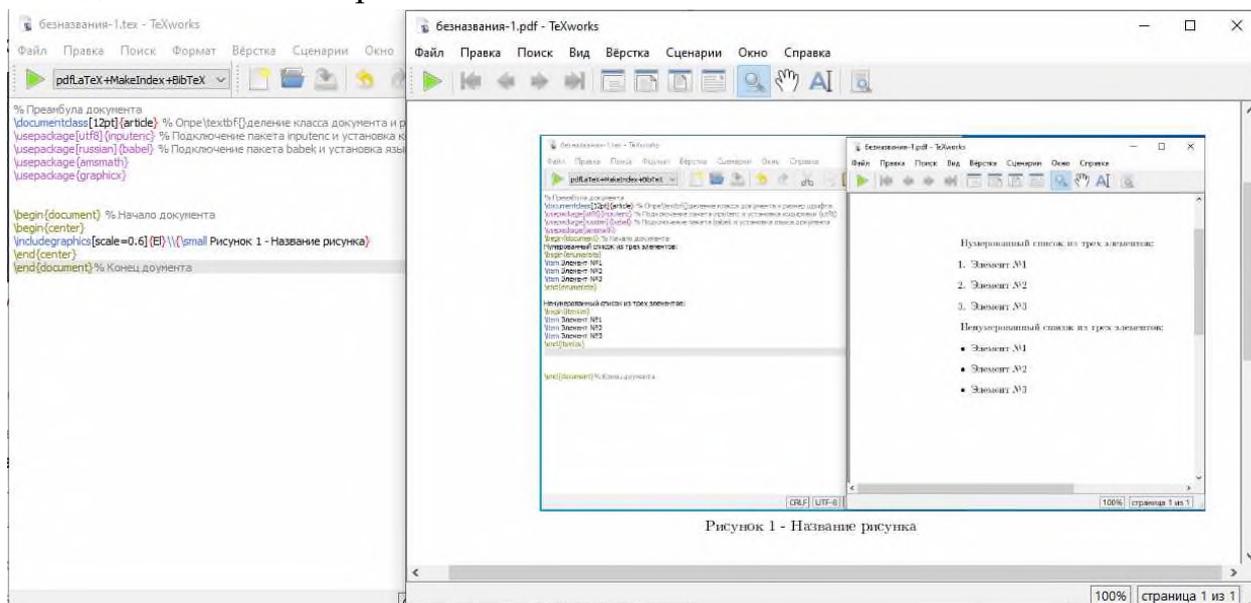


Рисунок 9 – Вставка рисунка в текст

Оформление таблиц осуществляется за счет окружения `tabbing`. В нем задается шаблон строки фиксированной ширины, затем создается таблица с фиксированной шириной колонок в которой строки между собой не связаны и части таблицы могут располагаться на разных страницах.

Для создания таблиц также применяются окружения `array` и `tabular`. Сразу сформулируем на основе окружения `tabular` правила верстки таблиц (рис.10):

1. Колонки с левым (l), правым (r) и центральным (c) выравниваем. Используемые идентификаторы выравнивания располагаются в аргументе окружения `tabular`.

2. Начало и конец таблицы соответственно прописываются командами `\begin{tabular}{|l| p{3,5cm} |}` и `\end{tabular}`. В данном примере в таблицу добавлены параметры – `{|l| p{3,5cm} |}` – таблица состоит из трех колонок, разделенных на всю высоту таблицу линиями `|`, причем в первой колонки текст выровнен по левому краю, размер второй колонки составляет 3,5 см и текст выровнен по ширине.

3. Горизонтальные линии в таблице прописываются командой `\hline`, разрыв между строками – `\\`.

4. Для объединения колонок используется `\multicolumn`.

The screenshot shows the TeXworks interface with two windows. The left window displays the LaTeX source code for a document. The right window shows the rendered PDF output, which includes a table of planetary data.

**LaTeX Source Code:**

```

% Преамбула документа
\documentclass[12pt]{article} % Опре\textbf{}деление класса докумен
\usepackage[utf8]{inputenc} % Подключение пакета inputenc и установ
\usepackage[russian]{babel} % Подключение пакета babel и установ
\usepackage{amsmath}
\usepackage{graphicx}

\begin{document} % Начало документа
\begin{tabular}{|l| p{3,5cm} |} \hline
Планета & Среднее удаление от Солнца,  $5 \cdot 10^6$  км \\ \hline
Меркурий & 58 \\ \hline
Венера & 108 \\ \hline
Земля & 150 \\ \hline
Марс & 227 \\ \hline
Юпитер & 778 \\ \hline
Сатурн & 1,424 \\ \hline
Уран & 2,874 \\ \hline
Нептун & 4,497 \\ \hline
Плутон & 5,894 \\ \hline
\end{tabular}
\end{document} % Конец документа

```

**Rendered Table:**

Планета	Среднее удаление от Солнца, $10^6$ км
Меркурий	58
Венера	108
Земля	150
Марс	227
Юпитер	778
Сатурн	1 424
Уран	2 874
Нептун	4 497
Плутон	5 894

Рисунок 10 – Верстка таблицы

Информация, набираемая в окружении `tabular`, всегда отображается на одной странице. Для многостраничных таблиц используется окружения `supertabular` или `longtable`.

## Оформление списка литературы

В LaTeX список литературы нумеруется автоматически с помощью окружения `thebibliography`. Окружение имеет обязательный аргумент – номер элемента библиографии, который займет больше всего места при печати (если список литературы содержит меньше 100 источников, то указывается аргумент 99).

Каждый элемент списка начинается с команды `\bibitem`, которая имеет обязательный аргумент – условное обозначение, задаваемое автором работы.

В текст ссылка на литературу вставляется с помощью команды `\cite`, имеющий обязательный аргумент – условное обозначение источника, придуманное автором заранее. При ссылке на несколько источников в обязательный аргумент вводится несколько условных обозначений через запятую. Если необходимо добавить в ссылку пояснение, то это делается с помощью необязательного аргумента, который ставится перед обязательным в квадратных скобках, и будет напечатан через запятую после номера ссылки.

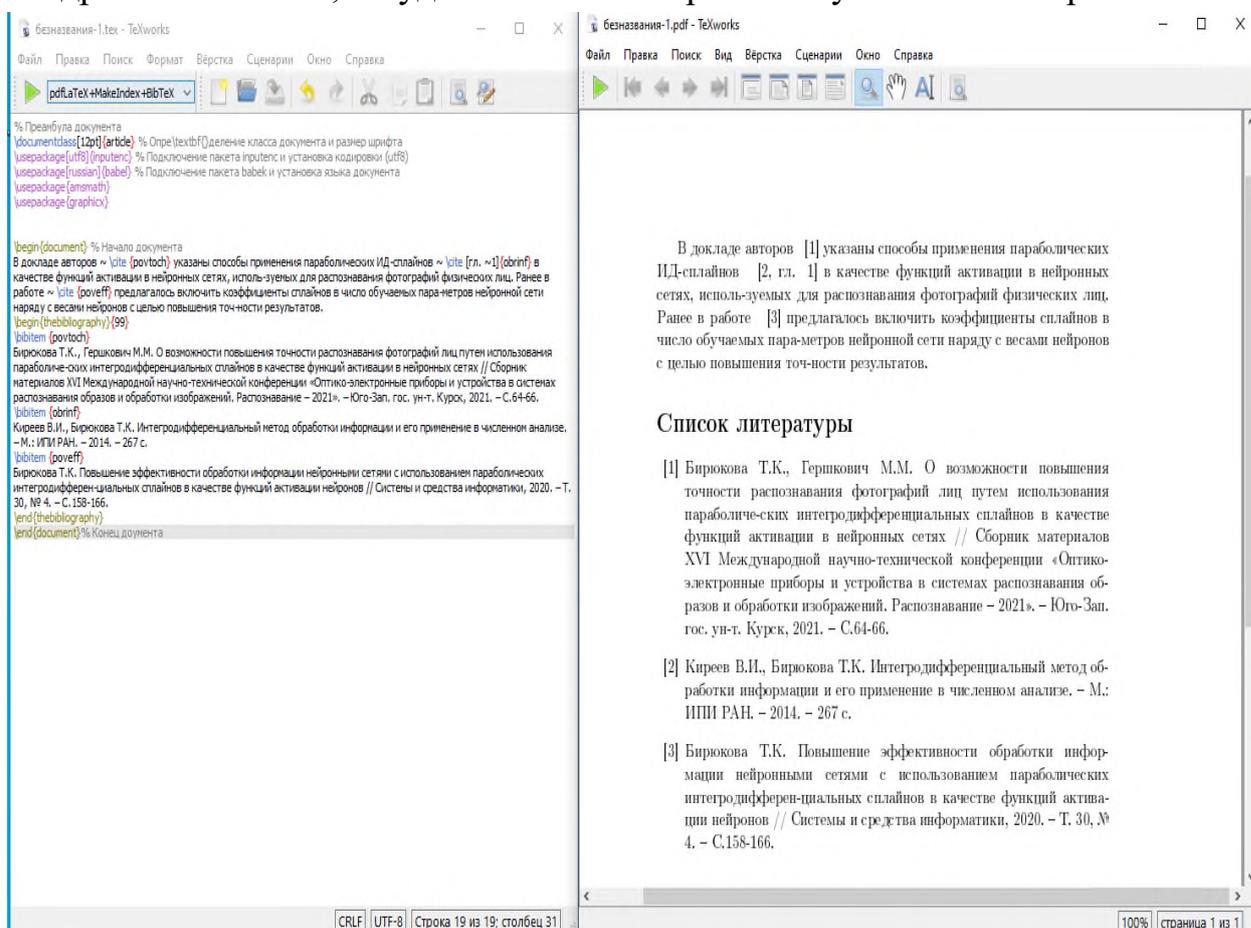


Рисунок 11 – Список литературы

## Задание

Составить отчет по результатам выполнения работы. Отчет должен содержать:

- титульный лист (Приложение 1);
- содержание (автособираемое);
- цель работы;
- содержательная часть;
- выводы по работе.

Содержательная часть отчета состоит из текста научной статьи, оформленной с помощью языка LaTeX и программного кода для верстки статьи. Текст научной статьи для оформления выдается преподавателем.

Нумерация страниц проставляется внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первая страница – титульный лист не нумеруется. На страницах с текстом статьи в колонтитуле указывается «Фамилия Имя Отчество, группа».

Для оформления текста Цели работы и выводов необходимо использовать параметры:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Все объекты в тексте (рисунки, таблицы, формулы) должны иметь нумерацию. В тексте на объекты и список литературы должны быть сделаны ссылки. Рисунки и таблицы должны иметь подписи (названия). Подпись рисунка имеет формат «Рисунок 1 – Название». Формулы должны быть выровнены по центру.

В разделе Выводы по работе необходимо написать, чему Вы научились, что узнали нового, какой инструмент текстового процессора, по Вашему мнению, является полезным, а какой – нет.

**Вопросы для защиты:**

1. Опишите структуру Latex-файла. Какие классы документов существуют?
2. Что такое окружение и какие бывают аргументы у специальных команд?
3. Как происходит обработка формул в Latex? Какие специальные команды и окружения применяются при верстке многострочных формул? Как оформить ссылку на формулу в тексте и происходит нумерация формул?
4. Какие применяются стили для текста в Latex и как создается оглавление?
5. С помощью каких окружений и команд оформляются таблицы и рисунки в Latex?
6. Как оформляется список литературы в Latex? Как оформить ссылку в тексте на источник из списка литературы?

**Список использованных источников**

1. Кузнецов, А.В. Основы LATEX : учебное пособие [Электронный ресурс] / А.В. Кузнецов. – М.: НИЯУ МИФИ, 2021. – 364 с. Режим доступа – <http://tug.ctan.org/info/russian/basiclatex-ru/BasicLatex.pdf>
2. Львовский С. М. Набор и вёрстка в системе LATEX [Электронный ресурс] / 5-е изд., переработанное / М.: МЦНМО, 2014. – 400 с. Режим доступа – <http://gidropraktikum.narod.ru/Lvovskii-2014.pdf>
3. Насыров, В. В. Пакеты прикладных программ для физиков: LATEX : учебное пособие [Электронный ресурс] / В.В. Насыров; [научный редактор А. И. Мазур]. — Хабаровск : Изд-во Тихоокеан. гос. ун-та, 2019. – 84 с. Режим доступа – [https://pnu.edu.ru/media/filer\\_public/a6/2b/a62b14a5-6c3a-418e-aba9-1d41684d8d9f/nasirov-posobie-pakets.pdf](https://pnu.edu.ru/media/filer_public/a6/2b/a62b14a5-6c3a-418e-aba9-1d41684d8d9f/nasirov-posobie-pakets.pdf)

### Оформление титульного листа отчета

Минобрнауки России  
ЮЗГУ  
ФФиПИ  
кафедра наименование кафедры

ЛАБОРАТОРНАЯ РАБОТА №1  
Технологии обработки текстовой информации

Выполнил(и):  
Проверил:

ст. гр. ХХ-016  
Фамилия И.О.  
Фамилия И.О.  
Иванова Е.Н.

Курск 2020 г.

Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 16 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

3: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

4: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

5: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по ширине, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки, позиция табуляции 12 см.

6: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 23.12.2021 11:40:58

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d79e5f1c11eabbf73e943df4a4851fda56d089

## МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра вычислительной техники



### Технологии обработки информации в таблицах

Методические указания  
по выполнению практической работы по дисциплине  
«Информатика» для студентов специальности  
10.05.02 Информационная безопасность  
телекоммуникационных систем

Курск 2021 г.

УДК 681.3

Составители: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина

Рецензент

Доцент кафедры программной инженерии,  
кандидат технических наук

*Ю.А. Халин*

**Технологии обработки информации в таблицах:** методические указания по выполнению практической работы / Юго-Зап. гос. ун-т; сост.: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина. – Курск, 2021. – 22 с.: ил. 3., табл. 1. – Библиограф.: с. 22.

Рассматриваются приемы использования встроенных средств табличного процессора для решения задач различного характера. Теоретический материал сопровождается примерами.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.  
Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## Цель работы

Получить знания о методах и средствах табличного процессора, позволяющих представлять числовую информацию в требуемом виде; научиться осуществлять выбор встроенного средства табличного процессора для получения результата, удовлетворяющего заданным параметрам; овладеть навыками использования различных инструментов, предоставляемых табличным процессором, при преобразовании и обработке числовой информации в соответствии с требованиями.

## Краткая теоретическая информация

Основным элементом таблицы является ячейка.

Ячейка – область, образованная пересечением строки и столбца. Она обозначается номером столбца и строки, на пересечении которых находится. Например, А1, В3. Адрес активной ячейки отображается в поле имени.

Диапазон (группа, блок) – непрерывная область ячеек, обозначенная номерами начальной и конечной ячеек, разделенных двоеточием или точкой, например, А1:С10, D8.Н12. Ячейке или диапазону может быть присвоено уникальное имя.

Присвоение или изменение имени осуществляется командой контекстного меню Присвоить имя... (рисунок 1).

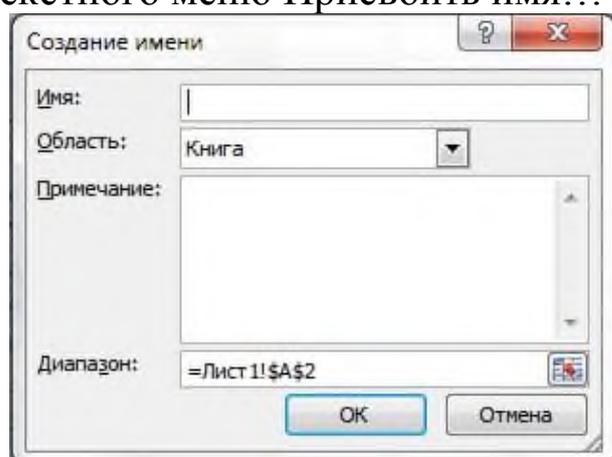


Рисунок 1 – Присвоение имени ячейки

Ячейка характеризуется следующими параметрами: адрес, содержание, значение, формат, статус.

Адрес ячейки может быть абсолютным, относительным и смешанным: относительный адрес: A1, E7; абсолютный адрес: \$A\$1, \$E\$7; смешанный адрес: \$A1, A\$1. Абсолютный адрес ячейки не меняется в операциях копирования, вставки или удаления ячеек, строк и столбцов. Если ячейке присвоен смешанный адрес, то при копировании будет меняться только тот параметр, перед которым не стоит знака \$. Например: для ячейки с адресом \$D6 – при копировании ячейки будет меняться только номер строки; D\$5 – при копировании будет меняться только адрес столбца.

В формулах возможны ссылки на адрес ячейки или на ее имя. Имя ячейки используется как абсолютный адрес.

Содержание ячейки. Содержание ячейки – это то, что вводится в нее через строку ввода. Поэтому ячейка может быть либо пустая, либо содержать данные: текст, текстовую константу, формулу, дату, время и пр.

Значение ячейки. Значением ячейки могут быть число, текстовая константа, дата, время, сообщения об ошибках. Значением пустой ячейки и ячейки, содержащей текст, является ноль.

Число может быть представлено в виде целого числа (123), вещественного числа с десятичной точкой (0,0001785) или в экспоненциальной форме (1,785E-4). Дробная часть числа отделяется от целой части запятой.

Текстовая константа – строка символов длиной до девяти знаков, используется в выражениях как операнд, при вводе текстовой константы она заключается в скобки и в кавычки, например, ("ноябрь").

Дата – значение функции дата. Дата вводится в следующей последовательности: день, месяц, год (17.05.99). В качестве разделителя используется точка. Электронная таблица позволяет выводить дату на экран в различных форматах.

Сообщения об ошибках:

#ДЕЛ/0! – деление на ноль;

#ИМЯ? – не определено имя переменной в формуле;

#Н/Д! – нет допустимых значений, аргумент функции не может быть определен;

#ПУСТО! – итога не существует;

#ЧИСЛО! – избыточное число либо неверное использование

числа, например, КОРЕНЬ(-1);

#ССЫЛКА! – неверная ссылка; ячейка, на которую она сделана, в рабочем листе не существует;

#ЗНАЧ! – неправильный тип аргумента; например, использование текста там, где необходимо число.

Если в формуле использовано одно из этих ошибочных значений, результат формулы также будет ошибочным. Ошибочные значения распространяются по всему рабочему листу, помечая все значения, зависящие от них, как некорректные. В этом случае достаточно найти и исправить ошибку, чтобы все остальные ячейки, связанные с ячейкой, содержащей ошибку, восстановили свое значение.

При вводе большого количества информации в ячейки таблицы легко допустить ошибку. В Excel существует инструмент для проверки введенных данных сразу после нажатия клавиши Enter – Проверка данных. Инструмент расположен в разделе Работа с данными в меню Данные.

## Проверка данных

Условия проверки вводимого значения можно настроить очень гибко (рисунок 2).

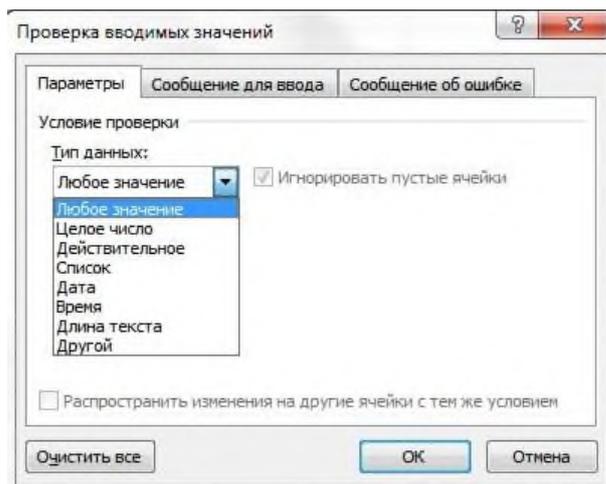


Рисунок 2 – Диалог Проверка вводимых значений

Можно разрешить ввод значений в ячейку только определенного типа, выбрав необходимое условие из выпадающего списка

Чтобы отображать комментарий, необходимо использовать

вкладку Сообщение для вывода (рисунок 3).

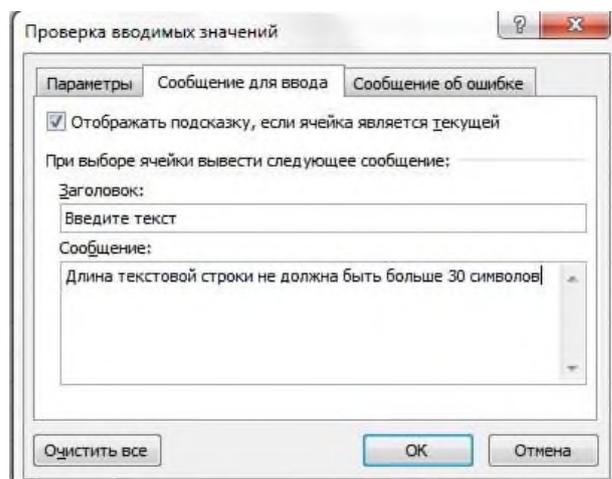


Рисунок 3 – Настройка сообщения – предупреждения

В отличие от обычного примечания, которое пропадает после того, как курсор мыши уходит с ячейки, этот комментарий отображается всегда, когда ячейка выделена. После ввода ошибочного значения Проверка данных может отобразить подробное сообщение о том, что было сделано не так (рисунок 4).

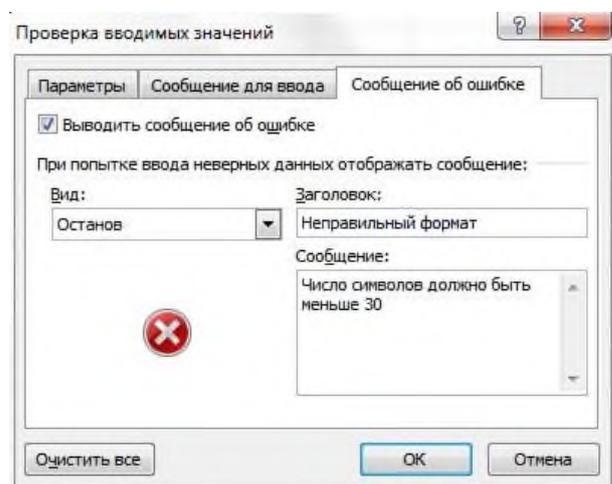


Рисунок 4 – Формирование сообщения о допущенной ошибке

Если в качестве Типа данных на вкладке Параметры выбрать Список, то можно сформировать связанные диапазоны. Если в Проверке данных нужно сделать ссылку на ячейку другого листа, то нужно сначала определить Имя для этой ячейки, а затем сослаться на это имя в правиле Проверки данных.

Проверка данных явно срабатывает при вводе в ячейку значе-

ний с клавиатуры с последующим нажатием клавиши Enter. В этом случае появляется окно с описанием ошибки (рисунок 5).

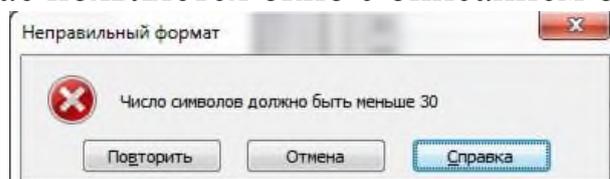


Рисунок 5 – Окно с описанием ошибки

Если значения вставляются через Буфер обмена или с использованием Автозаполнения, то проверка в явном виде не осуществляется. Чтобы убедиться, что данные в ячейке не соответствуют условиям, определенным в Проверке данных, нужно вызвать команду Обвести неверные данные из раздела Проверка данных меню Данные.

### Формат ячейки

К формату ячейки относятся ширина, режим отображения формул, формат отображения числовых величин, размещение содержимого ячейки, шрифт, цвет, границы, статус ячейки. Настройка параметров ячейки осуществляется с помощью окна диалога Формат ячеек. Выбрать нужный формат можно с помощью инструмента Формат раздела Ячейки в меню Главная. При этом откроется окно Формат ячеек (рисунок 6).

Ширина ячейки может быть от 1 до 127 символов, по умолчанию - 9 символов.

Формат отображения числовых величин: в виде целого (16, 154) или вещественного числа (1,1755, 5,439), в показательной форме (1,45E-4), в денежном формате (345,32) или (\$345,32), в процентном формате (35%). При представлении

Размещение содержимого ячейки. Содержимое ячейки может быть размещено справа, слева, по центру. По умолчанию текст прижимается к левому краю, числовые значения – к правому краю. Текст может быть размещен горизонтально, вертикально или под определенным углом. Управление размещением содержимого ячейки осуществляется командой Формат ячеек Выравнивание (рисунок 7).

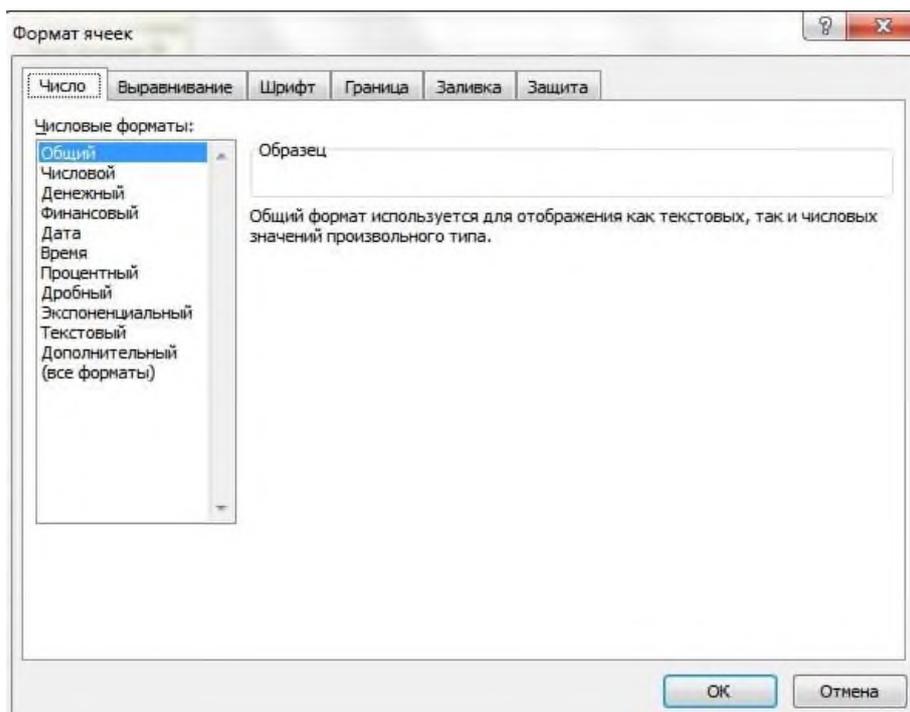


Рисунок 6 – Окно «Формат ячеек»

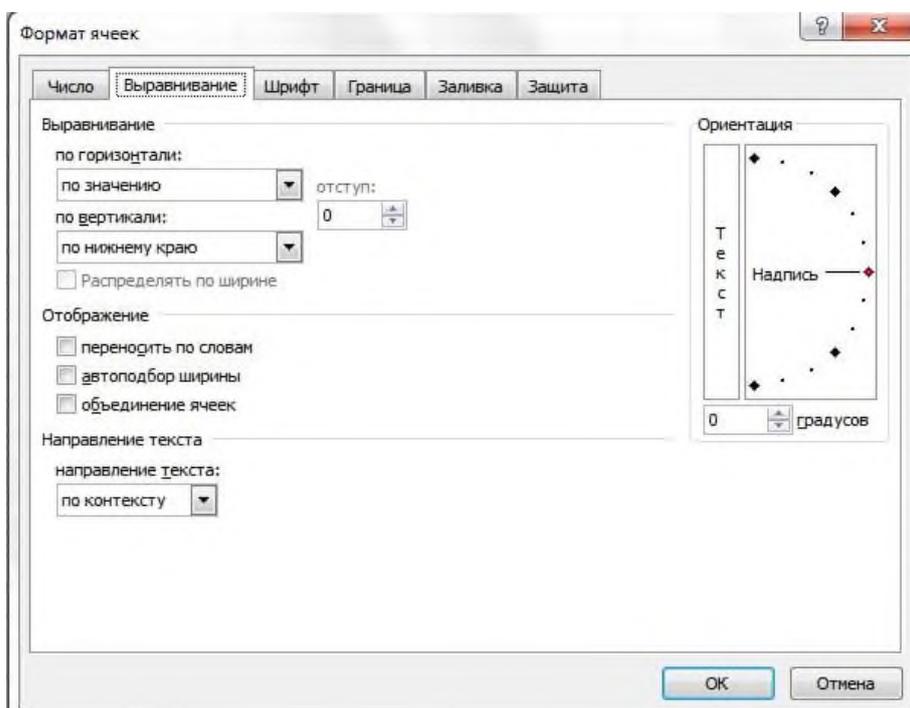


Рисунок 7 – Настройка свойств ячеек

Важное значение имеет флажок **Переносить по словам**. При установке этого флажка текст будет автоматически переноситься в пределах установленной ширины столбца. Если флажок **Перено-**

силь по словам сброшен, то вводимый текст располагается в одну строку, и если соседняя ячейка заполнена, то на экране будет видна только часть текста, уместяющаяся в ячейке.

Стили оформления и заполнения ячеек устанавливаются командами Формат ячеек Границы и соответствующими кнопками на Ленте. Ячейка может иметь два статуса: защищена или не защищена. В защищенную ячейку нельзя внести информацию или изменить ее содержание. Установка режима защиты осуществляется командой Формат ячеек Защита. Режим защиты ячейки вступает в силу только после защиты листа командой Рецензирование Защитить лист.

### Закрепление областей

При работе с таблицами, содержащими большое число строк и столбцов, возникают затруднения, так как исчезает либо шапка таблицы, либо левая колонка. Excel имеет средства, позволяющие зафиксировать шапку таблицы, верхнюю строку или первый столбец или и то и другое вместе. Для этого применяется вкладка Ленты «Вид» Закрепить области и выбрать необходимое.

Очень неудобно работать с таблицей, если не видны заголовки строк и столбцов. Для устранения этого недостатка области таблицы следует закрепить, что позволяет при просмотре областей списка одновременно видеть на экране часть его заголовка и расположенные слева столбцы. С целью закрепления областей на ленте вкладка Вид Закрепить области (рисунок 8).

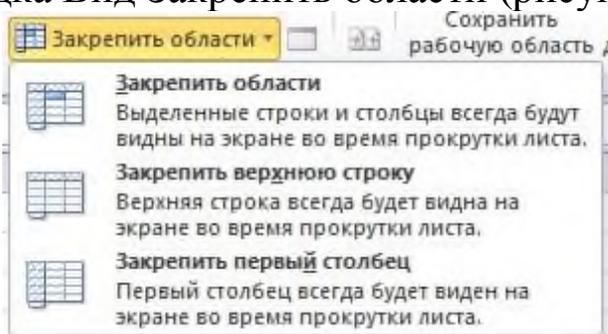


Рисунок 8 – Закрепление областей

Для отмены закрепления областей следует выполнить команду на ленте вкладка Вид щелкнуть по пиктограмме Закрепить области

и из списка выбрать Снять закрепление областей.

### Автозаполнение

Автозаполнение можно использовать для ввода в смежные ячейки одного столбца или одной строки последовательных рядов календарных данных, времени, чисел, комбинаций текста и чисел. Кроме того, можно создать собственный список автозаполнения.

Для автозаполнения ячеек необходимо в первую из заполняемых ячеек ввести начальное значение ряда. Затем следует выделить ячейку и при нажатой левой кнопке мыши перетаскивать маркер автозаполнения (маленький черный квадрат в правом нижнем углу выделенной ячейки) в сторону изменения значений. Указатель мыши при наведении на маркер принимает вид черного креста (рисунок 9). При перетаскивании вправо или вниз значения будут увеличиваться (рисунок 9 – рисунок 10), при перетаскивании влево или вверх - уменьшаться (рис. 12).

	А	В	С
1	Дни недели	Дата	
2	Понедельник	02.ноя	
3	Вторник		
4	Среда		
5	Четверг		
6	Пятница		
7	Суббота		
8	Воскресенье		
9			08.ноя

а)

	А	В	С
1	Дни недели	Дата	
2	Понедельник		
3	Вторник		
4	Среда		
5	Четверг		
6	Пятница		
7	Суббота		
8	Воскресенье	08.ноя	
9			02.ноя

б)

Рисунок 9 – Автозаполнение по столбцу с возрастанием (а) и убыванием (б)

	А	В	С	Д	Е	Ф	Г	Н
1	Дни недели	Понедельник	Вторник	Среда	Четверг	Пятница	Суббота	Воскресенье
2	Дата	02.ноя						
3								08.ноя

Рисунок 10 – Автозаполнение по строке с возрастанием

По окончании перетаскивания рядом с правым нижним углом заполненной области появляется кнопка Параметры автозаполнения. Для заполнения последовательным рядом чисел, а также для выбора способа заполнения календарными рядами после перетас-

кивания необходимо щелкнуть левой кнопкой мыши по кнопке Параметры автозаполнения и выбрать требуемый режим автозаполнения.

В появившемся меню для заполнения последовательным рядом чисел следует выбрать команду Заполнить.

## Формулы

Формулы представляют собой выражения, по которым выполняются вычисления. Формула всегда начинается со знака равно (=). Формула может включать функции, ссылки, операторы и константы.

Функция – стандартная формула, которая обеспечивает выполнение определенных действий над значениями, выступающими в качестве аргументов. Функции позволяют упростить формулы, особенно если они длинные или сложные.

Ссылка указывает на ячейку или диапазон ячеек листа, которые требуется использовать в формуле. Можно задавать ссылки на ячейки других листов той же книги и на другие книги. Ссылки на ячейки других книг называются связями.

Оператором называют знак или символ, задающий тип вычисления в формуле. Существуют математические, логические операторы, операторы сравнения и ссылок.

Константой называют постоянное (невычисляемое) значение. Формула и результат вычисления формулы константами не являются.

Решение систем линейных алгебраических уравнений

Основная запись системы линейных алгебраических уравнений

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1N}x_N = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2N}x_N = b_2 \\ \dots \\ a_{M1}x_1 + a_{M2}x_2 + \dots + a_{MN}x_N = b_M \end{cases}$$

Матричная формулировка имеет вид  $A \times X = B$ , где

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \dots & & & \\ a_{M1} & a_{M2} & \dots & a_{MN} \end{pmatrix}, B = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_N \end{pmatrix}, X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{pmatrix}$$

Решение системы уравнений в матричной формулировке  $X = A^{-1} \times B$ ,

где  $A^{-1}$  – матрица, обратная матрице  $A$ .

Для вычисления обратной матрицы – функция МОБР, а для умножения матрицы  $A^{-1}$  на вектор  $B$  – функция МУМНОЖ.

Для запуска этих функций следует пользоваться только комбинацией клавиш Ctrl+Shift+Enter

### Графическое представление данных

Диаграммы являются средством наглядного представления данных и облегчают выполнение сравнений, выявление закономерностей и тенденций данных.

Диаграммы создаются на основе данных, расположенных на рабочих листах. Как правило, используются данные одного листа. Это могут быть данные диапазонов как смежных, так и несмежных ячеек. Несмежные ячейки должны образовывать прямоугольник. При необходимости, в процессе или после создания диаграммы, в нее можно добавить данные, расположенные на других листах.

Диаграмма может располагаться как графический объект на листе с данными (не обязательно на том же, где находятся данные, взятые для построения диаграммы). На одном листе с данными может находиться несколько диаграмм. Диаграмма может располагаться на отдельном специальном листе.

Диаграмму можно напечатать. Диаграмма, расположенная на отдельном листе, печатается как отдельная страница. Диаграмма, расположенная на листе с данными, может быть напечатана вместе с данными листа или на отдельной странице.

Диаграмма постоянно связана с данными, на основе которых она создана, и обновляется автоматически при изменении исходных данных. Более того, изменение положения или размера элементов

данных на диаграмме может привести к изменению данных на листе.

Для построения диаграммы необходимо выделить диапазон ячеек. Затем на ленте во вкладке Вставка в группе Диаграммы выбрать требуемый тип диаграммы и щелкнуть по соответствующей пиктограмме или вызвать диалоговое окно Вставка диаграммы (рисунок 11).

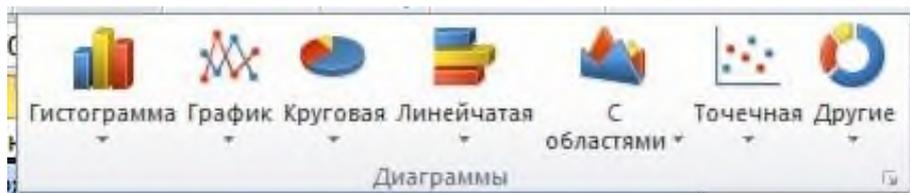


Рисунок 11 – Виды диаграмм

В диалоговом окне Вставка диаграммы выберете категорию диаграммы, после нажмите ОК, выбранная Вами диаграмма отразится на листе.

При выделении диаграммы на ленте появляются дополнительные вкладки для работы с диаграммами (Конструктор, Макет и Формат).

Теперь можно модифицировать созданную диаграмму, воспользовавшись предлагаемым набором инструментов.

Возможно изменение источника данных для созданной прежде диаграммы, для этого щелкните правой кнопкой в области диаграммы и выберите в контекстном меню пункт Выбрать данные, после этого появится диалоговое окно Выбор источника данных (рисунок 12).

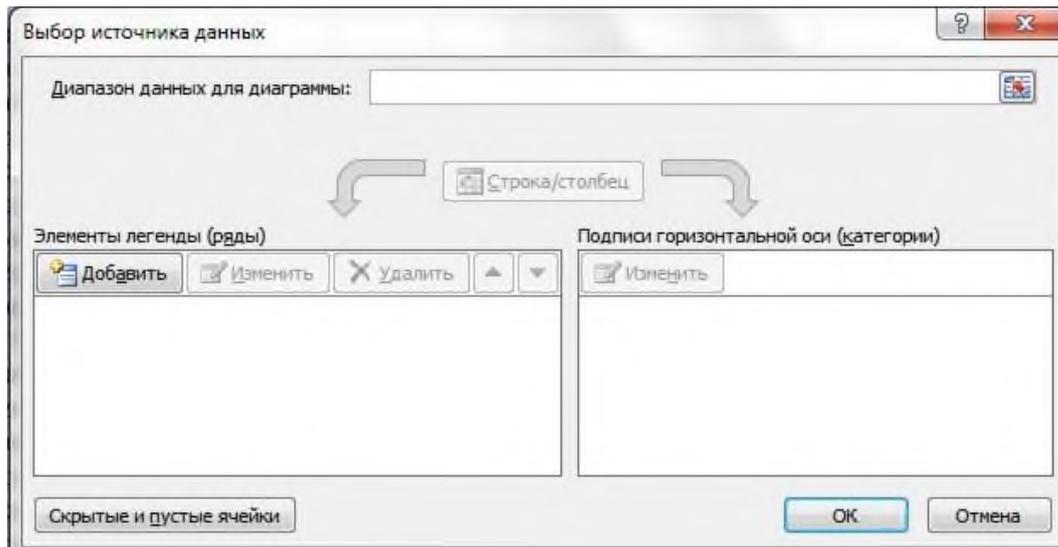


Рисунок 12 – Выбор источника данных

Опишем поля в этом диалоговом окне. Поле Диапазон данных для диаграммы. Здесь указывают диапазон данных ячеек, служащих источником данных для диаграммы.

Панель Элементы легенды (ряды). В этой панели можно добавить новые записи, отредактировать либо вовсе удалить прежние записи.

Панель Подписи горизонтальной оси (категории). Здесь можно удалить или изменить надписи на горизонтальной оси диаграммы.

Кнопка Строка/Столбец. При нажатии на эту кнопку меняются местами все значения

Еще одной возможностью изменения диаграммы является форматирование области, в которой была построена диаграмма. Для этого щелкните правой кнопкой мыши в области диаграммы и в контекстном меню выберите Формат области диаграммы. Появится диалоговое окно Формат области диаграммы.

## Задание

1. Обеспечить проверку корректности ввода чисел в заданной системе счисления, длина целой и дробной части которого определяется согласно варианта (таблица 1).

Требования к выполнению задания.

– при вводе числа в определенную ячейку должна появляться

подсказка о системе счисления и длине целой и дробной частей;

- после ввода (нажатия клавиши Enter) в случае ошибки должно появляться сообщение, какая именно произошла ошибка

2. Решить заданную систему уравнений (таблица 2), используя встроенные функции Excel для выполнения операций над матрицами.

Требования к выполнению задания.

- система уравнений должна быть записана в явном виде и в матричном;

- все матрицы должны иметь имена.

3. Построить график функции (по вариантам, таблица 3).

Требования к выполнению задания.

- заданная функция должна быть протабулирована от минимального значения аргумента до максимального с определенным шагом согласно задания;

- в качестве вида диаграмм следует использовать График (График с маркерами);

- Легенды на графике быть не должно;

- координатные оси должны быть подписаны, на них должны быть обозначены отрезки по делениям, пересекаться оси должны в центре координат (точка  $O(0; 0)$ ), оси должны иметь направление;

- Интервал между делениями и Интервал между подписями может быть задан по контексту;

- промежуточные линии сетки либо не показываются, либо показываются по горизонтальной и по вертикальной оси, причем их толщина должна быть меньше толщины основных осей;

- график должен иметь название, при записи математических операций в названии функций не следует использовать символы «\*», «^» и т.п.

Общие требования

Каждое задание выполняется на отдельном Листе Книги. Каждый Лист должен иметь тематическое название. При оформлении работы в Excel необходимо использовать шрифт Times New Roman, размер 14, начертание обычное. Размеры ячеек выбираются по контексту, каждая ячейка должна иметь пояснения, какая информация в ней находится.

Составить отчет по результатам выполнения лабораторной ра-

боты. Отчет должен содержать:

- титульный лист (Приложение А);
- Содержание;
- Цель работы;
- содержательная часть;
- Выводы по работе.

Содержательная часть отчета состоит из краткого описания результатов работы в Excel. Каждое задание выделяется в отдельный раздел, который должен иметь название. По каждому заданию приводится словесное описание, которое подтверждается скриншотами выполнения в Excel. Если задание связано с расчетами, обязательно приводятся скриншоты Excel с отображением формул.

Номера страниц проставляются внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первой страницей является титульный лист, нумерация проставляется, начиная с содержательной части отчета.

Макет Оглавления приведен в Приложении Б.

Для оформления заголовков разделов следует использовать стиль «Заголовок 1». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 12 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Для оформления текста отчета необходимо использовать стиль «Обычный». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

## Варианты задания 1

вариант	основание системы счисления	длина целой части	длина дробной части
1	2	4	4
2	3	5	3
3	4	6	2
4	5	7	3
5	6	8	4
6	7	7	3
7	8	6	2
8	9	5	3
9	10	4	4
10	11	5	3
11	12	6	2
12	13	7	3
13	14	8	4
14	15	7	3
15	16	6	2
16	8	4	4

Таблица 2

## Варианты задания 2

вариант	матрица А	вектор В
1	$\begin{pmatrix} 2 & 3 & 4 & 5 \\ 5 & 2 & 5 & 7 \\ 4 & 2 & 7 & 1 \\ 7 & 5 & 1 & 4 \end{pmatrix}$	$\begin{pmatrix} 6 \\ 2 \\ 3 \\ 2 \end{pmatrix}$
2	$\begin{pmatrix} 1 & 4 & 2 & 5 \\ 4 & 4 & 5 & 3 \\ 1 & 2 & 6 & 8 \\ 3 & 7 & 3 & 2 \end{pmatrix}$	$\begin{pmatrix} 8 \\ 6 \\ 7 \\ 9 \end{pmatrix}$
3	$\begin{pmatrix} 9 & 6 & 3 & 8 \\ 4 & 6 & 7 & 4 \\ 2 & 3 & 5 & 3 \\ 4 & 8 & 3 & 7 \end{pmatrix}$	$\begin{pmatrix} 3 \\ 8 \\ 5 \\ 9 \end{pmatrix}$
4	$\begin{pmatrix} 2 & 4 & 7 & 4 \\ 4 & 1 & 6 & 2 \\ 8 & 3 & 6 & 7 \\ 6 & 3 & 5 & 7 \end{pmatrix}$	$\begin{pmatrix} 2 \\ 0 \\ 3 \\ 1 \end{pmatrix}$
5	$\begin{pmatrix} 3 & 3 & 4 & 7 \\ 2 & 6 & 4 & 6 \\ 3 & 4 & 5 & 6 \\ 1 & 9 & 3 & 5 \end{pmatrix}$	$\begin{pmatrix} 3 \\ 4 \\ 8 \\ 2 \end{pmatrix}$
6	$\begin{pmatrix} 2 & 1 & 5 & 2 \\ 5 & 2 & 2 & 6 \\ 2 & 2 & 1 & 2 \\ 1 & 3 & 3 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 3 \\ 0 \\ 2 \end{pmatrix}$

7	$\begin{pmatrix} 7 & 6 & 2 & 7 \\ 4 & 9 & 5 & 5 \\ 2 & 3 & 4 & 4 \\ 1 & 5 & 6 & 6 \end{pmatrix}$	$\begin{pmatrix} 3 \\ 2 \\ 0 \\ 2 \end{pmatrix}$
8	$\begin{pmatrix} 0,12 & -0,43 & 0,14 & 0,64 \\ -0,07 & 0,34 & -0,72 & 0,32 \\ 1,18 & -0,08 & -0,25 & 0,43 \\ 1,17 & 0,53 & -0,84 & -0,53 \end{pmatrix}$	$\begin{pmatrix} -0,17 \\ 0,62 \\ 1,12 \\ 1,15 \end{pmatrix}$
9	$\begin{pmatrix} 0,12 & -0,43 & 0,14 & 0,64 \\ -0,07 & 0,34 & -0,72 & 0,32 \\ 1,18 & -0,08 & -0,25 & 0,43 \\ 1,17 & 0,53 & -0,84 & -0,53 \end{pmatrix}$	$\begin{pmatrix} -0,17 \\ 0,62 \\ 1,12 \\ 1,15 \end{pmatrix}$
10	$\begin{pmatrix} 3,7 & 5,6 & 9,5 & 2 \\ 4 & 3,36 & 31,1 & 1,5 \\ 2 & 7,93 & 4,2 & 6,3 \\ 2 & 42,7 & 3,7 & 6,2 \end{pmatrix}$	$\begin{pmatrix} 13 \\ 0 \\ 4,4 \\ 3 \end{pmatrix}$
11	$\begin{pmatrix} 1,3 & 1,6 & 5 & 2,2 \\ 4,4 & 6,7 & 13 & 2,5 \\ 2,8 & 0,73 & 12 & 67,8 \\ 2 & 3,4 & 13 & 6 \end{pmatrix}$	$\begin{pmatrix} 3 \\ 0 \\ 4 \\ 3 \end{pmatrix}$
12	$\begin{pmatrix} 1,3 & 1,6 & 5 & 2,2 \\ 4,4 & 6,7 & 13 & 2,5 \\ 2,8 & 0,73 & 12 & 67,8 \\ 2 & 3,4 & 13 & 6 \end{pmatrix}$	$\begin{pmatrix} 3 \\ 0 \\ 4 \\ 3 \end{pmatrix}$

13	$\begin{pmatrix} 5,3 & 1,6 & 5,5 & 2 \\ 4,4 & 6,7 & 13 & 2,5 \\ 2,8 & 0,73 & 12 & 67,8 \\ 2 & 3,4 & 13 & 6 \end{pmatrix}$	$\begin{pmatrix} 3 \\ 0 \\ 4 \\ 3 \end{pmatrix}$
14	$\begin{pmatrix} 1,3 & 1,6 & 5 & 2,2 \\ 4,1 & 6,4 & 3,9 & 5 \\ 2,1 & 3,3 & 2,04 & 6 \\ 2 & 4 & 3 & 6 \end{pmatrix}$	$\begin{pmatrix} 3,3 \\ 0 \\ 4,9 \\ 3,1 \end{pmatrix}$
15	$\begin{pmatrix} 3 & 6 & 5 & 0,2 \\ 4 & 6 & 8,3 & 5,3 \\ 2 & 3 & 2,6 & 6,1 \\ 2 & 4 & 0,93 & 6 \end{pmatrix}$	$\begin{pmatrix} 3 \\ 0 \\ 4,1 \\ 3,8 \end{pmatrix}$
16	$\begin{pmatrix} 3 & 6 & 5 & 2 \\ 4 & 6 & 3,6 & 5 \\ 2 & 3,4 & 2 & 6 \\ 2 & 44,7 & 3 & 6 \end{pmatrix}$	$\begin{pmatrix} 34,7 \\ 0 \\ 4,2 \\ 3 \end{pmatrix}$

## Варианты задания 3

вариант	функция	область определения	шаг изменения аргумента
1	$y = \sin x$	$x \in [-2\pi; 2\pi]$	$\Delta x = \frac{\pi}{8}$
2	$y = \cos x$	$x \in \left[-\frac{3\pi}{2}; \frac{3\pi}{2}\right]$	$\Delta x = \frac{\pi}{8}$
3	$y = 2 \cdot  x  - 3$	$x \in [-5; 5]$	$\Delta x = 0,25$
4	$y = \operatorname{tg} x$	$x \in \left[\frac{\pi}{12}; \frac{11\pi}{12}\right]$	$\Delta x = \frac{\pi}{24}$
5	$y = \operatorname{ctg} x$	$x \in \left[-\frac{5\pi}{12}; \frac{5\pi}{12}\right]$	$\Delta x = \frac{\pi}{24}$
6	$y = x^2 + x^{-3}$	$x \in [-1; 1]$	$\Delta x = 0,125$
7	$y = \lg x + \log_2 x$	$x \in [1; 32]$	$\Delta x = 3$
8	$y = \frac{1}{x^2 + 2x - 3}$	$x \in [-2; 3]$	$\Delta x = 0,25$
9	$y = \frac{1}{x^2 - 3x + 1}$	$x \in [-5; 5]$	$\Delta x = 0,5$
10	$y = e^x + \sqrt{x}$	$x \in [0; 2]$	$\Delta x = 0,25$
11	$y = \ln x$	$x \in [1; 21]$	$\Delta x = 0,5$
12	$y = x^2 -  x  - 2$	$x \in [-3; 4]$	$\Delta x = 0,25$
13	$y = \cos\left(x + \frac{\pi}{12}\right)$	$x \in \left[-2\pi; \frac{5\pi}{6}\right]$	$\Delta x = \frac{\pi}{12}$
14	$y = x^2 - 3$	$x \in [-3; 4]$	$\Delta x = 0,25$
15	$y = \frac{x^2}{2} + 3,5$	$x \in [-5; 3]$	$\Delta x = 0,25$
16	$y = x^3 - 3$	$x \in [-2; 2]$	$\Delta x = 0,2$

**Список использованных источников**

1. Колокольникова, А.И. Информатика [Текст] : учебное пособие / А.И. Колокольникова. – Москва ; Берлин : Директ-Медиа, 2020. – 289 с. : ил., табл. – Режим доступа: URL: <https://biblioclub.ru/>

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 23.12.2021 11:40:58

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d79e5f1c11eabbf73e943df4a4851fda56d089

## МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра вычислительной техники



### Информационные технологии для математических вычислений

Методические указания  
по выполнению практической работы по дисциплине  
«Информатика» для студентов специальности  
10.05.02 Информационная безопасность  
телекоммуникационных систем

Курск 2021 г.

УДК 681.3

Составители: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина

Рецензент

Доцент кафедры программной инженерии,  
кандидат технических наук

*Ю.А. Халин*

**Информационные технологии для математических вычислений:** методические указания по выполнению практической работы / Юго-Зап. гос. ун-т; сост.: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина. – Курск, 2021. – 31 с.: табл. 5. – Библиограф.: с. 27.

Рассматриваются практические приложения и рекомендации относительно широкого использования математического пакета SMathStudio для проведения вычислений, выполнения упрощений математических выражений и построения графиков функций. Теоретический материал сопровождается примерами.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.  
Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## Цель работы

Получить знания о вычислительных возможностях математического пакета; научиться анализировать возможности средств математического пакета при решении прикладных задач; приобрести навык использования различных инструментов пользовательского интерфейса математического пакета.

## Краткая теоретическая информация

### Интерфейс программы

Интерфейс программы состоит из трех областей:

- Основное меню
- Инструментальная панель
- Рабочее поле

В основное меню (рисунок 1) входят такие элементы, как Вставка, Вычисление, Файл и т.п.

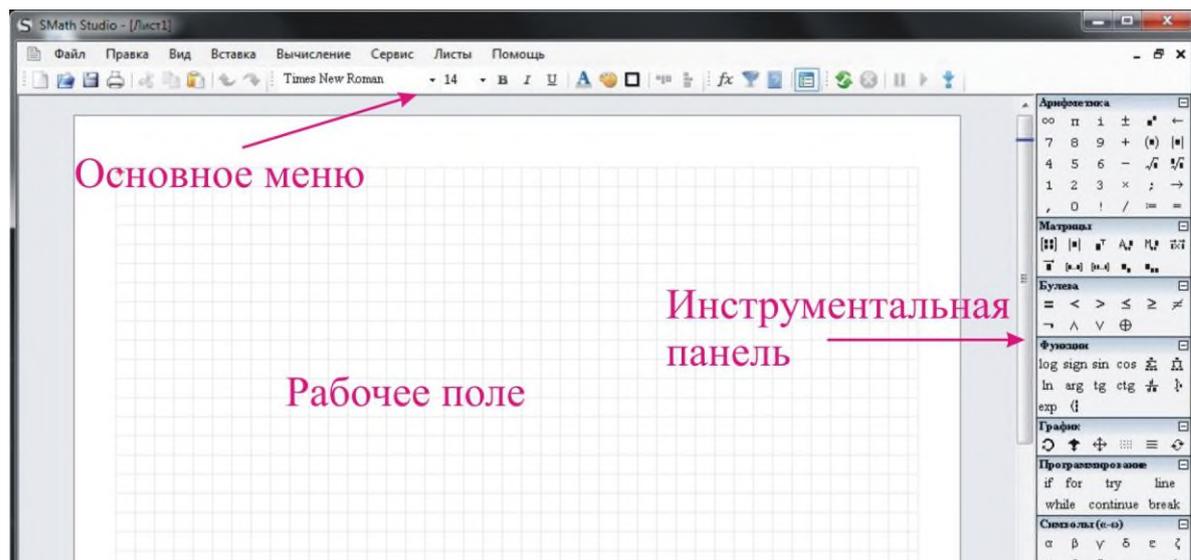


Рисунок 1 – Интерфейс программы SMath Studio

Инструментальная панель разделена по категориям:

- Панель «Арифметика» (рисунок 2).

Содержит цифры, математические символы, и основные операции.

- Панель «Матрицы» (рисунок 2).

Содержит команды для работы с матрицами. Позволяет нахо-

дальше определитель матрицы, транспонировать её, находить минор. А также содержит команду векторного умножения (векторы программа рассматривает как матрицу с одним столбцом или строкой).

– Панель «Булева» (рисунок 2).

Эта панель содержит набор для команд для булевой алгебры. А также позволяет задавать логические операции в командах ветвления и циклах.

– Панель «Функции» (рисунок 2).

Содержит набор часто используемых функций, таких как *sin*, *cos*, *log* и т.п.

– Панель «График» (рисунок 2).

Эта панель позволяет вращать, перемещать, увеличивать/уменьшать графики функций.

Двумерные графики строятся по переменной *x*, а трехмерные по двум переменным *x*, *y* (переменные должны вводиться в нижнем регистре).

– Панель «Программирование» (рисунок 2).

Содержит функции программирования, такие как: ветвление «IF», циклы «WHILE» и «FOR», вспомогательная функция «LINE».

– Последние две панели называются одинаково «Символы» и содержат греческие символы.



Рисунок 2 – Панели инструментов SMATH Studio

Рабочее поле занимает самую большую часть программы, здесь проводятся все вычисления. Основным элементом поля явля-

ется курсор или Фокус ввода (место, где будет набираться выражение), он может выглядеть в одном из трех вариантов:

- как красный крестик, если курсор определяет местоположение следующего объекта;
- угловой курсор используется при введении формул, указывает на текущий элемент выражения;
- в виде I-образной вертикальной черты, используется при вводе текста.

#### Основные правила написания расчетов в программе

Используемая в расчете переменная/функция должна быть заранее объявлена в программе, т.е. находиться выше либо левее от выражения. При объявлении переменной, можно использовать ранее объявленные переменные, встроенные и ранее объявленные функции, либо их сочетания. Например:  $A := \sin(b)$  (где переменная  $b$  объявлена выше либо левее).

Для символьных вычислений переменные объявлять не обязательно.

Переменная может содержать в себе не только скалярное значение (численное), но и матрицу/вектор.

При работе с Smath Studio доступны «горячие» клавиши (таблица 1):

Таблица 1

#### Горячие клавиши Smath Studio

Сочетание клавиш	Результат
Shift + :	Вставка оператора присвоения «:=» с панели «Арифметика»
Ctrl + .	Вставка оператора символьного вычисления «→» с панели «Арифметика»
=	Вставка оператора численного вычисления «=» с панели «Арифметика»
Enter	Завершение ввода выражения (курсор перемещается на строчку ниже)

## Единицы измерения и автоматическая помощь ввода

Smath Studio работает с именованными численными значениями. Единицы измерения можно рассматривать как встроенные переменные. SMath Studio использует единицы измерения системы единицы СИ (также известная как Международная система Единиц).

Чтобы назначить единицу измерения, нужно на нее умножить или вставить сразу после числового значения. Также можно использовать диалоговое окно Вставить единицу измерения, вызванное из меню Вставка (рисунок 3).

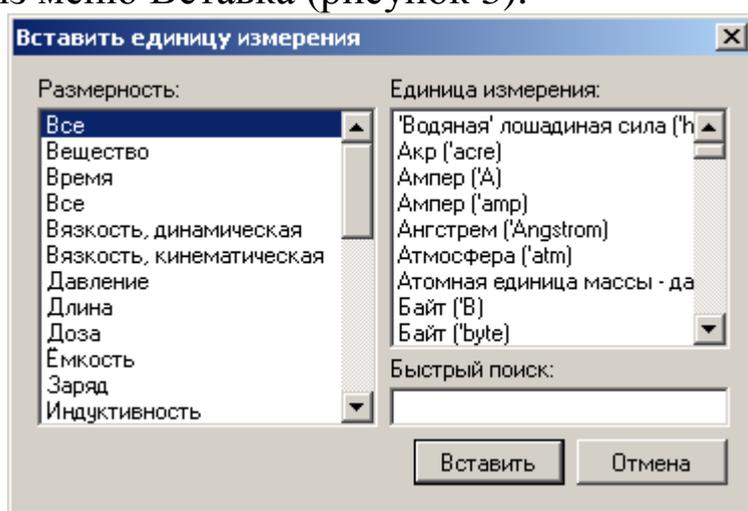


Рисунок 3 – Выбор единицы измерения

При работе с единицами измерения, результаты вычислений могут получиться не в тех единицах, которые вы хотели бы видеть. Для того чтобы исправить это, достаточно кликнуть мышкой на выражение, навести курсор на черный квадратик справа (это поле ввода), нажать  $\text{Ctrl} + \text{w}$  или открыть окно Единица измерения из меню Вставка и выбрать единицу из списка (рисунок 3).

Smath Studio можно использовать как простой калькулятор, для этого в любое место рабочего поля помещаем курсор и осуществляем ввод математического выражения с клавиатуры. Например, после ввода  $2 \cdot 27/3 + 2^4 - 3 =$  на листе Smath Studio будет следующее:

$$2 \cdot \frac{27}{3} + 2^4 - 3$$

Переменная – участок памяти компьютера, у которого есть

имя, он может содержать в себе число, матрицу или текст.

Матрица – прямоугольная таблица с элементами, которая может содержать другие матрицы, числа, текст и переменные.

Текст – любые символы, заключенные в двойные кавычки.

В Smach Studio переменные могут содержать:

– текст, например:

```
A := "надпись или любой другой текст" ;
```

– число, например:

```
A := 9
```

или

```
A := 5, 67
```

или

```
A := - 278, 43 ;
```

```
A := [ 3 2 ]
      [ -5 4 ]
      [ 0 1 ]
```

– матрицу, например:

При объявлении переменной, можно использовать: ранее объявленные переменные, встроенные и ранее объявленные функции либо их сочетания. Для символьных вычислений переменные объявлять не обязательно. Программа чувствительна к регистру букв, т.е. переменные «Value», «VALUE», «value» и «VaLuE» воспринимаются как разные, и никак между собой не связанные.

Знак присвоения («:=») позволяет присвоить переменной значение. Переменная будет равна этому значению ниже операции присвоения.

Любая логическая операция может вернуть одно из двух значений, ложь (false) или истина (true). В SMach Studio значение ложь(false) соответствует «0», а значение истина (true) – 1.

## Действия с матрицами

Матрица в общем случае это прямоугольный набор элементов, который может содержать:

– числа (целые, вещественные, комплексные);

- строковые значения;
- другие матрицы.

В SMath Studio нумерация элементов матрицы начинается с 1. Если вы будете использовать какую-либо функцию с нулевым элементом, то программа выдаст ошибку.

Создать матрицу можно несколькими способами:

- при помощи команды `mat` или кнопки «Кубическая матрица» с панели «Матрицы». В результате проделанных действий появится пустая матрица.

- командой `matrix(arg1, arg2)`, где `arg1` – число строк, `arg2` – число столбцов. Так `matrix(2, 3)` создаст матрицу, заполненную нулями, состоящую из 2 строк и 3 столбцов:

В SMath Studio существуют две команды, которые позволяют обращаться к элементам вектора и матрицы по отдельности: `el([вектор];[номер элемента])` и `el([матрица];[номер строки];[номер столбца])`. С помощью встроенных функций SMath Studio матрицы можно объединять, выделять в них подмассивы, определять размеры массивов, максимальные, минимальные значения, находить собственные числа и векторы. Для матриц определены следующие операции: умножение, обращение, транспонирование, и т.п.

Для работы с элементами матрицы используются индексы элементов. Индекс элемента определяется на панели инструментов Матрицы. Для проведения операций с матрицами используется панель инструментов Матрицы.

Работа с графиками

Графики функции в SMath Studio бывают 2-х видов: двумерные и трехмерные. Для работы с ними предназначена специальная панель «График».

Вставить график в расчет можно несколькими способами:

- 2D-график комбинацией `Shift + @`.
- при помощи меню Вставка (рисунок 4).

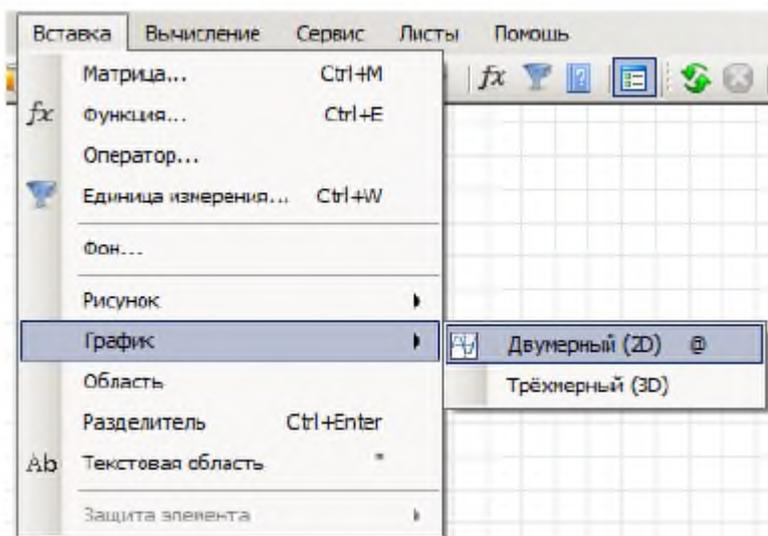


Рисунок 4 – Вставка графика

Для начала редактирования графика, необходимо его выделить и поместить курсор в поле ввода данных.

В нижней части графика есть пустое поле, куда нужно вставить:

- для 2D графиков – функцию от  $x$  (например,  $\sin(x)$  или  $\cos(x)$ )
- для 3D графиков – функцию от  $x$  и  $y$  (например,  $\sin(x+y)$ , но можно и  $\sin(x)$  или  $\cos(x)$ ).

В объявлении функции можно использовать любые имена переменных, но график строится только относительно « $x$ » и « $y$ ». Т.е. для графика функция должна быть вызвана с использованием этих переменных.

Если нужно построить графики сразу нескольких функции, то можно воспользоваться функцией `sys` (алгебраическая система), введя ее с клавиатуры или нажав кнопку с панели «Функции». При этом разные графики рисуются линиями разного цвета.

Для построения графика по точкам нужно использовать матрицы.

Программа позволяет выводить на графике, как надписи, так и разного вида «метки».

Формат матрицы для вывода надписей:

$$\begin{pmatrix} x1 & y1 & \text{"text"} & txtSize & \text{"txtColor"} \\ x2 & y2 & \text{"text"} & txtSize & \text{"txtColor"} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ xn & yn & \text{"text"} & txtSize & \text{"txtColor"} \end{pmatrix}$$

где  $x1...xn$  – координата  $x$  точки;

$y1...yn$  – координата  $y$  точки;

$\text{"text"}$  – надпись (пишется в кавычках);

$txtSize$  – размер шрифта;

$\text{"txtColor"}$  – цвет текста (пишется в кавычках).

Последние два столбца матрицы являются необязательными, их можно не использовать, при этом программа сама назначит цвет и размер шрифта.

Параметр  $\text{"text"}$  может содержать как текст, так и специальные символы. *Обычный текст* отображается на графике правее указанной координаты и может содержать почти любые символы. *Специальные символы* - символы, которые можно наносить на график как точки.

Программа воспримет спецсимвол только в том случае, если он записан в кавычках один и без пробелов (таблица 2).

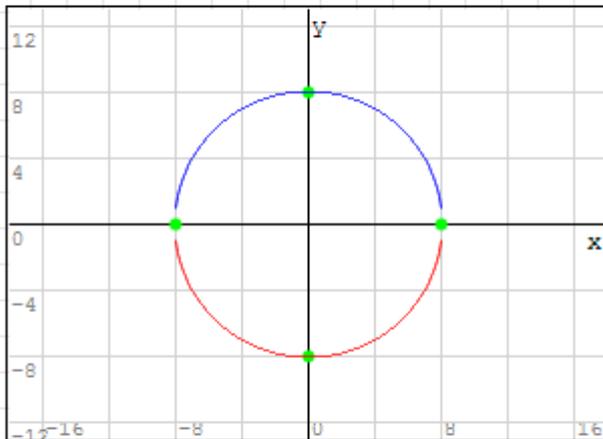
Таблица 2

Спецсимволы

Знак	Отображение на графике
+	
*	
x	
.	

График с использованием спецсимволов:

$$D := \begin{bmatrix} 8 & 0 & "." & 16 & \text{"lime"} \\ 0 & 8 & "." & 16 & \text{"lime"} \\ -8 & 0 & "." & 16 & \text{"lime"} \\ 0 & -8 & "." & 16 & \text{"lime"} \end{bmatrix}$$



$$\begin{cases} \sqrt{64 - x^2} \\ -\sqrt{64 - x^2} \end{cases} \\ D$$

Параметр `txtColor` позволяет задавать цвет надписи. Цвет задается в кавычках, некоторые возможные значения приведены ниже:

Значение	Цвет	Значение	Цвет
"aqua"		"navy"	
"black"		"olive"	
"blue"		"purple"	
"brown"		"red"	
"fuchsia"		"silver"	
"gray"		"teal"	
"green"		"violet"	
"lime"		"white"	
"maroon"		"yellow"	

## Символьные вычисления

При аналитических вычислениях результат получают в нечисловой форме в результате тождественных преобразований, среди которых более простыми есть раскрытия скобок. С помощью символьного процессора SMath Studio можно решать инженерные задачи в аналитическом виде и проводить широкий спектр аналитических преобразований, таких как, упрощение выражений и алгебраические преобразования, алгебраические и матричные операции, основные действия математического анализа, и т.д. Упрощение алгебраического выражения – это математическое преобразование, которое переводит степени и произведения в более простые соотношения. При расписании тригонометрических выражений функции кратного аргумента превращаются в функции одинарного аргумента, и т.д. SMath Studio разрешает упрощать логарифмические выражения, раскладывать на множители, приводить выражения к общему знаменателю, выносить множитель за скобки, раскладывать на элементарные дроби, выполнять подстановки и замены переменных. Символьные вычисления можно выполнять в таких вариантах

- с помощью команд меню;
- с помощью оператора символьного вычисления.

Для символьных вычислений предназначены команды меню Вычисление, которое объединяет математические операции. Для реализации второго подхода применяются все средства Smath Studio.

### Нахождение корней уравнения

Для числового поиска корней уравнения в Smath Studio используется встроенная функция solve([уравнение];[имя переменной]). Она позволяет решать уравнение вида

$$f(x) = 0,$$

где  $f(x)$  – уравнение, действительные корни которого необходимо найти;

$x$  – неизвестная.

$$\text{solve}(x^2 - 4; x) = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

В качестве первого аргумента функции `solve(...)` может быть записано уравнение без правой части (например,  $x^2$ ) или с правой частью ( $x^2 = 0$ ). Во втором случае записи вместо обычного знака « $=$ » между левой и правой частями уравнения необходимо писать знак булево равно с панели инструментов Булева:

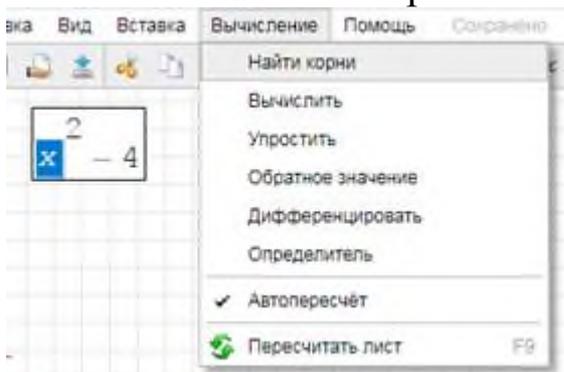
$$\text{solve}(x^2 - 4 = 0; x) = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

Использование функции `solve([уравнение];[имя переменной];[левая граница интервала];[правая граница интервала])` возвращает вектор, который имеет все корни уравнения, требует задания интервала, внутри которого ведется поиск:

$$\text{solve}(x^2 - 4; x; -3; 3) = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

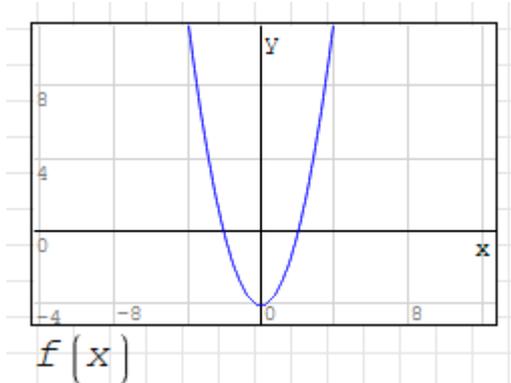
$$\text{solve}(x^2 - 4 = 0; x; -3; 3) = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

Существует возможность символьного решения уравнения. Для этого необходимо записать уравнение на листе программы, выделить переменную уравнения (например, « $x$ »), обратиться к инструменту Найти корни меню Вычисления. После этого в строке ниже появятся значения переменной:



$$\begin{array}{l} \boxed{x^2 - 4} \\ \left[ \begin{array}{c} -2 \\ 2 \end{array} \right] \end{array}$$

Можно также находить решение уравнения графически. Графическое решение заключается в определении по графику значения переменной, при которой значение функции отвечает правой части уравнения. Для этого необходимо привести уравнение к виду  $f(x) = 0$ , построить график функции  $y = f(x)$ . Координата  $x$  точки пересечения графика функции с осью  $Ox$  – искомый корень уравнения.



Для решения алгебраического уравнения, левая часть которого является полиномом  $a_0x^0 + a_1x^1 + a_2x^2 + \dots + a_nx^n = 0$  предназначена функция `polyroots`. Формат обращения к функции:

$$V := \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$$

$$X := \text{polyroots}(V)$$

Здесь  $V$  – вектор коэффициентов уравнения.

Для формирования вектора  $V$  необходимо использовать инструмент  с панели инструментов Матрицы. Далее задать количество строк – количество коэффициентов при  $x$  и число столбцов –

для вектора это «1»:

**Вставка матрицы**

---

Строки:

Столбцы:

---

Пусть задано уравнение  $x^3 - 3x + 1 = 0$ . Формируем вектор  $V$ :

$$V := \begin{bmatrix} 1 \\ -3 \\ 0 \\ 1 \end{bmatrix}$$

Определяем значения корней с помощью функции `polyroots`:

$$X := \text{polyroots}(V)$$

Выводим значения корней:

$$X = \begin{bmatrix} 0,3473 \\ 1,5321 \\ -1,8794 \end{bmatrix}$$

## Решение систем линейных алгебраических уравнений (СЛАУ)

Все методы решения систем линейных алгебраических уравнений можно разделить на две основных группы: прямые (метод Крамера, метод Гаусса и т.п.) и итеративные методы.

**Матричные способы решения СЛАУ**

В матричной форме система записывается  $A \times X = B$ . Если определитель матрицы  $A$  отличен от нуля, то система уравнений имеет единственное решение. В явном виде вектор решения:  $X = A^{-1} \times B$ .

Пусть нужно решить систему уравнений:

$$\begin{cases} x_1 + 2x_2 + x_3 = 8, \\ 2x_1 + x_2 + 2x_3 = 10, \\ 3x_1 + 3x_2 + x_3 = 12. \end{cases}$$

Решение в матричной форме:

$$A := \begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 2 \\ 3 & 3 & 1 \end{bmatrix} \quad B := \begin{bmatrix} 8 \\ 10 \\ 12 \end{bmatrix}$$

$$X := A^{-1} \cdot B = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$X_1 = 1$$

$$X_2 = 2$$

$$X_3 = 3$$

Решение СЛАУ использованием функции `roots`

Перед использованием функции `roots` необходимо все уравнения привести к виду  $f(x_1, x_2, x_3, \dots, x_n) = 0$ . Решение ищется в виде:

$$X := \text{roots} \left( \left( \begin{array}{c} f_1(x_1, x_2, x_3, \dots, x_n) \\ \vdots \\ f_m(x_1, x_2, x_3, \dots, x_n) \end{array} \right); \left( \begin{array}{c} x_1 \\ \vdots \\ x_n \end{array} \right) \right)$$

Пусть нужно решить систему уравнений:

$$\begin{cases} x_1 + 2x_2 + x_3 = 8, \\ 2x_1 + x_2 + 2x_3 = 10, \\ 3x_1 + 3x_2 + x_3 = 12. \end{cases}$$

Решение:

$$X := \text{roots} \left( \begin{bmatrix} x_1 + 2 \cdot x_2 + x_3 - 8 \\ 2 \cdot x_1 + x_2 + 2 \cdot x_3 - 10 \\ 3 \cdot x_1 + 3 \cdot x_2 + x_3 - 12 \end{bmatrix}; \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \right)$$

$$X = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

## Интегралы и производные

Для дифференцирования служит функция `diff`, которая может вернуть как символьное выражение производной, так и значение в точке. Формат записи функции:

`diff([выражение];[переменная])`

или

`diff([выражение];[переменная];[порядок]).`

Для ввода знака дифференциала можно использовать соответствующий инструмент в панели инструментов Функции .

Например,

$$\frac{d}{d x} x^2 - 3 \cdot x + \ln(x) - \sin(x)^2 = \frac{1 + (-3 + 2 \cdot (x - \cos(x) \cdot \sin(x))) \cdot x}{x}$$

$$f(x) := \frac{d}{d x} x^2 - 3 \cdot x + \ln(x) - (\sin(x))^2$$

$$f(1) = -0,9093$$

Узнать величину определенного интеграла можно с помощью функции `int([выражение];[переменная];[нижний предел];[верхний предел])`:

$$\int_1^2 x^2 d x = 2,3333$$

**Задание**

1. Решить уравнение (по вариантам) (таблица 3).

С использованием функции solve, polyroots. Построить график заданной функции, указав найденные корни.

2. Решить СЛАУ матричным методом, методом Крамера и с помощью функции root (таблица 4).

3. Найти минимум и максимум функции на заданном интервале (таблица 5), точки перегиба, подтвердить полученные результаты графически.

4. Найти площадь фигуры, ограниченной линиями (таблица 6). Результаты подтвердить графически.

5. Найти значение определенного интеграла (таблица 7).

Таблица 3

Варианты для Задания 1

вариант	уравнение	Интервал локализации корней
1	2	3
1	$x^3 - 2,92x^2 + 1,4355x + 0,791 = 0$	[1; 3]
2	$x^3 - 2,56x^2 - 1,325x + 4,395 = 0$	[-2; 3]
3	$x^3 + 2,84x^2 - 5,606x - 14,766 = 0$	[-3,5; 2,5]
4	$x^3 + 1,41x^2 - 5,472x - 7,38 = 0$	[-2,5; 2,5]
5	$x^3 + 0,85x^2 - 0,432x + 0,044 = 0$	[-1,6; 1,1]
6	$x^3 - 0,12x^2 - 1,478x + 0,192 = 0$	[-1,6; 1,6]
7	$x^3 + 0,77x^2 - 0,251x - 0,017 = 0$	[-1,6; 0,8]
8	$x^3 + 0,88x^2 - 0,3999x - 0,0376 = 0$	[-1,4; 1]
9	$x^3 + 0,78x^2 - 0,827x - 0,1467 = 0$	[-1,4; 2,5]
10	$x^3 + 2,28x^2 - 1,9347x - 3,90757 = 0$	[-2,6; 1,4]
11	$x^3 - 0,805x^2 - 0,7x + 2,77 = 0$	[-2,6; 3,2]
12	$x^3 - 0,345x^2 - 5,569x + 3,15 = 0$	[-3; 3]
13	$x^3 - 3,335x^2 - 1,679x + 8,05 = 0$	[-2; 3,4]

Таблица 3 (продолжение)

1	2	3
14	$x^3 - 2,5x^2 - 0,0099x + 0,517 = 0$	$[-1; 2,8]$
15	$x^3 - 3x^2 + 0,569x + 1,599 = 0$	$[-1,2; 3]$
16	$x^3 - 2,2x^2 + 0,82x + 0,23 = 0$	$[-2,5; 2,5]$
17	$x^3 - 5x^2 + 0,903x + 6,77 = 0$	$[-1,2; 4,6]$
18	$x^3 - 7,5x^2 + 0,499x + 4,12 = 0$	$[-1; 7,4]$
19	$x^3 - 7,8x^2 + 0,899x + 8,1 = 0$	$[-1,6; 9]$
20	$x^3 + 2x^2 - 4,9x - 3,22 = 0$	$[-3,4; 2]$
21	$x^3 + 3x^2 - 0,939x - 1,801 = 0$	$[-3,4; 1,2]$
22	$x^3 + 5,3x^2 + 0,6799x - 13,17 = 0$	$[-4,6; 3]$
23	$x^3 - 6,2x^2 - 12,999x + 11,1 = 0$	$[-2,4; 8,2]$
24	$x^3 - 0,34x^2 - 4,339x - 0,09 = 0$	$[-3,2; 2,7]$
25	$x^3 - 1,5x^2 + 0,129x + 0,07 = 0$	$[-1; 3]$
26	$x^3 - 5,5x^2 + 2,79x + 0,11 = 0$	$[-1; 3]$
27	$x^3 - 5,7x^2 - 6,219x - 2,03 = 0$	$[-1; 3]$
28	$x^3 - 3,78x^2 - 7,459x - 4,13 = 0$	$[-1; 3]$
29	$x^3 - 5x^2 - 9,9119x + 0,01 = 0$	$[-1; 3]$
30	$x^3 - 7x^2 - 1,339x - 7,55 = 0$	$[-1; 3]$

Таблица 4

## Варианты для Задания 2

вариант	задание	вариант	задание
1	2	3	4
1	$\begin{cases} 3x_1 + x_2 + 2x_3 = 4, \\ x_1 + 3x_3 = -1, \\ -x_1 + 2x_2 - x_3 = -1. \end{cases}$	16	$\begin{cases} -2x_1 + 4x_2 + x_3 = 2, \\ 2x_1 + 2x_2 - 3x_3 = 14, \\ -x_1 + 2x_2 + x_3 = 0. \end{cases}$
2	$\begin{cases} 2x_1 + 2x_2 - x_3 = 2, \\ x_1 + 2x_2 - 2x_3 = -1, \\ 3x_1 - x_2 + 2x_3 = 6. \end{cases}$	17	$\begin{cases} x_1 - 2x_2 + 2x_3 = -1, \\ 3x_1 + x_2 + 4x_3 = 11, \\ -2x_1 + x_2 = -4. \end{cases}$

Таблица 4 (продолжение)

1	2	3	4
3	$\begin{cases} 2x_1 - x_3 = -3, \\ 2x_1 - x_2 + 3x_3 = 5, \\ 3x_1 + x_2 - 2x_3 = 9. \end{cases}$	18	$\begin{cases} 2x_1 + x_2 - x_3 = -2, \\ x_1 - 2x_2 + x_3 = 1, \\ 2x_1 - x_2 + x_3 = 2. \end{cases}$
4	$\begin{cases} x_1 + 2x_2 + x_3 = -1, \\ x_1 + 5x_2 + 4x_3 = -4, \\ 2x_1 + x_2 - 2x_3 = 0. \end{cases}$	19	$\begin{cases} x_1 + 9x_2 + 6x_3 = 10, \\ x_1 + 2x_2 - 2x_3 = 2, \\ 2x_1 - x_2 + x_3 = 9. \end{cases}$
5	$\begin{cases} x_1 + 3x_3 = 4, \\ x_1 + 2x_2 - 2x_3 = 1, \\ 2x_1 - x_2 + 3x_3 = 4. \end{cases}$	20	$\begin{cases} 3x_1 + 2x_2 - 2x_3 = -11, \\ 2x_1 - x_2 = 4, \\ x_1 + 2x_2 - x_3 = -8. \end{cases}$
6	$\begin{cases} -2x_1 + 4x_2 + x_3 = 2, \\ 2x_1 + 2x_2 - 3x_3 = 14, \\ -x_1 + 2x_2 + x_3 = 0. \end{cases}$	21	$\begin{cases} 3x_1 + x_2 + 2x_3 = 4, \\ x_1 + 3x_3 = -1, \\ -x_1 + 2x_2 - x_3 = -1. \end{cases}$
7	$\begin{cases} x_1 - 2x_2 + 2x_3 = -1, \\ 3x_1 + x_2 + 4x_3 = 11, \\ -2x_1 + x_2 = -4. \end{cases}$	22	$\begin{cases} 2x_1 + 2x_2 - x_3 = 2, \\ x_1 + 2x_2 - 2x_3 = -1, \\ 4x_1 + x_2 = 5. \end{cases}$
8	$\begin{cases} 2x_1 + x_2 - x_3 = -2, \\ x_1 - 2x_2 + x_3 = 1, \\ 2x_1 - x_2 + x_3 = 2. \end{cases}$	23	$\begin{cases} 2x_1 - x_3 = -3, \\ 2x_1 - x_2 + 3x_3 = 5, \\ 3x_1 + x_2 - 2x_3 = 9. \end{cases}$
9	$\begin{cases} x_1 + 9x_2 + 6x_3 = 10, \\ x_1 + 2x_2 - 2x_3 = 2, \\ 2x_1 - x_2 + x_3 = 9. \end{cases}$	24	$\begin{cases} x_1 + 2x_2 + x_3 = -1, \\ x_1 + 5x_2 + 4x_3 = -4, \\ 2x_1 + x_2 - 2x_3 = 0. \end{cases}$
10	$\begin{cases} 3x_1 - 2x_2 - 2x_3 = -11, \\ 2x_1 - x_2 = 4, \\ x_1 + 2x_2 - x_3 = -8. \end{cases}$	25	$\begin{cases} x_1 + 3x_3 = 4, \\ x_1 + 2x_2 - 2x_3 = 1, \\ 2x_1 - x_2 + 3x_3 = 4. \end{cases}$
11	$\begin{cases} 3x_1 + x_2 + 2x_3 = 4, \\ x_1 + 3x_3 = -1, \\ -x_1 + 2x_2 - x_3 = -1. \end{cases}$	26	$\begin{cases} -2x_1 + 4x_2 + x_3 = 2, \\ 2x_1 + 2x_2 - 3x_3 = 14, \\ -x_1 + 2x_2 + x_3 = 0. \end{cases}$

Таблица 4 (окончание)

1	2	3	4
12	$\begin{cases} 3x_1 + 4x_2 - 3x_3 = 1, \\ x_1 + 2x_2 - 2x_3 = -1, \\ 3x_1 - x_2 + 2x_3 = 6. \end{cases}$	27	$\begin{cases} x_1 - 2x_2 + 2x_3 = -1, \\ 3x_1 + x_2 + 4x_3 = 11, \\ -2x_1 + x_2 = -4. \end{cases}$
13	$\begin{cases} 2x_2 - x_3 = -3, \\ 2x_1 - x_2 + 3x_3 = 5, \\ 3x_1 + x_2 - 2x_3 = 9. \end{cases}$	28	$\begin{cases} 2x_1 + x_2 - x_3 = -2, \\ x_1 - 2x_2 + x_3 = 1, \\ 2x_1 - x_2 + x_3 = 2. \end{cases}$
14	$\begin{cases} x_1 + 2x_2 + x_3 = -1, \\ x_1 + 5x_2 + 4x_3 = -4, \\ 2x_1 + x_2 - 2x_3 = 0. \end{cases}$	29	$\begin{cases} x_1 + 9x_2 + 6x_3 = 10, \\ x_1 + 2x_2 - 2x_3 = 2, \\ 2x_1 - x_2 + x_3 = 9. \end{cases}$
15	$\begin{cases} x_1 + 3x_3 = 4, \\ x_1 + 2x_2 - 2x_3 = 1, \\ 2x_1 - x_2 + 3x_3 = 4. \end{cases}$	30	$\begin{cases} 3x_1 + 2x_2 - 2x_3 = -11, \\ 2x_1 - x_2 = 4, \\ x_1 + 2x_2 - x_3 = -8. \end{cases}$

Таблица 5

## Варианты для Задания 3

вариант	уравнение	Интервал исследования
1	2	3
1	$y = \frac{x^3 + 4}{x^2}$	[0,5; 5]
2	$y = \frac{x^2 - x + 1}{x - 1}$	[1,2; 5]
3	$y = \frac{2}{x^2 + 2x}$	[-1,5; -0,5]
4	$y = \frac{4x^2}{3 + x^2}$	[-2,5; 2,5]
5	$y = \frac{12x}{9 + x^2}$	[-6; 6]
6	$y = \frac{x^2 - 3x + 3}{x - 1}$	[1,1; 3,1]

Таблица 5 (продолжение)

1	2	3
7	$y = \frac{-x^3 + 4}{x^2}$	$[-4; -0,2]$
8	$y = \frac{x^2 - 4x + 1}{x - 4}$	$[1,2; 3,8]$
9	$y = \frac{2x^3 + 1}{x^2}$	$[0,2; 2,5]$
10	$y = \frac{(x-1)^2}{x^2}$	$[0,5; 5,4]$
11	$y = \frac{x^2}{(x-1)^2}$	$[-4; 0,7]$
12	$y = \left(1 + \frac{1}{x}\right)^2$	$[-3; -0,3]$
13	$y = \frac{12 - 3x^2}{x^2 + 12}$	$[-3; 3]$
14	$y = \frac{9 + 6x - 3x^2}{x^2 - 2x + 13}$	$[-3; 3]$
15	$y = \frac{3x}{x^2 + 4}$	$[-5; 4]$
16	$y = \left(\frac{x-1}{x+1}\right)^2$	$[-0,5; 4]$
17	$y = \frac{3x^4 + 1}{x^3}$	$[0,5; 3]$
18	$y = \frac{4x}{(x+1)^2}$	$[-0,5; 5]$
19	$y = \frac{4(x-1)}{(x+1)^2}$	$[0,5; 6]$
20	$y = \frac{1 - 2x^3}{x^2}$	$[-2; -0,52]$

Таблица 5 (окончание)

1	2	3
21	$y = \frac{4}{x^2 + 2x - 3}$	$[-2,5; 0,5]$
22	$y = \frac{4}{3 + 2x - x^2}$	$[-0,5; 2,5]$
23	$y = \frac{-7 + 2x + x^2}{x^2 + 2x - 3}$	$[-2,5; 0,5]$
24	$y = \frac{1}{x^4 - 1}$	$[-0,8; 0,8]$
25	$y = -\left(\frac{x}{x+2}\right)^2$	$[-1; 3]$
26	$y = \frac{x^3 - 32}{x^2}$	$[-20; -0,3]$
27	$y = \frac{4(x+1)^2}{x^2 + 2x + 4}$	$[-3; 0]$
28	$y = \frac{3x - 2}{x^3}$	$[0,5; 3]$
29	$y = \frac{x^2 - 6x + 9}{(x-1)^2}$	$[1,5; 5]$
30	$y = \frac{x^3 - 27x + 54}{x^3}$	$[-6; -2]$

Таблица 6

## Варианты для Задания 4

вариант	задание	вариант	задание
1	2	3	4
1	$y = -x^2 + 4; y = 0$	16	$x = 4 - y^2; y = x - 2$
2	$x = -y^2 + 3; x = 0$	17	$y = 6x - x^2; y = 0$
3	$xy = 4; x = 1; x = 4; y = 0$	18	$xy = 7; y = 8 - x$

Таблица 6 (продолжение)

1	2	3	4
4	$y = x^2 + 2; x = -1; x = 2;$ $y = 0$	19	$y^2 = 4x; x = 6;$ $y = 0$
5	$y = \sin x; x = 0; x = \pi; y = 0$	20	$y^2 = x + 5; x = 0$
6	$y = x^2 + 2; y = 4 - x;$ $y = x + 4$	21	$xy = 6; y = x - 1;$ $x = 6; y = 0$
7	$y^2 = 2x + 1; x - y - 1 = 0$	22	$xy = 4; y = 5 - x$
8	$xy = 5; x = 2; x = 5; y = 0$	23	$y = x^2 + 1; y = 5$
9	$y = x^2 + 2; y = 4 - x;$ $y = x + 4; y = 0$	24	$x = 4 - y^2; y = 2 - x$
10	$y = 4 - x^2; y = x + 2$	25	$y = x^3; x = 2; y = 0$
11	$x = y^2 - 4; y = -x - 2$	26	$y = x^2; y = 2 - x$
12	$y = \sqrt{x + 1}; x = 0; x = 3;$ $y = 0$	27	$y = \cos x; x = -\frac{\pi}{2};$ $x = \frac{\pi}{2}; y = 0$
13	$y = x^2; y = -x^2 + 2$	28	$x = y^2; x = 4; y = 0$
14	$y^2 = 2x + 4; x = 0$	29	$y = x^2 - 3; x = 4$
15	$y^2 = x^3; y = 8; x = 0$	30	$xy = 6; y = 7 - x$

## Варианты для Задания 5

вариант	задание	вариант	задание
1	2	3	4
1	$\int_0^1 (3x^2 - 2x + 1) dx$	16	$\int_{\frac{\pi}{8}}^{\frac{\pi}{6}} \frac{dx}{\cos^2 2x}$
2	$\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} (\cos x - 1) dx$	17	$\int_0^1 \frac{dx}{\sqrt[3]{(3x+8)^2}}$
3	$\int_1^9 3(\sqrt{x} - x) dx$	18	$\int_0^1 \sqrt{1+xdx}$
4	$\int_1^2 \left(x + \frac{1}{x}\right)^2 dx$	19	$\int_4^8 \frac{dx}{\sqrt{8x}}$
5	$\int_4^9 \sqrt{x}(1 + \sqrt{x}) dx$	20	$\int_2^{13} \frac{dx}{\sqrt[5]{(3-x)^4}}$
6	$\int_1^2 (\sqrt{x} + \sqrt[3]{x}) dx$	21	$\int_0^{\frac{\pi}{3}} \sin 3x dx$
7	$\int_0^3 (1 + e^x) dx$	22	$\int_0^3 e^{\frac{x}{3}} dx$
8	$\int_1^8 \frac{dx}{\sqrt[3]{x^2}}$	23	$\int_0^1 \frac{dx}{\sqrt{4-x^2}}$
9	$\int_1^2 \left(x^2 + \frac{1}{x^4}\right) dx$	24	$\int_1^{\sqrt{2}} \sqrt{2-x^2} dx$
10	$\int_0^2 (x^2 - 2x) dx$	25	$\int_0^{\frac{\pi}{4}} \sin 4x dx$

Таблица 7 (продолжение)

1	2	3	4
11	$\int_1^8 \left( \sqrt[3]{x^2} + \frac{1}{\sqrt[3]{x^2}} \right) dx$	26	$\int_0^1 \frac{dx}{4x+1}$
12	$\int_0^4 (1 + \sqrt{x})^2 dx$	27	$\int_{\frac{\pi}{9}}^{\frac{\pi}{6}} \frac{dx}{\sin^2 3x}$
13	$\int_1^4 \frac{dx}{x^2}$	28	$\int_0^1 \frac{dx}{\sqrt{(3x+1)^3}}$
14	$\int_{-4}^1 \frac{dx}{x^3}$	29	$\int_3^8 \sqrt{1+xdx}$
15	$\int_0^{\pi} (\sin x + 3) dx$	30	$\int_0^4 \left( e^{-\frac{x}{3}} - 2e^{2x} \right) dx$

Составить отчет по результатам выполнения лабораторной работы. Отчет должен содержать:

- титульный лист (Приложение А);
- Содержание;
- Цель работы;
- Задание;
- Скриншот листа выполнения задания с отображением формул и с отображением результатов вычислений;
- Выводы по работе.

Номера страниц проставляются внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первой страницей является титульный лист, нумерация проставляется, начиная с Цели работы отчета.

Макет Оглавления приведен в Приложении Б.

Для оформления заголовков разделов следует использовать стиль «Заголовок 1». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;

- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 12 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Для оформления текста отчета необходимо использовать стиль «Обычный». Параметры стиля:

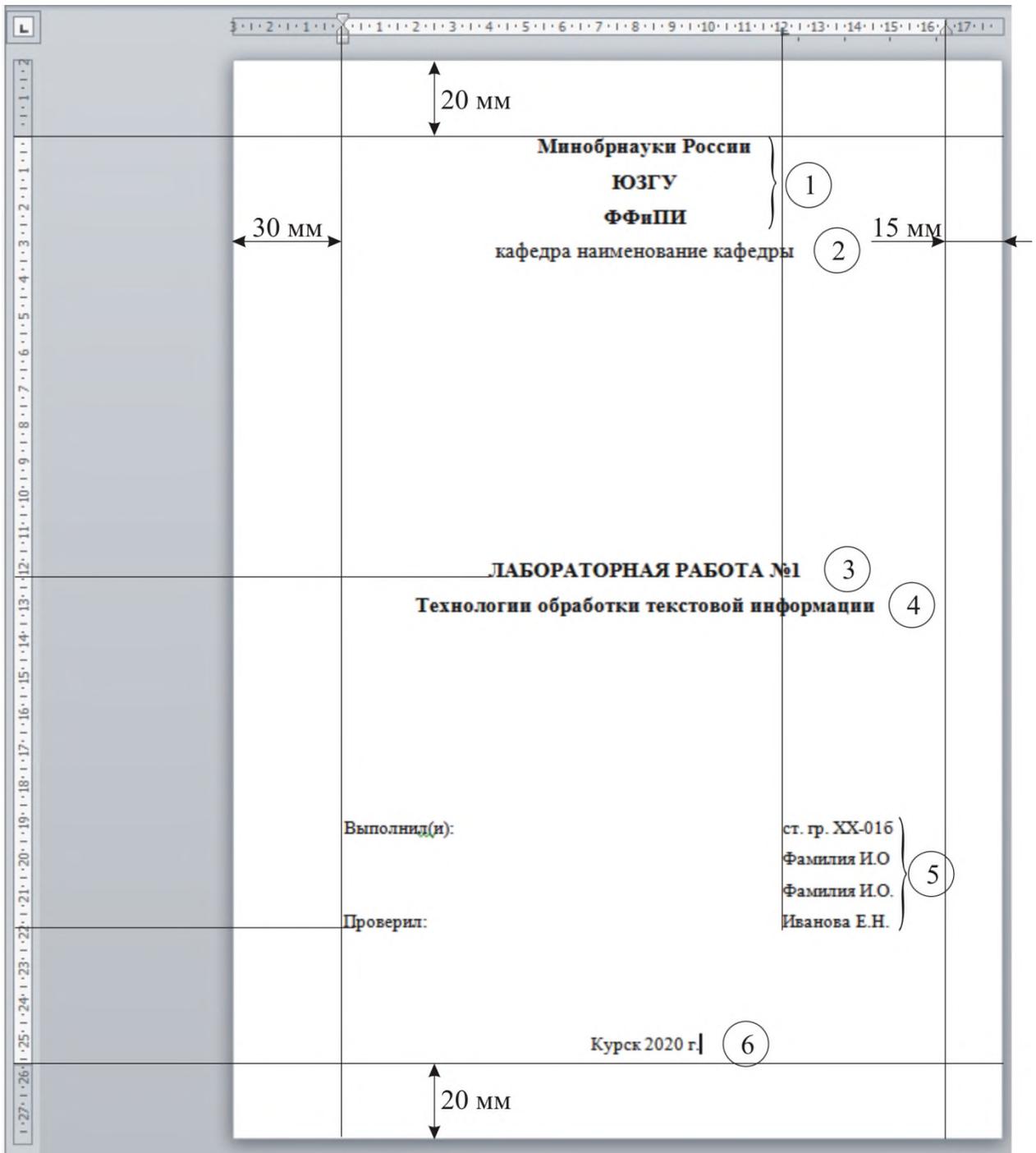
- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

### **Список использованных источников**

1. Обзор и краткое введение [Электронный ресурс] : сайт – Режим доступа: URL: <https://sites.google.com/site/mikkhalichlab/rukovodstvo/s-sego-nacat/nacalo>

2. Аверкин, С. SMath Studio. Краткое руководство [Электронный ресурс] – Режим доступа: URL: <https://en.smath.com/view/c1be00e3-eb8c-78a5-b1f9-f6e15457ecbc/summary>. SMathStudio.0\_85.Article.pdf

## Оформление титульного листа отчета по лабораторной работе



Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

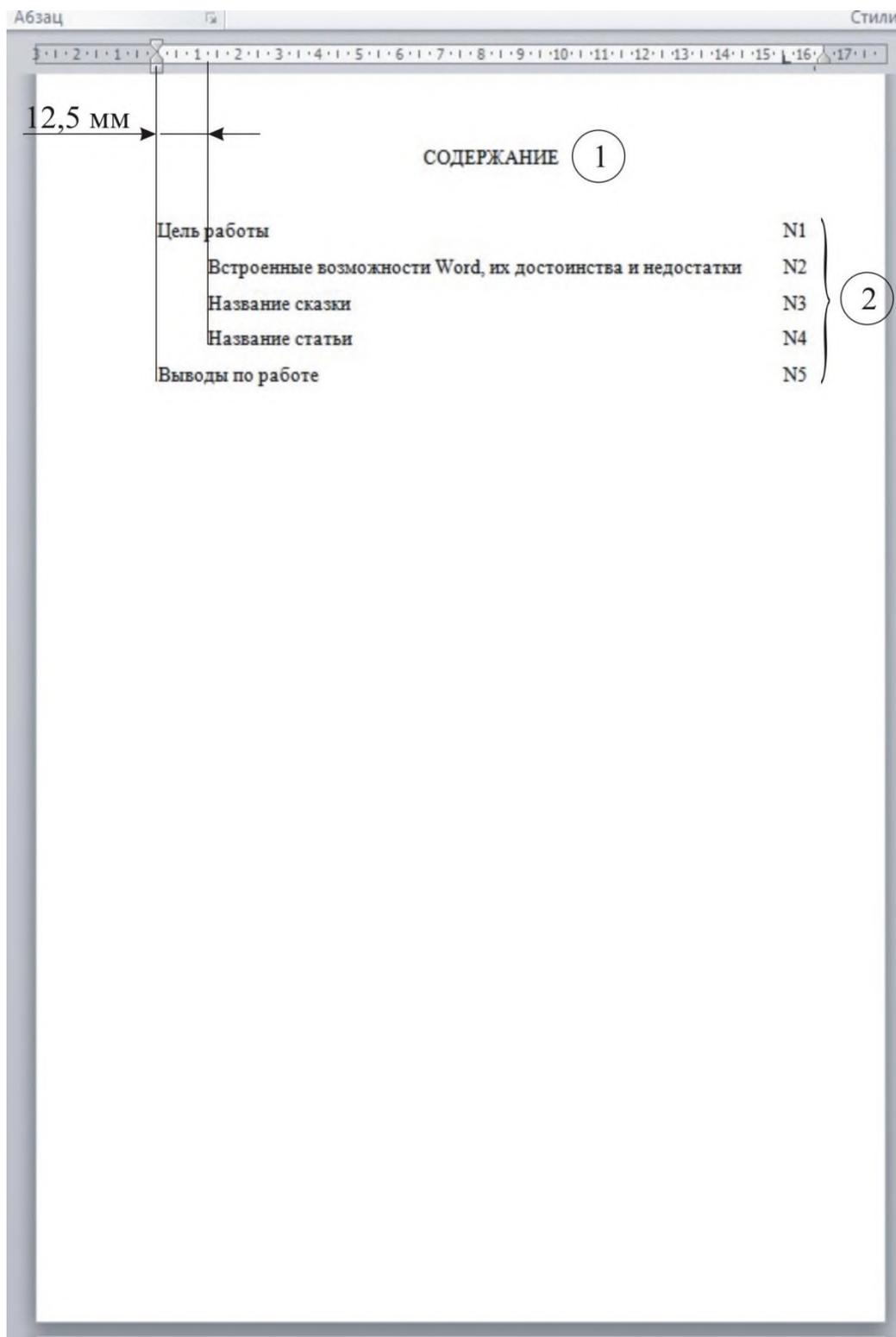
2: Шрифт Times New Roman, размер 16 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

3: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

4: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

5: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по ширине, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки, позиция табуляции 12 см.

6: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

**Оформление Содержания отчета по лабораторной работе**

Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по левому краю, отступы слева 0, справа 0, интервалы перед 0, после 0, междустрочный 1,5 строки.

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 23.12.2021 11:40:58

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d79e5f1c11eabbf73e943df4a4851fda56d089

## МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра вычислительной техники



### Технологии представления информации в графическом виде

Методические указания  
по выполнению практической работы по дисциплине  
«Информатика» для студентов специальности  
10.05.02 Информационная безопасность  
телекоммуникационных систем

Курск 2021 г.

УДК 681.3

Составители: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина

Рецензент

Доцент кафедры программной инженерии,  
кандидат технических наук

*Ю.А. Халин*

**Технологии представления информации в графическом виде:** методические указания по выполнению практической работы / Юго-Зап. гос. ун-т; сост.: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина. – Курск, 2021. – 18 с.: табл. 1. – Библиограф.: с. 14.

Рассматриваются приемы использования средств графического онлайн сервиса для создания граф-схем алгоритмов. Теоретический материал сопровождается примерами.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.  
Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## Цель работы

Получить знания о возможностях графического сервиса; научиться анализировать возможности средств графического сервиса при решении прикладных задач; приобрести навык использования различных инструментов пользовательского интерфейса графического сервиса.

## Краткая теоретическая информация

### Интерфейс сервиса

При переходе на страницу он-лайн графического сервиса draw.io (<https://app.diagrams.net/>) открывается страница с окном, представленным на рисунок 1, где нужно выбрать один из двух вариантов: создание новой диаграммы (Create New Diagram) или открытие существующей (Open Existing Diagram).

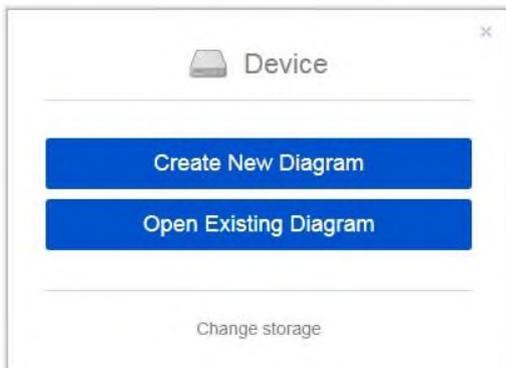


Рисунок 1 – Окно выбора диаграммы

При создании новой диаграммы появляется следующее окно, где нужно выбрать тип создаваемой диаграммы (рисунок 2). Так как наша задача – создание граф-схемы алгоритма, то следует щелкнуть на вариант из категории Basic – Flowchart (Общие – Блок-схема) (рисунок 2). Для подтверждения своего выбора нажмите на кнопку Create (внизу окна). Следующее окно – стандартное диалоговое окно сохранения пока пустого файла draw.io, после закрытия которого появляется интерфейс графического сервиса (рисунок 3).

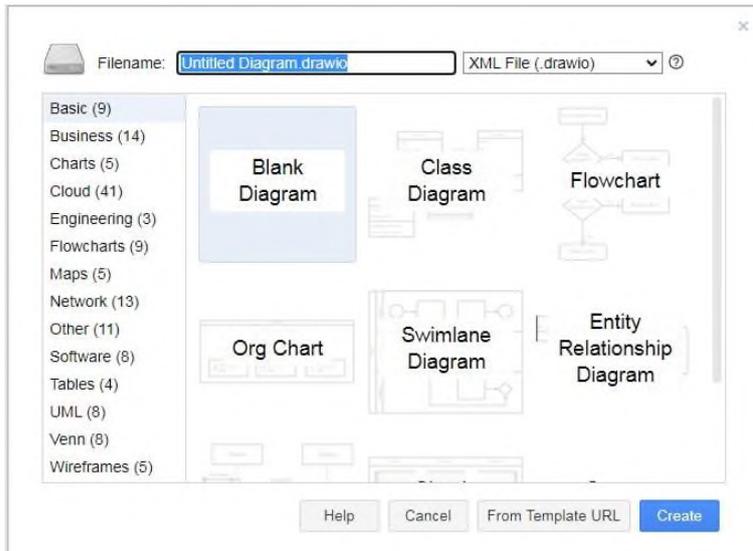


Рисунок 2 – Выбор типа диаграммы

Интерфейс программы состоит из области выбора графических шаблонов, там же выбираются фигуры, области панелей инструментов и рабочего поля (рисунок 3).

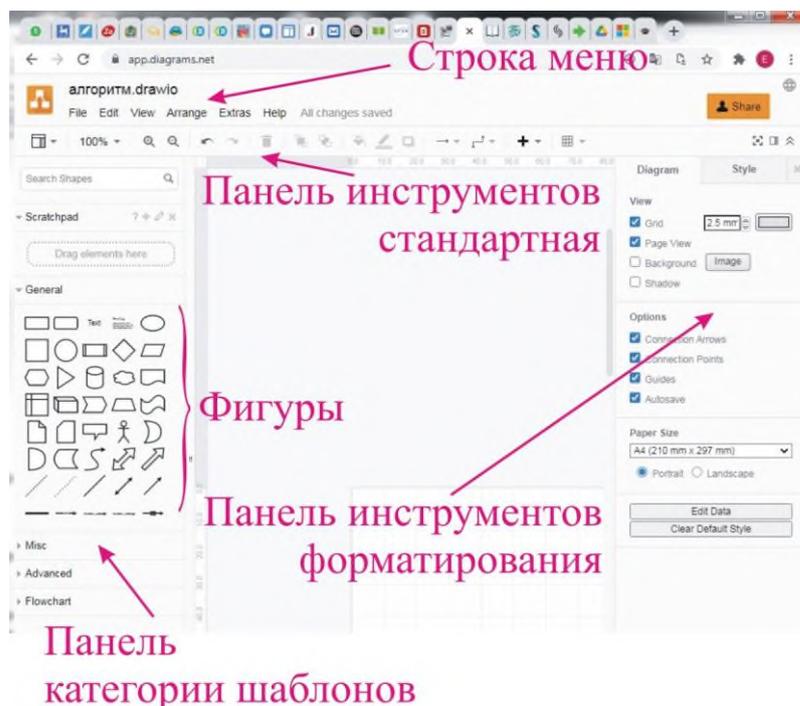


Рисунок 3 – Интерфейс графического сервиса draw.io

Рабочее поле занимает центральную часть, отображается в виде бумажной страницы с нанесенной на нее сеткой, которая облегчает размещение фигур. Горизонтальная и вертикальная линейки

помогают позиционировать фигуры и показывают размер страницы документа. Убедитесь, что в меню View (Вид) включены опции Ruler (Линейка), Grid (Сетка), Guides (Направляющие) и Connection Points (Точки соединения).

Draw.io предлагает несколько способов изменения масштаба изображения.

Первый способ.

В строке меню необходимо открыть меню View (Вид) → Page Scale... (Масштаб страницы). В результате откроется окно (рисунок 4), в котором задается масштаб отображения изображения.

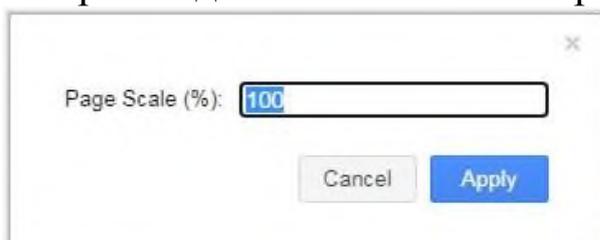


Рисунок 4 – Окно задания масштаба изображения

Второй способ.

Нажимая клавишу Ctrl и одновременно прокручивая колесо на мыши, можно быстро увеличивать или уменьшать масштаб отображения изображения. При изменении масштаба отображения документа на экране изменяются деления на вертикальной и горизонтальной линейках. При этом они все равно отображают действительные величины.

## **Формирование изображения схемы**

Draw.io включает множество фигур, которые предназначены для быстрого создания схем, диаграмм. Фигуры – это строительные блоки всех схем в draw.io. все фигуры объединены по темам в шаблоны.

Для эффективной работы с любым типом фигур в каждой схеме необходимо понимать главные принципы их поведения, различия между типами фигур и то, как взаимодействовать с фигурами draw.io.

Одним из основных видов взаимодействия с фигурами является их перетаскивание из трафарета на страницу диаграммы или схемы с целью добавления этой фигуры в изображения. Другим ви-

дом является предварительное выделение фигуры перед каким-либо действием с ней. Для этого необходимо поместить указатель поверх той фигуры, которую хотите выделить, и, когда под указателем появится четырехнаправленная стрелка, щелкнуть на фигуре. Определить, выделена фигура или нет, можно по наличию у нее манипуляторов выделения (жирных точек голубого цвета). Чтобы снять выделение, щелкните на пустой области страницы документа. Чтобы удалить фигуру со страницы документа, ее нужно выделить и нажать клавишу Delete.

При выделении фигуры вокруг нее появляется прямоугольная область выделения. Эта область всегда имеет восемь манипуляторов выделения: по одному для каждого из четырех углов и по одному на каждой из сторон поля выделения. У правого верхнего угла выделения появляется один манипулятор вращения (рисунок 5).



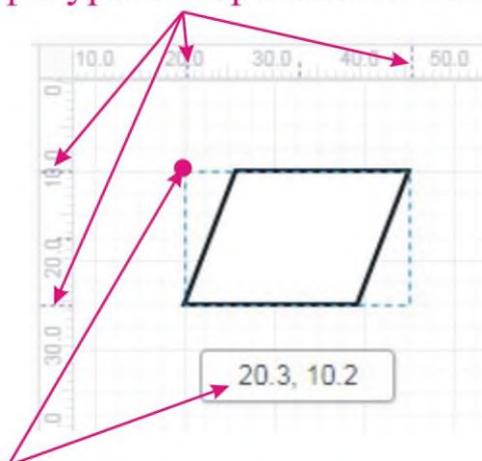
Рисунок 5 – Область выделения фигуры с манипуляторами

С помощью манипуляторов можно изменять размеры фигуры и ее угол поворота. Точно размер фигуры можно задать на панели инструментов форматирования. Для этого необходимо выделить фигуру, на вкладке Arrange (Формат). Здесь в разделе Size (Размер) в поле Width (Ширина) задается ширина, в поле Height (Высота) – высота фигуры. Единицы изменения задаются в разделе Units (Единицы измерения) меню View (Вид). Выберите вариант Millimeters (Миллиметры). Устанавливая размеры фигуры, убедитесь, что отсутствует флажок Constrain Proportions (Сохраняя пропорции).

Draw.io определяет положение фигуры на странице документа, используя горизонтальную и вертикальную линейки, и отображает это положение в прямоугольнике, ниже фигуры (рисунок 6). Точно указать местоположение фигуры на листе можно на вкладке Arrange (Формат) панели инструментов форматирования. Здесь в разделе Position (Положение) в поле Left (Левый угол) задается расстояние от левого края листа до левого верхнего угла выделения

фигуры, в поле Top (Верхний угол) – расстояние от верхнего края листа до левого верхнего угла выделения фигуры.

Координаты текущего положения фигуры отображаются на Линейке



Координаты позиционирования (положение верхней левой точки области выделения фигуры)

Рисунок 6 – Отображение фигуры на линейках при позиционировании

Чтобы перенести фигуры в другое место, необходимо их выделить. Выделять можно по-отдельности при нажатой клавише Shift, или рамкой протягиванием мыши.

Все элементы схемы должны быть соединены. Для этого используются соединительные линии. Draw.io позволяет легко соединять фигуры блок-схем с помощью Connection (соединительных линий). При перемещении соединенных фигур коннекторы остаются присоединенными к перемещаемым фигурам. Чтобы разбить эту связь, нужно переместить концевую точку коннектора или удалить его совсем.

Для соединения фигур можно использовать два способа.

Способ первый.

Сначала поместить все необходимые фигуры блок-схемы на страницу документа. Затем на стандартной панели инструментов



выбрать Waypoints (Соединительная линия). Указатель подвести к точке соединения, при этом она окрашивается в зеленый цвет и, не отпуская левую кнопку мыши, протянуть к точке соединения следующей фигуры. И когда она выделится зеленым цветом, отпустить кнопку мыши (рисунок 7).

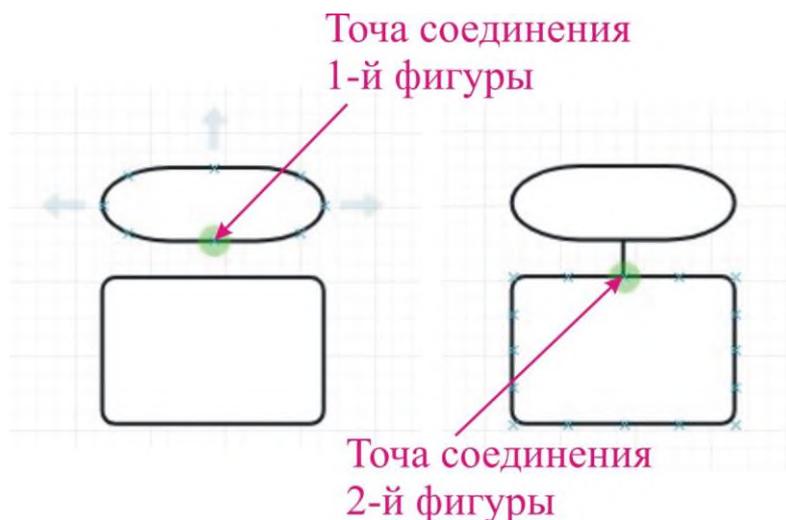


Рисунок 7 – Соединение фигур на схеме (способ первый)

Способ второй.

Сначала поместить фигуру блок-схемы на страницу документа. Затем выделить ее, навести указатель мыши на стрелку, идущую от манипулятора выделения, расположенного на нужной стороне фигуры. При этом стрелка окрашивается в яркий цвет и, не отпуская левую кнопку мыши, протянуть к точке соединения следующей фигуры. И когда она выделится зеленым цветом, отпустить кнопку мыши (рисунок 8).

Для изменения толщины линии нужно выбрать линию, а затем на вкладке Style (Стиль) панели инструментов форматирования задать требуемую ширину. На этой же вкладке можно задать формы начала и конца линии.

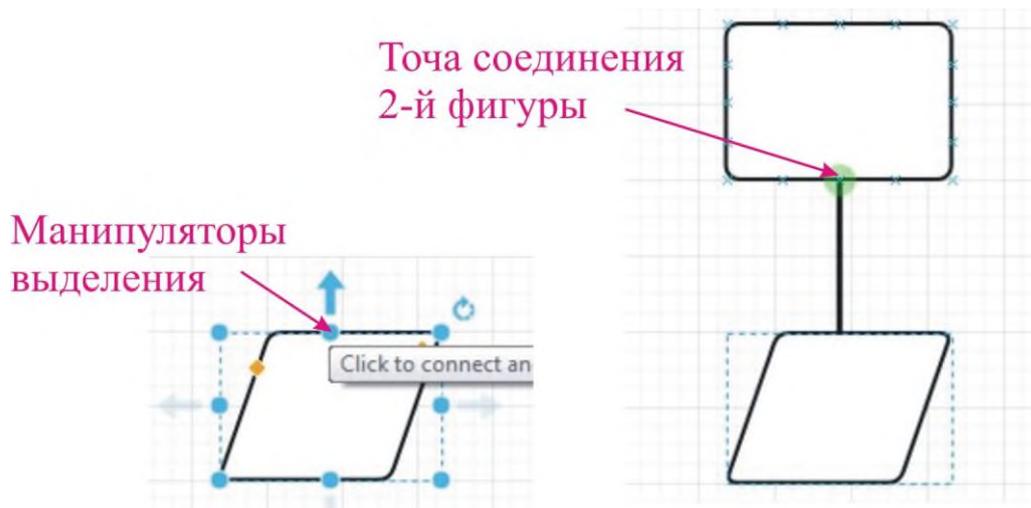


Рисунок 8 – Соединение фигур на схеме (способ второй)

### Добавление текста в фигуры

Чтобы добавить в фигуру текст или изменить его, необходимо дважды щелкнуть левой кнопкой мыши на внутреннем поле фигуры и можно вводить текст. После набора текста его можно отформатировать, используя вкладку Text (Текст) панели инструментов форматирования.

Область фигуры, в которой отображается текст, называется текстовым блоком. При вводе текста в точке ввода появляется мигающий курсор, который показывает место вставки вводимого текста.

Для создания текста, который не связан с какой-либо фигурой, используется текстовая фигура, содержащая только текст. Текстовая фигура – это прямоугольная фигура без границ вокруг нее и цветовой заливки внутри. Для вставки текстовой фигуры необходимо нажать на инструмент Insert (Вставка) стандартной панели инструментов (рисунок 9).

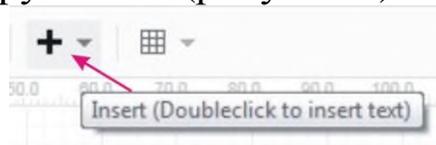


Рисунок 9 – Инструмент Insert (Вставка) стандартной панели инструментов

Из раскрывшегося списка выбрать Text (Текст). После на странице документа появится область с текстовым курсором (рисунок 10).

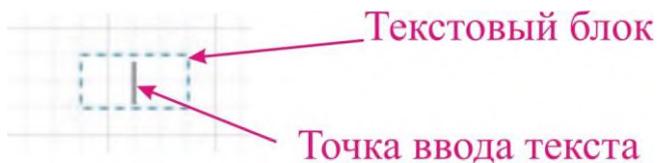


Рисунок 10 – Текстовый блок

Чтобы придать тексту требуемый вид необходимо воспользоваться вкладкой Text (Текст) панели инструментов форматирования. На ней можно выбрать и нужный шрифт, и размер.

К коннектору можно добавлять текст, как и к любой другой фигуре. Необходимо дважды щелкнуть по коннектору и в раскрывшемся текстовом блоке набрать необходимый текст.

Внутри элементов блок-схем помещают краткие записи и обозначения, служащие для понимания выполняемых ими функций. Текст более подробного описания и пояснения функций помещают в специальной фигуре Annotation (Примечание). Чтобы добавить в фигуру текст, нужно выделить ее и начать вводить нужный текст. Линия соединения примечания с фигурой блок-схемы должна быть горизонтальной.

### **Основные правила представления граф-схем**

В таблице 1 представлены основные блоки для составления алгоритмов с указанием размеров. Размер а должен выбираться из ряда 10, 15, 20 мм. Допускается увеличивать размер а на число, кратное 5 мм. Размер b равен  $1,5 \cdot a$ .

Основным направлением потока в схемах алгоритмов принято направление сверху-вниз, слева-направо. Если линии потока идут в основном направлении и не имеют изломов, стрелками их не обозначают. В остальных случаях направление линии потока следует обязательно обозначать стрелкой.

Записи внутри фигур должны быть представлены так, чтобы их можно было читать слева направо и сверху вниз, независимо от направления потока.

В случае необходимости слияния линий потока место слияния должно быть обозначено точкой или символом «0».

Таблица 1

## Основные блоки для составления блок-схем алгоритмов

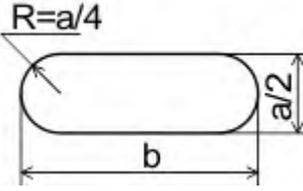
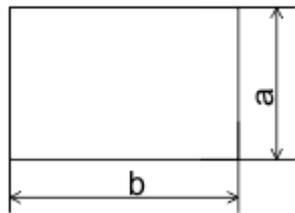
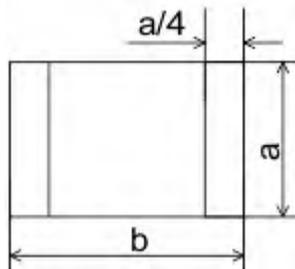
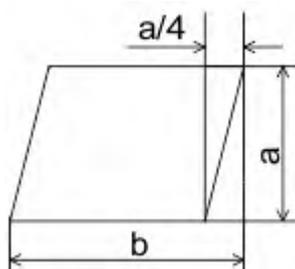
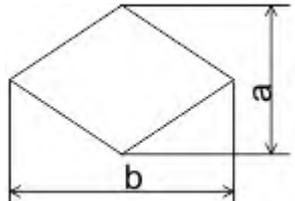
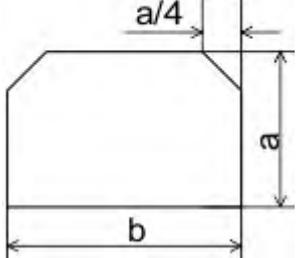
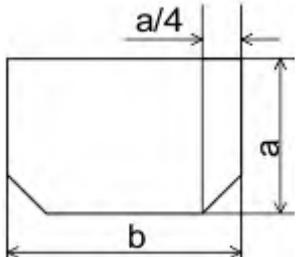
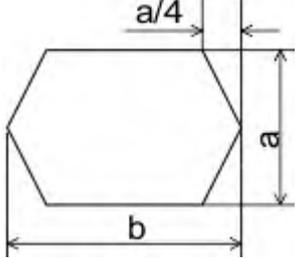
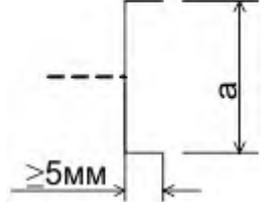
Название	Обозначение	Описание
1	2	3
Терминатор		Начало, конец, прерывание процесса обработки данных или выполнения программы
Процесс		Выполнение операции или группы операций, в результате которых изменяется значение, форма представления или расположение данных
Предопределенный процесс		Использование ранее созданных и отдельно описанных алгоритмов или программ
Ввод-вывод		Преобразование данных в форму, пригодную для обработки (ввод) или отображения результатов обработки (вывод)
Решение		Выбор направления выполнения алгоритма или программы в зависимости от некоторых переменных условий. Блок решения имеет 1 вход и по крайней мере 2 выхода

Таблица 1 (продолжение)

1	2	3
Границы цикла	<p>Начало цикла</p>  <p>Конец цикла</p> 	<p>Символ, состоящий из двух частей, отображает начало и конец цикла. Обе части символа имеют один идентификатор.</p> <p>Условия для инициализации, приращения, завершения и т.д. помещаются внутри символа в начале или в конце в зависимости от расположения операции, проверяющей условие.</p>
Подготовка		<p>Выполнение операций, меняющих команды или группу команд, с целью воздействия на некоторую последующую функцию (установка переключателя, модификация регистра, инициализация программы)</p>
Комментарий		<p>Пояснение к элементу схемы (или линии связи)</p>
Соединитель		<p>При большой насыщенности схемы отдельные линии потока между удаленными символами допускается обрывать. При этом в конце (начале) обрыва должен быть помещен символ «Соединитель». Внутри блока соединителя указывается имя уникального идентификатора.</p>

### Задание

Изобразить средствами draw.io блок-схемы заданного алгоритма в соответствии с правилами.

Составить отчет по результатам выполнения лабораторной работы. Отчет должен содержать:

- титульный лист (Приложение А);
- Содержание;
- Цель работы;
- Задание;
- Скриншот листа draw.io с полученной граф-схемой;
- Выводы по работе.

Номера страниц проставляются внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первой страницей является титульный лист, нумерация проставляется, начиная с Цели работы отчета.

Макет Оглавления приведен в Приложении Б.

Для оформления заголовков разделов следует использовать стиль «Заголовок 1». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 12 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Для оформления текста отчета необходимо использовать стиль «Обычный». Параметры стиля:

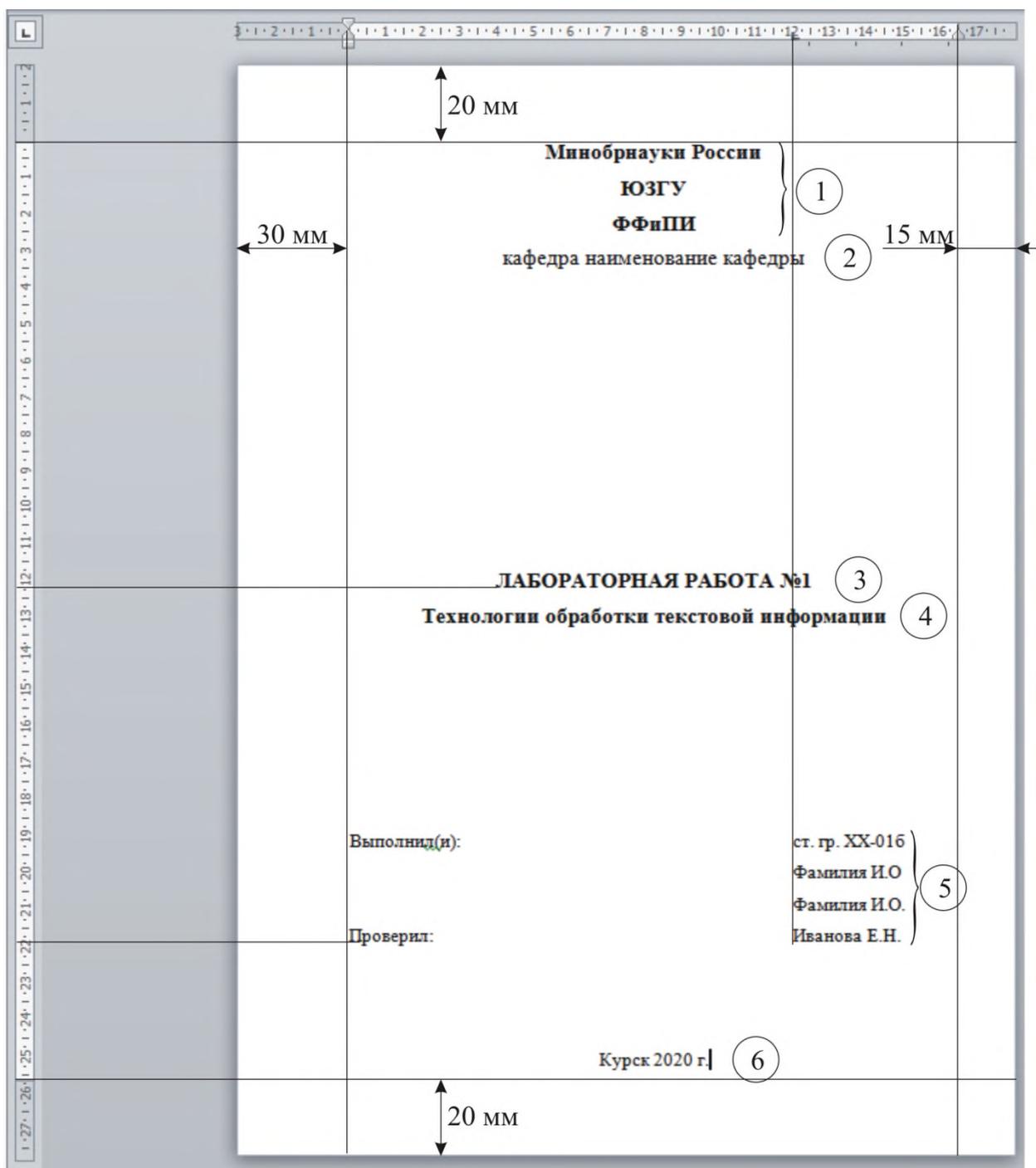
- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

**Список использованных источников**

1. Единая система программной документации. Схемы алгоритмов и программ. Обозначения условные графические ГОСТ 19.003-80. – М.: Государственный комитет СССР по стандартам, 1982. – С. 16-27.

2. Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения ГОСТ 19.701-90 (ИСО 5807-85). – М.: Стандартинформ, 2010. – С. 137-158.

## Оформление титульного листа отчета по лабораторной работе



Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 16 пт, начертание обыч-

ное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

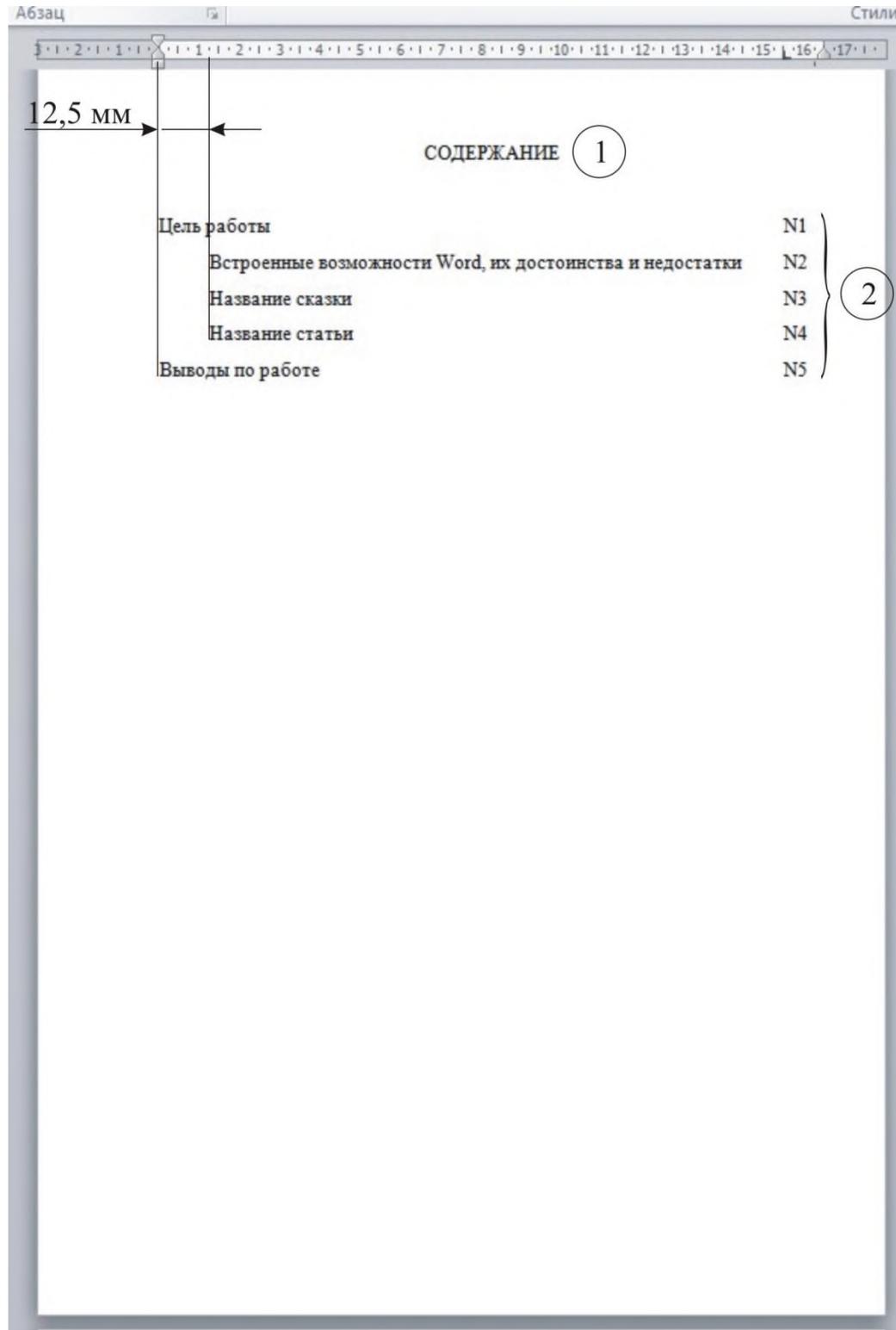
3: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

4: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

5: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по ширине, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки, позиция табуляции 12 см.

6: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

## Оформление Содержания отчета по лабораторной работе



Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по левому краю, отступы слева 0, справа 0, интервалы перед 0, после 0, междустрочный 1,5 строки.

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 23.12.2021 11:40:58

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d79e5f1c11eabbf73e943df4a4851fda56d089

## **МИНОБРАЗОВАНИЯ РОССИИ**

**Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)**

**Кафедра вычислительной техники**



### **Кодирование информации**

**Методические указания  
по выполнению практической работы по дисциплине  
«Информатика» для студентов специальности  
10.05.02 Информационная безопасность  
телекоммуникационных систем**

Курск 2021 г.

УДК 681.3

Составители: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина

Рецензент

Доцент кафедры программной инженерии,  
кандидат технических наук

*Ю.А. Халин*

**Кодирование информации:** методические указания по выполнению практической работы / Юго-Зап. гос. ун-т; сост.: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина. – Курск, 2021. – 22 с.: табл. 5. – Библиограф.: с. 16.

Рассматриваются алгоритмы кодирования информации. Приводятся приемы использования встроенных средств табличного процессора для кодирования/декодирования текстовой информации. Теоретический материал сопровождается примерами.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.  
Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## **Цель работы**

Получить знания о методах и алгоритмах кодирования информации; научиться анализировать возможности средств табличного процессора для реализации различных методов кодирования; приобрести навык использования различных инструментов табличного процессора для выполнения кодирования/декодирования информации

## **Краткая теоретическая информация**

### **Информация**

Информация есть информация –  
не материя и не энергия.

*Норберт Винер*

Развитие человеческого общества в целом связано с накоплением и обменом информации. Тысячи лет люди в процессе накопления и распространения информации использовали только свою память и речь. Появление телеграфа, телефона, радио, телевидения позволило передавать огромные потоки информации.

Таким образом информацию можно передавать, воспринимать, использовать, запоминать, принимать, кроме того создавать, копировать, формализовать, распространять, преобразовывать, комбинировать, обрабатывать, делить на части, упрощать, собирать, хранить, искать, измерять, разрушать, и др.

Все эти процессы, связанные с определенными операциями над информацией, называются информационными процессами.

Информация обладает следующими свойствами, характеризующими ее качественные признаки: достоверность, полнота, ценность, своевременность, понятность, доступность, краткость и др.

Для того чтобы человек мог быстро ориентироваться в огромном потоке информации, ему необходимы надёжные и неутомимые помощники, которые должны накапливать её, сортировать и перерабатывать в соответствии с запросами потребителя, выдавать ему нужную информацию в систематизированном и удобном для использования виде. Компьютеризация дает людям быстрый доступ к источникам информации, избавляет их от рутинной работы, обес-

печивает высокий уровень автоматизации обработки информации в производственной и социальной сферах.

Информация – фундаментальное понятие, поэтому определить его исчерпывающим образом через какие-то более простые понятия невозможно. Можно лишь, обращаясь к различным аспектам этого понятия, пояснить его смысл. В широком смысле информация – это отражение реального мира, выражаемое в виде сигналов и знаков. Сигналы отражают физические характеристики различных процессов и объектов.

В интуитивном, житейском смысле под информацией мы понимаем совокупность интересующих нас сведений, знаний и т.д. При этом подразумевается, что существует источник информации и её потребитель. От источника к потребителю информация передается с помощью сообщения. Сообщение может содержать слова, выражения или знаки, смысл и значение которых заранее известны адресату.

Обычно информация хранится, передается, обрабатывается в символической (знаковой) форме. Алфавитный подход к измерению информации позволяет определить количество информации в тексте и является объективным, т.е. не зависит от субъекта (человека), воспринимающего текст.

Одна и та же информация может быть представлена в разной форме, с помощью различных знаковых систем. Язык – это определенная знаковая система представления информации. Существуют естественные (разговорные) языки и формальные языки (нотная грамота, язык математики, язык мимики и жестов, дорожные знаки и т.д.).

### **Кодирование информации**

Конечный упорядоченный набор знаков, используемых для передачи информации, называется алфавитом. Последовательность символов алфавита – словом. Сообщение, как правило, представляет собой последовательность слов. Довольно часто передаваемая информация кодируется. Кодирование информации – переход от одной формы представления информации к другой, более удобной для хранения, обработки или передачи.

Способ кодирования зависит от цели, ради которой оно осу-

ществляется: сокращение записи, засекречивание (шифровка) информации, удобство обработки и т.д.

В связи с развитием информационных технологий кодирование является центральным вопросом при решении самых разных задач программирования, таких как:

- представление данных произвольной структуры (числа, текст, графика) в памяти компьютера;
- обеспечение помехоустойчивости при передаче данных по каналам связи;
- сжатие информации в базах данных.

Правило отображения одного алфавита  $A$  на другой  $B$  называется кодом, а сама процедура – кодированием сообщения. Обратная процедура сопоставления кодовым словам алфавита  $B$  символов алфавита  $A$  называется декодированием.

Например, при передаче сообщения по телеграфному каналу используется азбука Морзе (таблица 1).

Таблица 1

## Азбука Морзе

Русский символ	Код Морзе	Русский символ	Код Морзе
А	· –	Р	· – ·
Б	– · · ·	С	· · ·
В	· – –	Т	–
Г	– – ·	У	· · –
Д	– · ·	Ф	· · – ·
Е (также и Ё)	·	Х	· · · ·
Ж	· · · –	Ц	– · · ·
З	– – · ·	Ч	– – – ·
И	· ·	Ш	– – – –
Й	· – – –	Щ	– – · –
К	– · –	Ъ	– – · – –
Л	· – · ·	Ы	– · – –
М	– –	Ь	– · · –
Н	– ·	Э	· · – · ·
О	– – –	Ю	· · – –
П	· – – ·	Я	· – · –

Азбука Морзе является общеизвестным кодом из символов те-

леграфного алфавита, в котором буквам русского языка соответствуют кодовые слова (последовательности) из «точек» и «тире».

В технических устройствах хранения, передачи и обработки информации для её кодирования часто используют алфавиты, содержащие лишь два различных символа. Например, «плюс», «минус» («+», «-») или «1» и «0». Наличие всего двух символов значительно упрощает электрические схемы с электронными переключателями, которые принимают только два состояния – они либо проводят ток, либо нет.

Как известно, в качестве единицы измерения количества информации принят 1 бит (*англ. bit – binary digit* – двоичная цифра). Бит – один символ двоичного алфавита: 0 или 1. Наряду с битом получила распространение укрупнённая единица – байт, равный 8 битам. Из битов складывается все многообразие данных, которые обрабатывает компьютер. Комбинируя восемь нулей и единиц различными способами, можно получить 256 различных комбинаций. Этого количества достаточно, чтобы каждому символу поставить в соответствие свою неповторимую комбинацию из восьми нулей и единиц. Эти комбинации определяются кодовой таблицей ASCII (*American Standard Code for Information Interchange* – американский стандартный код для обмена информацией).

Одним байтом кодируется любой печатный знак (буква, цифра, любой другой символ). Например, число 7 выражается байтом 00000111; буква L – 01001100; знак «+» имеет двоичный код 00101011. Байт – один символ, который представляет комбинацию из 8 бит.

Порядковый номер символа в таблице ASCII называют десятичным кодом этого символа. Чтобы его определить необходимо сложить номер строки с номером столбца, которые соответствуют выбранному символу. Например, десятичный код цифры 7 равен 55 (7+48), а символа «%» – 37 (5+32). Таким образом, каждый символ имеет десятичный и двоичный код. Первые 32 символа являются управляющими и предназначены, в основном, для передачи различных команд.

В таблице 2 представлены ASCII коды для букв русского алфавита (пробел кодируется кодом 32<sub>10</sub>).

## ASCII – коды

	192	208	224	240
0	А	Р	а	р
1	Б	С	б	с
2	В	Т	в	т
3	Г	У	г	у
4	Д	Ф	д	ф
5	Е	Х	е	х
6	Ж	Ц	ж	ц
7	З	Ч	з	ч
8	И	Ш	и	ш
9	Й	Щ	й	щ
10	К	Ъ	к	ъ
11	Л	Ы	л	ы
12	М	Ь	м	ь
13	Н	Э	н	э
14	О	Ю	о	ю
15	П	Я	п	я

Известны два класса методов кодирования: равномерное и неравномерное кодирование. Под равномерным кодированием понимается использование кодов со словами постоянной длины. Для того чтобы декодирование равномерного кода было возможным, разным символам алфавита источника должны соответствовать разные кодовые слова. При этом длина кодового слова должна быть не меньше  $\lceil \log_n m \rceil$  символов, где  $m$  – размер исходного алфавита,  $n$  – размер кодового алфавита, знак  $\lceil \rceil$  означает округление до большего целого.

Для кодирования информации на русском языке, алфавит которого включает 33 буквы, равномерным двоичным кодом требуется построить кодовые слова длиной не меньше  $\lceil \log_2 33 \rceil = 6$ .

При неравномерном кодировании используются кодовые слова разной длины. Причем кодовые слова обычно строятся так, чтобы часто встречающиеся символы кодировались более короткими

кодowymi словами, а редкие символы – более длинными (за счет этого и достигается «сжатие» данных).

Под сжатием данных понимается компактное представление данных, достигаемое за счет избыточности информации, содержащейся в сообщениях. Большое значение для практического использования имеет неискажающее сжатие, позволяющее полностью восстановить исходное сообщение. При неискажающем сжатии происходит кодирование сообщения перед началом передачи или хранения, а после окончания процесса сообщение однозначно декодируется (это соответствует модели канала без шума (помех)).

Методы сжатия данных можно разделить на две группы: статические методы и адаптивные методы. Статические методы сжатия данных предназначены для кодирования конкретных источников информации с известной статистической структурой, порождающих определенное множество сообщений. Эти методы базируются на знании статистической структуры исходных данных. К наиболее известным статическим методам сжатия относятся коды Хаффмана, Шеннона, Фано, Гилберта-Мура, арифметический код и другие методы, которые используют известные сведения о вероятностях порождения источником различных символов или их сочетаний.

Если статистика источника информации неизвестна или изменяется с течением времени, то для кодирования сообщений такого источника применяются адаптивные методы сжатия. В адаптивных методах при кодировании очередного символа текста используются сведения о ранее закодированной части сообщения для оценки вероятности появления очередного символа. В процессе кодирования адаптивные методы «настраиваются» на статистическую структуру кодируемых сообщений, т.е. коды символов меняются в зависимости от накопленной статистики данных. Это позволяет адаптивным методам эффективно и быстро кодировать сообщение за один просмотр.

### **Код Хаффмана**

Метод оптимального побуквенного кодирования был разработан в 1952 г. Д. Хаффманом. Оптимальный двоичный код Хаффмана обладает минимальной средней длиной кодового слова среди

всех побуквенных кодов для источника с алфавитом

$$A = \{a_1, a_2, \dots, a_n\} \text{ и вероятностями } p_i = P(a_i), i = \overline{1, n}, \sum_{i=1}^n p_i = 1.$$

Алгоритм построения оптимального кода Хаффмана заключается в следующих шагах.

1. Упорядочим символы исходного алфавита  $A = \{a_1, a_2, \dots, a_n\}$  по убыванию их вероятностей  $p_1 \geq p_2 \geq \dots \geq p_n$ .

2. Если  $A = \{a_1, a_2\}$ , то  $a_1 \rightarrow 0, a_2 \rightarrow 1$ .

3. Если  $A = \{a_1, a_2, \dots, a_j, \dots, a_n\}$  и известны коды  $\langle a_j \rightarrow b_j \rangle, j = \overline{1, n}$ , то для алфавита  $A' = \{a_1, a_2, \dots, a'_j, a''_j, \dots, a_n\}$  с новыми символами  $a'_j, a''_j$  вместо  $a_j$ , и вероятностями  $p_j = p'_j + p''_j$  код символа  $a_j$  заменяется на коды  $a'_j \rightarrow b_j0$  и  $a''_j \rightarrow b_j1$ .

Например, пусть имеется алфавит  $A = \{a_1, a_2, a_3, a_4, a_5, a_6\}$  с вероятностями  $p_1 = 0,4, p_2 = 0,2, p_3 = 0,2, p_4 = 0,1, p_5 = 0,05, p_6 = 0,05$

В рассматриваемом случае символы алфавита уже упорядочены в соответствии с вероятностями их появления в сообщении. Будем складывать две наименьшие вероятности и включать суммарную вероятность на соответствующее место в упорядоченном списке вероятностей до тех пор, пока в списке не останется два символа. Тогда закодируем эти два символа 0 и 1. Далее кодовые слова достраиваются, как показано на рисунке 1.

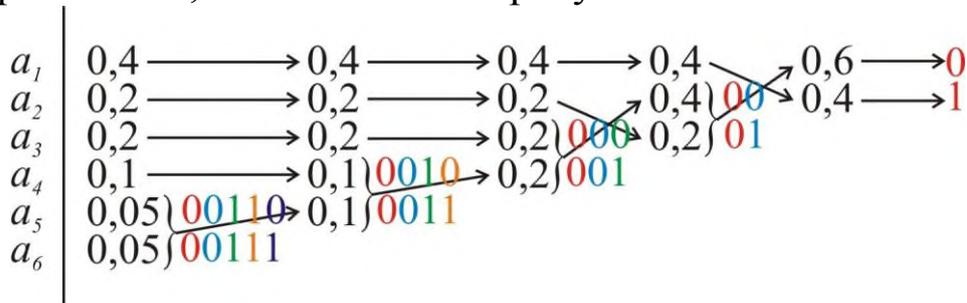


Рисунок 1 – Процесс построения кода Хаффмана

Результат кодирования представлен в таблице 3.

Результат кодирования кодом Хаффмана

Символ	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$
Вероятность	0,4	0,2	0,2	01	0,05	0,05
Код	1	01	000	0010	00110	00111

### Код Шеннона

Пусть имеется алфавит  $A = \{a_1, a_2, \dots, a_n\}$  с вероятностями символов  $p_i = P(a_i)$ ,  $i = \overline{1, n}$ ,  $\sum_{i=1}^n p_i = 1$ .

Код Шеннона строится следующим образом:

1. Упорядочим символы исходного алфавита  $A = \{a_1, a_2, \dots, a_n\}$  по убыванию их вероятностей  $p_1 \geq p_2 \geq \dots \geq p_n$ .

2. Вычислим величины  $Q_i$ ,  $i = \overline{1, n}$ , которые называются кумулятивные вероятности:

$$Q_0 = 0$$

$$Q_1 = p_1$$

$$Q_2 = p_1 + p_2$$

...

$$Q_i = p_1 + p_2 + \dots + p_i$$

...

$$Q_n = p_1 + p_2 + \dots + p_i + \dots + p_n = 1.$$

3. Представим  $Q_i$ ,  $i = \overline{1, n}$  в двоичной системе счисления и возьмем в качестве кодового слова первые  $\lceil \log_2 p_i \rceil$  двоичных знаков после запятой.

Например, пусть имеется алфавит  $A = \{a_1, a_2, a_3, a_4, a_5, a_6\}$  с вероятностями  $p_1 = 0,4$ ,  $p_2 = 0,2$ ,  $p_3 = 0,2$ ,  $p_4 = 0,1$ ,  $p_5 = 0,05$ ,  $p_6 = 0,05$ . Построенный код приведен в таблице 4.

Таблица 4

## Результат кодирования кодом Шеннона

Символ	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$
Вероятность	0,4	0,2	0,2	0,1	0,05	0,05
Величины $Q_i, i = \overline{1, n}$	0	0,4	0,6	0,8	0,9	0,95
Код	00	011	100	1100	11100	11110

**Код Фано**

Метод Фано, заключается в следующем.

1. Упорядоченный по убыванию вероятностей список символов алфавита источника делится на две части так, чтобы суммы вероятностей букв, входящих в эти части, как можно меньше отличались друг от друга.

2. Символам первой части приписывается 0, а символам из второй части – 1. Далее также поступают с каждой из полученных частей.

3. Процесс продолжается до тех пор, пока весь список не разобьется на части, содержащие по одной букве.

Например, пусть имеется алфавит  $A = \{a_1, a_2, a_3, a_4, a_5, a_6\}$  с вероятностями  $p_1 = 0,4$ ,  $p_2 = 0,2$ ,  $p_3 = 0,2$ ,  $p_4 = 0,1$ ,  $p_5 = 0,05$ ,  $p_6 = 0,05$ . Построенный код приведен в таблице 5.

Таблица 5

## Результат кодирования кодом Фано

Символ	Вероятность	Формирование кодового слова				Код
$a_1$	0,4	0	0			00
$a_2$	0,2		1			01
$a_3$	0,2	1	0			10
$a_4$	0,1			0		110
$a_5$	0,05		1		0	1110
$a_6$	0,05			1	1	1111

## Шифр Цезаря

Шифр Цезаря, также известный как шифр сдвига, – один из самых простых и наиболее широко известных методов шифрования.

Шифр Цезаря – это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется символом, находящимся на некотором постоянном числе позиций левее или правее него в алфавите. Например, в шифре со сдвигом 3 «А» была бы заменена на «Г», «Б» станет «Д», и так далее.



Рисунок 2 – Шифр Цезаря со сдвигом 3

Используемое преобразование обычно обозначают как ROTN, где N – сдвиг, ROT – сокращение от слова ROTATE, в данном случае «циклический сдвиг». Число разных преобразований конечно и зависит от длины алфавита. Для русского языка возможно 32 разных преобразования (преобразования ROT0 и ROT33 сохраняют исходный текст, а дальше начинаются уже повторения).

### Задание

#### Вариант 1

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент кодирования фразы на русском языке с помощью азбуки Морзе.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода фразы;
- на листе табличного процессора должна быть отведена ячейка для вывода закодированной фразы;
- ячейки должны содержать комментарии, поясняющие их назначение.

#### Вариант 2

Используя стандартные средства табличного процессора (таб-

лица 1), разработать инструмент декодирования фразы, закодированной с помощью азбуки Морзе.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода закодированной фразы;
- на листе табличного процессора должна быть отведена ячейка для вывода декодированной фразы;
- ячейки должны содержать комментарии, поясняющие их назначение.

Вариант 3

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент кодирования фразы на русском языке с помощью кода ASCII.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода фразы;
- на листе табличного процессора должна быть отведена ячейка для вывода закодированной фразы;
- ячейки должны содержать комментарии, поясняющие их назначение.

Вариант 4

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент декодирования фразы, закодированной с помощью кода ASCII.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода закодированной фразы;
- на листе табличного процессора должна быть отведена ячейка для вывода декодированной фразы;
- ячейки должны содержать комментарии, поясняющие их назначение.

Вариант 5

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент кодирования фразы на русском языке с помощью кода Хаффмана.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка

ка для ввода фразы;

– на листе табличного процессора должна быть отведена ячейка для вывода закодированной фразы;

– ячейки должны содержать комментарии, поясняющие их назначение.

#### Вариант 6

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент кодирования фразы на русском языке с помощью кода Шеннона.

Требования к выполнению задания.

– на листе табличного процессора должна быть отведена ячейка для ввода фразы;

– на листе табличного процессора должна быть отведена ячейка для вывода закодированной фразы;

– ячейки должны содержать комментарии, поясняющие их назначение.

#### Вариант 7

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент кодирования фразы на русском языке с помощью кода Фано.

Требования к выполнению задания.

– на листе табличного процессора должна быть отведена ячейка для ввода фразы;

– на листе табличного процессора должна быть отведена ячейка для вывода закодированной фразы;

– ячейки должны содержать комментарии, поясняющие их назначение.

#### Вариант 8

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент кодирования фразы на русском языке с помощью шифра Цезаря.

Требования к выполнению задания.

– на листе табличного процессора должна быть отведена ячейка для ввода фразы;

– на листе табличного процессора должна быть отведена ячейка для вывода закодированной фразы;

– на листе табличного процессора должна быть отведена ячейка

ка для ввода величины сдвига;

– ячейки должны содержать комментарии, поясняющие их назначение.

#### Вариант 9

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент декодирования фразы, закодированной с помощью шифра Цезаря.

Требования к выполнению задания.

– на листе табличного процессора должна быть отведена ячейка для ввода закодированной фразы;

– на листе табличного процессора должна быть отведена ячейка для вывода декодированной фразы;

– на листе табличного процессора должна быть отведена ячейка для ввода величины сдвига;

– ячейки должны содержать комментарии, поясняющие их назначение.

#### Общие требования

При оформлении работы в Excel необходимо использовать шрифт Times New Roman, размер 14, начертание обычное. Размеры ячеек выбираются по контексту, каждая ячейка должна иметь пояснения, какая информация в ней находится.

Составить отчет по результатам выполнения лабораторной работы. Отчет должен содержать:

– титульный лист (Приложение А);

– Содержание;

– Цель работы;

– Задание;

– Словесный подробный алгоритм выполнения задания с указанием функций процессора, используемых для выполнения конкретного действия;

– Скриншот интерфейса инструмента преобразования чисел;

– Скриншот листа выполнения задания с отображением формул и с отображением результатов вычислений;

– Выводы по работе.

Номера страниц проставляются внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первой страницей является титульный лист, нумерация проставляется, начиная с Цели рабо-

ты отчета.

Макет Оглавления приведен в Приложении Б.

Для оформления заголовков разделов следует использовать стиль «Заголовок 1». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 12 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Для оформления текста отчета необходимо использовать стиль «Обычный». Параметры стиля:

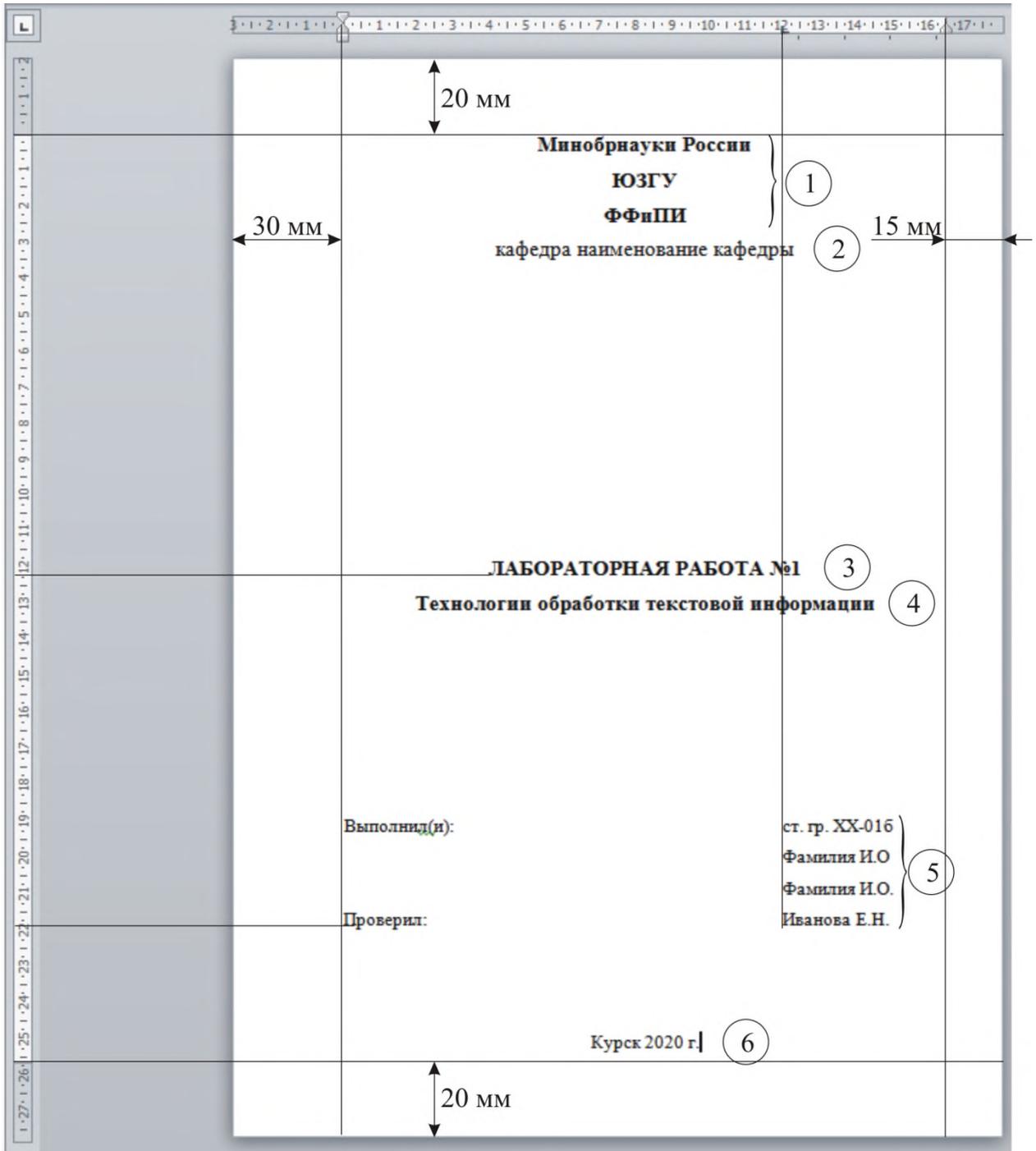
- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

### **Список использованных источников**

1. Котенко, В.В. Теория информации [Текст] : учебное пособие / В.В. Котенко. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2018. – 240 с. – Режим доступа: URL: <https://biblioclub.ru/>

2. Майстренко, Н.В. Основы теории информации и криптографии: учебное электронное издание [Текст] / Н.В. Майстренко, А.В. Майстренко. – Тамбов : Тамбовский государственный технический университет (ТГТУ), 2018. – 81 с. – Режим доступа: URL: <https://biblioclub.ru/index.php?page=book&id=494706>.

## Оформление титульного листа отчета по лабораторной работе



Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 16 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

3: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

4: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

5: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по ширине, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки, позиция табуляции 12 см.

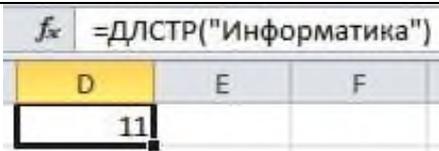
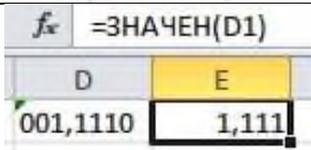
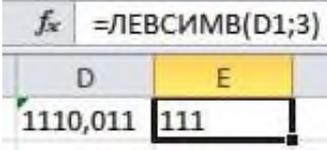
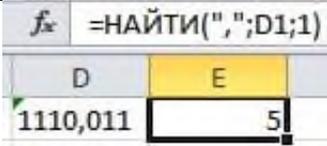
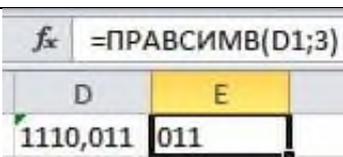
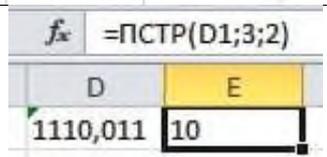
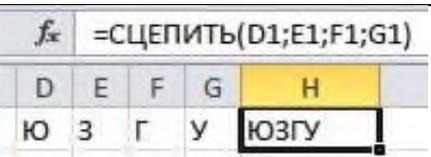
6: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

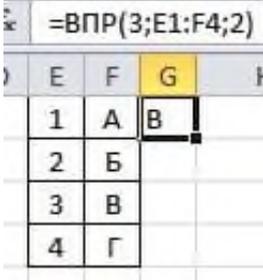


1: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по левому краю, отступы слева 0, справа 0, интервалы перед 0, после 0, междустрочный 1,5 строки.

## Справка по функциям табличного процессора

функция	назначение	пример использования
1	2	3
ДЛСТР(текст)	возвращает количество знаков в текстовой строке	
ЗНАЧЕН(текст)	преобразует текстовый аргумент в число	
ЛЕВСИМВ(текст; количество_знаков)	возвращает указанное количество знаков с начала строки текста	
НАЙТИ(искомый_текст; просматриваемый_текст; начальная позиция)	Возвращает позицию начала искомой строки текста в содержащей ее строке текста. Прописные и строчные буквы различаются	
ПРАВСИМВ(текст; число_знаков)	возвращает указанное число знаков с конца строки текста	
ПСТР(текст; начальная_позиция; количество_знаков)	возвращает заданное число знаков из строки текста, начиная с указанной позиции	
СЦЕПИТЬ(текст1; текст2;...)	объединяет несколько текстовых строк в одну	

1	2	3																				
<p>ВПР(искомое_значение; таблица; номер_столбца; ...)</p>	<p>ищет значение в крайнем левом столбце таблицы и возвращает значение ячейки, находящейся в указанном столбце той же строки</p>	 <p>=ВПР(3;E1:F4;2)</p> <table border="1" data-bbox="1093 369 1356 604"> <thead> <tr> <th></th> <th>E</th> <th>F</th> <th>G</th> </tr> </thead> <tbody> <tr> <th>1</th> <td>А</td> <td></td> <td>В</td> </tr> <tr> <th>2</th> <td>Б</td> <td></td> <td></td> </tr> <tr> <th>3</th> <td>В</td> <td></td> <td></td> </tr> <tr> <th>4</th> <td>Г</td> <td></td> <td></td> </tr> </tbody> </table>		E	F	G	1	А		В	2	Б			3	В			4	Г		
	E	F	G																			
1	А		В																			
2	Б																					
3	В																					
4	Г																					
<p>ГПР(искомое_значение; таблица; номер_строки; ...)</p>	<p>ищет значение в верхней строке таблицы и возвращает значение ячейки, находящейся в указанной строке того же столбца</p>	 <p>=ГПР(3;E1:H2;2)</p> <table border="1" data-bbox="1069 761 1380 918"> <thead> <tr> <th></th> <th>E</th> <th>F</th> <th>G</th> <th>H</th> <th>I</th> </tr> </thead> <tbody> <tr> <th>1</th> <td>2</td> <td>3</td> <td>4</td> <td></td> <td>В</td> </tr> <tr> <th>2</th> <td>А</td> <td>Б</td> <td>В</td> <td>Г</td> <td></td> </tr> </tbody> </table>		E	F	G	H	I	1	2	3	4		В	2	А	Б	В	Г			
	E	F	G	H	I																	
1	2	3	4		В																	
2	А	Б	В	Г																		

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 23.12.2021 11:40:58

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d79e5f1c11eabbf73e943df4a4851fda56d089

## МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра вычислительной техники



## Технологии представления числовой информации в ЭВМ

Методические указания  
по выполнению практической работы по дисциплине  
«Информатика» для студентов специальности  
10.05.02 Информационная безопасность  
телекоммуникационных систем

Курск 2021 г.

УДК 681.3

Составители: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина

Рецензент

Доцент кафедры программной инженерии,  
кандидат технических наук

*Ю.А. Халин*

**Технологии представления числовой информации в ЭВМ:**

методические указания по выполнению практической работы /  
Юго-Зап. гос. ун-т; сост.: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина. – Курск, 2021. – 13 с.: табл. 1. – Библиограф.: с. 13.

Приводятся алгоритмы и методы преобразования числовой информации к виду, адаптированному для ЭВМ. Рассматриваются приемы использования встроенных средств табличного процессора для решения задач преобразования числовой информации. Теоретический материал сопровождается примерами.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

## **Цель работы**

Получить знания о видах представления числовой информации в ЭВМ, методах и алгоритмах преобразования числовой информации; научиться осуществлять выбор метода преобразования числовой информации для получения результата, удовлетворяющего заданным параметрам; овладеть навыками использования различных инструментов, предоставляемых табличным процессором, для преобразования числовой информации в соответствии с требованиями.

## **Краткая теоретическая информация**

### **Системы счисления**

Первым необходимым понятием в рассмотрении вопроса о представлении чисел в ЭВМ является система счисления. История систем счисления восходит к античному периоду развития математики. Высшим достижением древней арифметики является открытие позиционного принципа представления чисел. Первой из известных систем счисления, основанных на позиционном принципе, была вавилонская 60-ричная система счисления, возникшая в Древнем Вавилоне примерно во 2-м тысячелетии до нашей эры.

В обыденной жизни для представления числовой информации используется десятичная система счисления или система счисления с основанием 10. В цифровых устройствах обработки информации используется двоичная система счисления или система счисления с основанием 2 (основная), а также восьмеричная и шестнадцатеричная.

Система счисления – способ наименования и представления чисел с помощью символов, имеющих определенные количественные значения. Системы счисления могут быть непозиционные и позиционные. В непозиционной системе счисления количественное значение символа не зависит от его позиции в ряду символов, изображающих это число. Примером такой системы является римская система счисления, счетные палочки. Позиционные системы счисления – системы, в которых количественное значение символа зависит от его позиции в ряду символов, изображающих это число.

Данные системы удобны тем, что в них для записи числа требуется небольшое количество символов.

Основание системы счисления это количество различных символов, используемых для изображения числа. В общем случае любое число, представленное в позиционной системе счисления, можно записать в виде:

$$X_P = a_{m-1}P^{m-1} + a_{m-2}P^{m-2} + \dots + a_1P^1 + a_0P^0 + a_{-1}P^{-1} + \dots + a_{-n}P^{-n},$$

где  $P$  – основание системы счисления;

$m+1$  – количество разрядов в целой части числа;

$n$  – количество разрядов в дробной части числа;

$a_i, i = -s, m$  – цифры числа  $X$ , записанного в системе счисления с основанием  $P$ .

Запись чисел в системе счисления с основанием  $N$ .

Для записи чисел в системе счисления с основанием  $N$  нужно иметь алфавит из  $N$  символов. Обычно для этого используют арабские цифры, если  $N \leq 10$ , и буквы латинского алфавита, если  $N > 10$ . Алфавит для различных систем счисления приведен в таблице 1.

Таблица 1

Характеристики систем счисления

основание	название	алфавит
2	двоичная	0 1
8	восьмеричная	0 1 2 3 4 5 6 7
10	десятичная	0 1 2 3 4 5 6 7 8 9
16	шестнадцатеричная	0 1 2 3 4 5 6 7 8 9 A B C D E F

Чтобы не путать, в какой системе счисления записано число, его снабжают указателем справа внизу. Например, число в десятичной системе –  $5_{10}$ , в двоичной –  $101_2$ . Иногда двоичное число обозначают префиксом  $0b$  или символом  $\&$  (амперсанд), например,  $0b101$  или соответственно  $\&101$ .

В системах счисления с основанием  $N$  (кроме десятичной) знаки читаются по одному. Например, двоичное число  $101_2$  произносится «один ноль один».

### Перевод чисел из десятичной системы счисления в систему счисления с основанием $N$ .

Правила перевода из десятичной системы счисления ( $CC_{10}$ ) в систему счисления с основанием  $N$  ( $CC_N$ ) будем рассматривать для неправильных дробей, т.е. чисел, содержащих и дробную, и целую части.

Для перевода целого числа  $X$  из  $CC_{10}$  в  $CC_N$  надо число  $X$  последовательно делить (по правилам деления с остатком) на основание системы счисления, в которую это число переводится –  $N$ , до тех пор, пока не будет получено частное, равное нулю. Число в новой системе счисления сформируется из остатков от деления, записанных в порядке, обратном их получению. Например, переведем число  $352_{10}$  в  $CC_2$ .

352		2											
352		176		2									
0		176		88		2							
		0		88		44		2					
		0		44		22		2					
		0		22		11		2					
		0		10		5		2					
		1		4		2		2					
		1		2		1		2					
		0		0		0		2					
		1											

$$352_{10} = 101100000_2$$

Переведем число  $352_{10}$  в  $CC_5$ .

$$\begin{array}{r|l}
 352 & 5 \\
 \hline
 350 & 70 \\
 \hline
 2 & 70 \\
 \hline
 & 0 \\
 & 14 \\
 & 10 \\
 & 4 \\
 & 2 \\
 & 0 \\
 & 2
 \end{array}$$

$$352_{10} = 4204_5$$

Для перевода правильной дроби  $X$  в систему счисления с основанием  $N$  необходимо умножить исходную дробь (а дальше только дробные части произведения, выделяя целые части) последовательно на основание системы счисления  $N$ . Полученные в результате умножения целые части произведения являются значениями разрядов дробного числа в системе счисления с основанием  $N$ . Вычисления можно закончить в случаях, если:

- полученная в результате умножения дробь равна 0;
- достигнута заданная точность.

Например, переведем число  $0,352_{10}$  в  $CC_2$ . Точность перевода – 4 знака после запятой:

$0,352$	$0,704$	$0,408$	$0,816$
$\underline{\quad 2}$	$\underline{\quad 2}$	$\underline{\quad 2}$	$\underline{\quad 2}$
$0,704$	$1,408$	$0,816$	$1,632$
<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>

$$0,352_{10} = 0.0101_2$$

Переведем число  $0,352_{10}$  в  $CC_5$ .

$0,352$	$0,76$	$0,8$
$\underline{\quad 5}$	$\underline{\quad 5}$	$\underline{\quad 5}$
$1,760$	$3,800$	$4,000$
<b>1</b>	<b>3</b>	<b>4</b>

$$0,352_{10} = 0.134_5$$

В последнем примере после получения третьего произведения процесс завершен, так как цифры дробной части оказались равными нулю.

## Перевод чисел из системы счисления с основанием $N$ в десятичную систему счисления

Правила перевода будем рассматривать также для неправильных дробей, т.е. чисел, содержащих и дробную, и целую части.

Для перевода числа  $X$  из  $CC_N$  в  $CC_{10}$  необходимо выполнить следующие действия:

- пронумеровать все разряды числа  $X$ , причем младший разряд целой части имеет номер «0», влево номера разрядов получаются в результате инкремента («+1»), вправо номера разрядов получаются в результате декремента («-1»);
- получить вес каждого разряда по формуле  $N^i$ , где  $N$  – основание системы счисления;  $i$  – номер разряда;
- вычислить сумму произведений значения каждого разряда числа на вес этого разряда.

Полученная сумма – это и есть число в  $CC_{10}$ . Например, переведем число  $312,023_4$  в  $CC_{10}$ :

$$\begin{array}{ccccccc} 3 & 1 & 2, & 0 & 2 & 3 & \\ \begin{array}{ccccccc} 2 & 1 & 0 & -1 & -2 & -3 \\ 4^2 & 4^1 & 4^0 & 4^{-1} & 4^{-2} & 4^{-3} \end{array} & = & 3 \cdot 4^2 & + & 1 \cdot 4^1 & + & 2 \cdot 4^0 & + & 0 \cdot 4^{-1} & + & 2 \cdot 4^{-2} & + & 3 \cdot 4^{-3} \end{array}$$

Веса разрядов

Номера разрядов

$$312,023_4 = 3 \cdot 4^2 + 1 \cdot 4^1 + 2 \cdot 4^0 + 0 \cdot 4^{-1} + 2 \cdot 4^{-2} + 3 \cdot 4^{-3} = 54,171875_{10}$$

## Перевод чисел из двоичной системы счисления в восьмеричную и шестнадцатеричную систему счисления

Правила перевода будем рассматривать также для неправильных дробей, т.е. чисел, содержащих и дробную, и целую части.

Для перевода числа  $X$  из  $CC_2$  в  $CC_8$  необходимо выполнить следующие действия:

- разбить цифры двоичного числа на группы по три цифры (триады), начиная от десятичной точки влево и вправо, при необходимости можно дописать слева и справа незначащие нули;
- каждую группу заменить восьмеричной цифрой.

Например, переведем двоичное число  $11110001010101,111001_2$  в восьмеричную систему счисления:

$$| \underbrace{011}_3 | \underbrace{110}_6 | \underbrace{001}_1 | \underbrace{010}_2 | \underbrace{101}_5 | , \underbrace{111}_7 | \underbrace{001}_1 |$$

$$11110001010101,111001_2 = 36125,71_8$$

Для перевода числа  $X$  из  $CC_2$  в  $CC_{16}$  необходимо выполнить следующие действия:

- разбить цифры двоичного числа на группы по четыре цифры (тетрады), начиная от десятичной точки влево и вправо, при необходимости можно дописать слева и справа незначащие нули;
- каждую группу заменить шестнадцатеричной цифрой.

Например, переведем двоичное число  $11110001010101,111001_2$  в шестнадцатеричную систему счисления:

$$| \underbrace{0011}_3 | \underbrace{1100}_{C(12)} | \underbrace{0101}_5 | \underbrace{0101}_5 | , \underbrace{1110}_{E(14)} | \underbrace{0100}_4 |$$

$$11110001010101,111001_2 = 3C55,E4_{16}$$

### **Перевод чисел из восьмеричной системы счисления и шестнадцатеричной системы счисления в двоичную систему счисления**

Правила перевода будем рассматривать также для неправильных дробей, т.е. чисел, содержащих и дробную, и целую части.

Для перевода числа  $X$  из  $CC_8$  в  $CC_2$  необходимо каждую восьмеричную цифру заменить двоичной триадой. В записи двоичного числа левые и правые нули писать не нужно.

Например, переведем восьмеричное число  $657,024_8$  в двоичную систему счисления:

$$\begin{array}{ccccccc} \underline{6} & \underline{5} & \underline{7} & , & \underline{0} & \underline{2} & \underline{4} \\ 11010111 & 000010100 & & & & & \end{array}$$

$$657,024_8 = 110101111,0000101_2$$

Для перевода числа  $X$  из  $CC_{16}$  в  $CC_2$  необходимо каждую шестнадцатеричную цифру заменить двоичной тетрадой. В записи двоичного числа левые и правые нули писать не нужно.

Например, переведем шестнадцатеричное число  $2F50,C2D_8$  в двоичную систему счисления:

$$\begin{array}{ccccccc} \underline{2} & \underline{F} & \underline{5} & \underline{0} & , & \underline{C} & \underline{2} & \underline{D} \\ 001011101010000 & 110000101011 & & & & & & \end{array}$$

$$2F50,C2D_{16} = 10111101010000,110000101011_2$$

## Задание

### Вариант 1

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент перевода чисел (неправильных дробей) из десятичной системы счисления в систему счисления с основанием  $N$ .

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода десятичного числа, предназначенного для перевода;
- на листе табличного процессора должна быть отведена ячейка для ввода основания новой системы счисления ( $N$ );
- на листе табличного процессора должна быть отведена ячейка для вывода числа в новой системе счисления;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина целой части исходного десятичного числа не должна превосходить 8 разрядов, длина дробной части исходного десятичного числа не должна превосходить 4 разрядов, основание новой системы счисления должна быть не больше 16.

### Вариант 2

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент перевода чисел (неправильных дробей) из системы счисления с основанием  $N$  в десятичную систему счисления.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода числа, предназначенного для перевода;
- на листе табличного процессора должна быть отведена ячейка для ввода основания исходной системы счисления ( $N$ );
- на листе табличного процессора должна быть отведена ячейка для вывода числа в десятичной системе счисления;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина целой части исходного числа не должна превосходить 8 разрядов, длина дробной части исходного числа не должна превосходить 4 разрядов, основание системы счисления должна быть не больше 16.

### Вариант 3

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент перевода чисел (неправильных дробей) из восьмеричной системы счисления в шестнадцатеричную систему счисления.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода числа, предназначенного для перевода;
- на листе табличного процессора должна быть отведена ячейка для вывода числа в шестнадцатеричной системе счисления;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина целой части исходного числа не должна превосходить 8 разрядов, длина дробной части исходного числа не должна превосходить 4 разрядов.

### Вариант 4

Используя стандартные средства табличного процессора (таблица 1), разработать инструмент перевода чисел (неправильных дробей) из шестнадцатеричной системы счисления в восьмеричную систему счисления.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода числа, предназначенного для перевода;
- на листе табличного процессора должна быть отведена ячейка для вывода числа в восьмеричной системе счисления;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина целой части исходного числа не должна превосходить 8 разрядов, длина дробной части исходного числа не должна превосходить 4 разрядов.

### Общие требования

При оформлении работы в Excel необходимо использовать шрифт Times New Roman, размер 14, начертание обычное. Размеры ячеек выбираются по контексту, каждая ячейка должна иметь пояснения, какая информация в ней находится.

Составить отчет по результатам выполнения лабораторной работы. Отчет должен содержать:

- титульный лист (Приложение А);
- Содержание;
- Цель работы;
- Задание;
- Словесный подробный алгоритм выполнения задания с указанием функций процессора, используемых для выполнения конкретного действия;
- Скриншот интерфейса инструмента преобразования чисел;
- Скриншот листа выполнения задания с отображением формул и с отображением результатов вычислений;
- Выводы по работе.

Номера страниц проставляются внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первой страницей является титульный лист, нумерация проставляется, начиная с содержательной части отчета.

Макет Оглавления приведен в Приложении Б.

Для оформления заголовков разделов следует использовать стиль «Заголовок 1». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 12 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Для оформления текста отчета необходимо использовать стиль «Обычный». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

## Справка по функциям табличного процессора

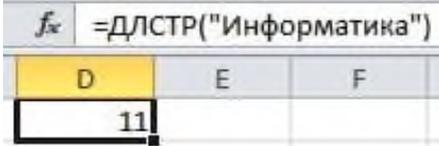
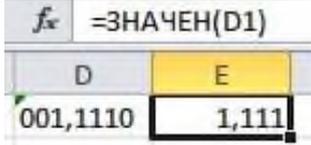
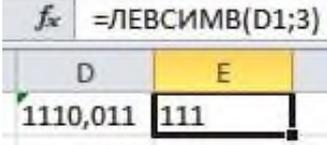
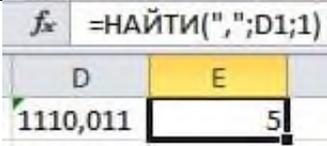
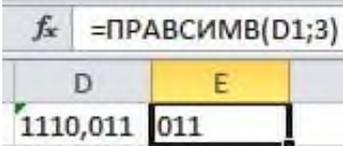
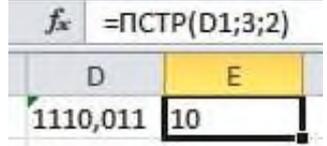
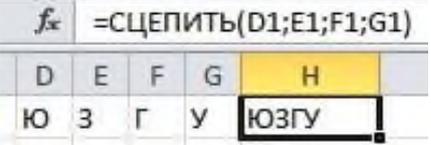
функция	назначение	пример использования
1	2	3
ДЛСТР(текст)	возвращает количество знаков в текстовой строке	
ЗНАЧЕН(текст)	преобразует текстовый аргумент в число	
ЛЕВСИМВ(текст; количество_знаков)	возвращает указанное количество знаков с начала строки текста	
НАЙТИ(искомый_текст; просматриваемый_текст; начальная позиция)	Возвращает позицию начала искомой строки текста в содержащей ее строке текста. Прописные и строчные буквы различаются	
ПРАВСИМВ(текст; число_знаков)	возвращает указанное число знаков с конца строки текста	
ПСТР(текст; начальная_позиция; количество_знаков)	возвращает заданное число знаков из строки текста, начиная с указанной позиции	
СЦЕПИТЬ(текст1; текст2;...)	объединяет несколько текстовых строк в одну	

Таблица 1 (продолжение)

1	2	3
ВПР(искомое_значение; таблица; номер_столбца; ...)	ищет значение в крайнем левом столбце таблицы и возвращает значение ячейки, находящейся в указанном столбце той же строки	
ГПР(искомое_значение; таблица; номер_строки; ...)	ищет значение в верхней строке таблицы и возвращает значение ячейки, находящейся в указанной строке того же столбца	

### Список использованных источников

1. Колокольникова, А.И. Информатика [Текст] : учебное пособие / А.И. Колокольникова. – Москва ; Берлин : Директ-Медиа, 2020. – 289 с. : ил., табл. – Режим доступа: URL: <https://biblioclub.ru/>
2. Информатика [Текст] : лабораторный практикум / авт.-сост. О.В. Вельц. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2018. – 117 с. – Режим доступа: URL: <https://biblioclub.ru/index.php?page=book&id=494706>.

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 23.12.2021 11:40:58

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d79e5f1c11eabbf73e943df4a4851fda56d089

## **МИНОБРНАУКИ РОССИИ**

**Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)**

**Кафедра вычислительной техники**



### **Компьютерная арифметика**

**Методические указания  
по выполнению практической работы по дисциплине  
«Информатика» для студентов специальности  
10.05.02 Информационная безопасность  
телекоммуникационных систем**

Курск 2021 г.

УДК 681.3

Составители: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина

Рецензент

Доцент кафедры программной инженерии,  
кандидат технических наук

*Ю.А. Халин*

**Компьютерная арифметика:** методические указания по выполнению практической работы / Юго-Зап. гос. ун-т; сост.: С.В. Дегтярев, Е.Н. Иванова, Т.Н. Конаныхина. – Курск, 2021. – 21 с.: табл. 3. – Библиограф.: с. 17.

Приводятся алгоритмы двоичного кодирования. Рассматривается методика выполнения арифметических операций в компьютере. Приводятся приемы использования встроенных средств табличного процессора для выполнения арифметических операций. Теоретический материал сопровождается примерами.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.  
Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## Цель работы

Получить знания о видах двоичного кодирования в ЭВМ, методах и алгоритмах выполнения арифметических операций; научиться выполнять арифметические операции над кодами двоичных чисел; овладеть навыками использования различных инструментов, предоставляемых табличным процессором, для выполнения операций двоичной арифметики.

## Краткая теоретическая информация

### Способы представления двоичных чисел в компьютерной арифметике

Все современные ЭВМ имеют достаточно развитую систему команд, включающую десятки и сотни машинных операций. Однако выполнение любой операции основано на использовании простейших микроопераций типа «сложение» и «сдвиг». Это позволяет иметь единое арифметико-логическое устройство для выполнения любых операций, связанных с обработкой информации.

В принципы работы вычислительных машин заложен принцип двоичного кодирования. В таблице 1 представлены правила сложения двоичных цифр двух чисел  $A$  и  $B$ . Здесь показаны правила сложения двоичных цифр  $A_i$  и  $B_i$  одноименных разрядов  $i$  ( $i \geq 0$ ) с учетом возможных переносов из предыдущего разряда  $P_{i-1}$  ( $P_0 = 0$ ). Именно данные таблицы 1 положены в основу выполнения любой операции ЭВМ.

В обычной математике мы работаем с числами, которые могут быть и положительными, и отрицательными. При записи двоичных чисел со знаком в их формате необходимо предусмотреть два поля: поле, определяющее знак числа, и поле, характеризующее модуль числа. Под знак числа отводится специальный знаковый бит (один двоичный разряд). Остальные разряды определяют модуль числа. Знаковый разряд приписывается слева от модуля числа (старший разряд), причём знаку «+» соответствует нулевое значение знакового бита, а знаку «-» – единичное.

Кодируя не только значение числа, но и его знак, мы получаем

прямой код числа.

Таблица 1

## Правила сложения двоичных цифр

значения разрядов двоичных чисел $A$ и $B$			значение разряда суммы	значение переноса в следующий разряд
$A_i$	$B_i$	$P_{i-1}$	$S_i$	$P_i$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Действия над прямыми кодами двоичных чисел при выполнении операции сложения создают большие трудности, связанные с необходимостью учета значений знаковых разрядов:

– во-первых, следует отдельно обрабатывать значащие разряды чисел и разряды знака;

– во-вторых, значение разряда знака влияет на алгоритм выполнения операции.

Во всех ЭВМ все операции выполняются над числами со знаками, представленными специальными машинными кодами.

В истории развития компьютеров использовались три основных варианта представления знаковых чисел:

- прямой код или знак и величина;
- обратный код или код с дополнением до единицы;
- дополнительный код или код с дополнением до двух.

Итак, прямой код двоичного числа образуется из абсолютного значения этого числа и кода знака, например:

$$A_{10} = +12 \text{ ПК}(A_2) = 01100 \\ + 12_{10}$$

$$B_{10} = -21 \text{ ПК}(B_2) = 110101 \\ - 21_{10}$$

Во всех трёх кодах положительные числа выглядят одинаково. Различия в форме записи отрицательных чисел в обратном и дополнительном кодах касаются только способа представления модуля числа, а способ кодирования и место расположения знакового бита остаются неизменными.

Обратный код получается путем инвертирования всех битов модуля (1 меняем на 0, а 0 меняем на 1).

$$A_{10} = +12 \text{ ПК}(A_2) = 01100 \text{ ОК}(A_2) = 01100 \\ + 12_{10}$$

$$B_{10} = -21 \text{ ПК}(B_2) = 110101 \text{ ОК}(B_2) = 101010 \\ - 21_{10}$$

Дополнительный код можно получить следующими способами. Первый:

– получить обратный код;

– к обратному коду прибавить «1» арифметическим сложением, например:

$$A_{10} = +12 \text{ ПК}(A_2) = 01100 \text{ ОК}(A_2) = 01100 \text{ ДК}(A_2) = 01100 \\ + 12_{10}$$

$$B_{10} = -21 \text{ ПК}(B_2) = 110101 \text{ ОК}(B_2) = 101010 \text{ ДК}(B_2) = 101011 \\ - 21_{10}$$

Второй:

Выполнить инверсию всех старших (за исключением знакового бита) битов числа кроме последней (младшей) единицы и тех нулей, которые стоят после нее, например:

$$\text{ПК}(A_2) = 110101 \text{ ДК}(A_2) = 101011$$

□□□□□□□□□□□□□□□□  
инверсия

Дополнительный код – наиболее распространенный способ представления отрицательных чисел.

## Простейшие арифметические операции с целыми числами

При выполнении алгебраического сложения знаковый разряд

и цифры модуля рассматриваются как единое целое и обрабатываются совместно. Перенос из старшего (знакового) разряда в обратном и дополнительном кодах учитывается по-разному. В случае обратного кода единица переноса из знакового разряда прибавляется к младшему разряду суммы. При использовании дополнительного кода единица переноса из знакового разряда отбрасывается.

Сложение (вычитание). Операция вычитания приводится к операции сложения путем преобразования чисел в обратный или дополнительный код согласно таблице 2.

Таблица 2

Требуемая операция	Необходимое преобразование
$A + B$	$A + B$
$A - B$	$A + (-B)$
$-A + B$	$(-A) + B$
$-A - B$	$(-A) + (-B)$

Здесь  $A$  и  $B$  неотрицательные числа. Скобки в представленных выражениях указывают на замену операции вычитания операцией сложения с обратным или дополнительным кодом соответствующего числа. Сложение двоичных чисел осуществляется последовательно, поразрядно в соответствии с таблицей 1. При выполнении сложения чисел необходимо соблюдать следующие правила:

- слагаемые должны иметь одинаковое число разрядов. Для выравнивания разрядной сетки слагаемых можно дописывать незначащие нули слева к целой части числа до знакового разряда и незначащие нули справа к дробной части числа;

- знаковые разряды участвуют в сложении так же, как и значащие.

- приписанные незначащие нули изменяют свое значение при преобразованиях по общему правилу.

- знак результата формируется автоматически;

- результат представляется в том коде, в котором представлены исходные слагаемые.

Числа, хранящиеся в компьютере, не могут быть сколь угодно

большими и имеют некоторые предельные значения. Пусть для хранения чисел используются 8-ми разрядные регистры. Тогда при попытке записать в него число  $-129_{10}=110000001_2$  возникнет ситуация, связанная с нехваткой технического устройства для записи 9-го разряда. Такая «аварийная» ситуация называется переполнением разрядной сетки, или просто переполнением. (англ. overflow - переполнение «сверху»).

Правила сложения чисел в прямых кодах:

- складывать можно только числа с одинаковыми знаками;
- перед выполнением операции знак чисел запоминается, и складываются модули чисел (в знаковом бите записывается «0»);
- если получившаяся сумма в знаковом разряде имеет «1» («–»), то произошло переполнение, в этом случае числового значения в ответе писать не нужно;
- если получившаяся сумма в знаковом разряде имеет «0» («+»), то запомненное значение знакового разряда числа нужно восстановить, это и будет результат операции сложения.

Например, сложим числа  $A_{10} = -5$  ПК( $A_2$ ) = 10000101 и  $B_{10} = -29$  ПК( $B_2$ ) = 10011101 в формате 8 бит:

Числа имеют одинаковые знаки, запоминаем знак  $Z_n = 1$ , складываем модули

$$\begin{array}{r} + 00000101 \\ \quad 00011101 \\ \hline 00100010 \end{array}$$

Сумма 00100010 в знаковом разряде содержит «0» («+»), восстанавливаем истинный знаковый разряд, тогда прямой код суммы это ПК( $SUM_2$ ) = 10100010, что соответствует  $SUM_{10} = -34$ .

Правила сложения чисел в обратных кодах (перед началом операции числа должны быть представлены в обратных кодах):

- дополняем каждое число битом переноса, в который заносим значение, равное значению знакового разряда числа;
- выполняем операцию сложения по правилам двоичной арифметики, не выделяя знаковый бит и бит переноса;
- если при вычислении суммы появился перенос из старшего разряда, то его нужно прибавить к числу;

– если значение бита переноса суммы не равно значению знакового разряда, то произошло переполнение, в этом случае числового значения в ответе писать не нужно;

– если значение бита переноса суммы равно значению знакового разряда, то получен обратный код суммы.

Например, сложим числа  $A_{10} = -5$  ОК( $A_2$ ) = 11111010 и  $B_{10} = -29$  ОК( $B_2$ ) = 11100010 в формате 8 бит. При сложении для каждого числа введем еще один бит – бит переноса ( $C$ ), начальное значение этого разряда равен значению знакового разряда числа ( $S$ ):

$$\begin{array}{r}
 C \quad S \\
 +1 \quad | \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \\
 \quad \quad | \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \\
 \hline
 1 \quad | \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0
 \end{array}$$

При сложении возник перенос из бита  $C$ . По правилам сложения в обратных кодах этот перенос нужно прибавить к числу:

$$\begin{array}{r}
 C \quad S \\
 +1 \quad | \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \\
 \quad \quad | \quad 0 \quad 1 \\
 \hline
 1 \quad | \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1
 \end{array}$$

Значение бита переноса равно значению знакового бита двоичного кода суммы, следовательно, переполнения нет.

Обратный код суммы в знаковом разряде содержит «1» («←»), тогда прямой код суммы это ПК( $SUM_2$ ) = 10100010, что соответствует  $SUM_{10} = -34$ .

Рассмотрим еще один пример. сложим числа  $A_{10} = -78$  ОК( $A_2$ ) = 110110001 и  $B_{10} = -59$  ОК( $B_2$ ) = 11100010 в формате 8 бит. При сложении для каждого числа введем еще один бит – бит переноса ( $C$ ), начальное значение этого разряда равен значению знакового разряда числа ( $S$ ):

$$\begin{array}{r}
 C \quad S \\
 +1 \mid 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \\
 1 \mid 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \\
 \hline
 1 \mid 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1
 \end{array}$$

При сложении возник перенос из бита  $C$ . По правилам сложения в обратных кодах этот перенос нужно прибавить к числу:

$$\begin{array}{r}
 C \quad S \\
 +1 \mid 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \\
 0 \mid 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \\
 \hline
 1 \mid 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1
 \end{array}$$

Значение бита переноса не равно значению знакового бита двоичного кода суммы, следовательно, произошло переполнение.

Правила сложения чисел в дополнительных кодах (перед началом операции числа должны быть представлены в дополнительных кодах):

- дополняем каждое число битом переноса, в который заносим значение, равное значению знакового разряда числа;
- выполняем операцию сложения по правилам двоичной арифметики, не выделяя знаковый бит и бит переноса;
- если при вычислении суммы появился перенос из старшего разряда, то его нужно отбросить;
- если значение бита переноса суммы не равно значению знакового разряда, то произошло переполнение, в этом случае числового значения в ответе писать не нужно;
- если значение бита переноса суммы равно значению знакового разряда, то получен дополнительный код суммы.

Например, сложим числа  $A_{10} = -5$  ДК( $A_2$ ) = 11111011 и  $B_{10} = -29$  ДК( $B_2$ ) = 11100011 в формате 8 бит. При сложении для каждого числа введем еще один бит – бит переноса ( $C$ ), начальное значение этого разряда равен значению знакового разряда числа ( $S$ ):

$$\begin{array}{r|cccccccc}
 C & S & & & & & & & & \\
 +1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & \\
 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & \\
 \hline
 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 
 \end{array}$$

При сложении возник перенос из бита  $C$ . По правилам сложения в дополнительных кодах этот перенос нужно отбросить.

Значение бита переноса равно значению знакового бита двоичного кода суммы, следовательно, переполнения нет.

Дополнительный код суммы в знаковом разряде содержит «1» («←»), тогда прямой код суммы это  $ПК(SUM_2) = 10100010$ , что соответствует  $SUM_{10} = -34$ .

Рассмотрим еще один пример. сложим числа  $A_{10} = -78$  ДК( $A_2$ ) = 110110010 и  $B_{10} = -59$  ДК( $B_2$ ) = 111000101 в формате 8 бит. При сложении для каждого числа введем еще один бит – бит переноса ( $C$ ), начальное значение этого разряда равен значению знакового разряда числа ( $S$ ):

$$\begin{array}{r|cccccccc}
 C & S & & & & & & & & \\
 +1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & \\
 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & \\
 \hline
 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 
 \end{array}$$

При сложении возник перенос из бита  $C$ . По правилам сложения в дополнительных кодах этот перенос нужно отбросить.

Значение бита переноса не равно значению знакового бита двоичного кода суммы, следовательно, произошло переполнение.

## Задание

### Вариант 1

Используя стандартные средства табличного процессора (таблица 3), разработать инструмент сложения целых чисел  $A$  и  $B$ , представленных в прямых кодах.

Требования к выполнению задания.

– на листе табличного процессора должна быть отведена ячей-

ка для ввода двоичного числа  $A$ , представленного в прямом коде;

- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $B$ , представленного в прямом коде;

- на листе табличного процессора должна быть отведена ячейка для вывода суммы в прямом коде;

- ячейки должны содержать комментарии, поясняющие их назначение;

- длина числа  $A$  не должна превосходить 8 разрядов, длина числа  $B$  не должна превосходить 8 разрядов;

- если операция не может быть выполнена, должно появляться соответствующее сообщение, если в результате выполнения операции произошло переполнение, должно появиться соответствующее сообщение.

#### Вариант 2

Используя стандартные средства табличного процессора (таблица 3), разработать инструмент вычитания целых чисел  $A$  и  $B$ , представленных в прямых кодах.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $A$ , представленного в прямом коде;

- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $B$ , представленного в прямом коде;

- на листе табличного процессора должна быть отведена ячейка для вывода разности в прямом коде;

- ячейки должны содержать комментарии, поясняющие их назначение;

- длина числа  $A$  не должна превосходить 8 разрядов, длина числа  $B$  не должна превосходить 8 разрядов;

- если операция не может быть выполнена, должно появляться соответствующее сообщение, если в результате выполнения операции произошло переполнение, должно появиться соответствующее сообщение.

#### Вариант 3

Используя стандартные средства табличного процессора (таблица 3), разработать инструмент сложения целых чисел  $A$  и  $B$ , представленных в обратных кодах.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $A$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $A$ , преобразованного в обратный код;
- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $B$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $B$ , преобразованного в обратный код;
- на листе табличного процессора должна быть отведена ячейка для вывода суммы в обратном коде;
- на листе табличного процессора должна быть отведена ячейка для вывода суммы, преобразованной в прямой код;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина числа  $A$  не должна превосходить 8 разрядов, длина числа  $B$  не должна превосходить 8 разрядов;
- если операция не может быть выполнена, должно появляться соответствующее сообщение, если в результате выполнения операции произошло переполнение, должно появиться соответствующее сообщение.

#### Вариант 4

Используя стандартные средства табличного процессора (таблица 3), разработать инструмент вычитания целых чисел  $A$  и  $B$ , представленных в обратных кодах.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $A$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $A$ , преобразованного в обратный код;
- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $B$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $B$ , преобразованного в обратный код;

- на листе табличного процессора должна быть отведена ячейка для вывода разности в обратном коде;
- на листе табличного процессора должна быть отведена ячейка для вывода разности, преобразованной в прямой код;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина числа  $A$  не должна превосходить 8 разрядов, длина числа  $B$  не должна превосходить 8 разрядов;
- если операция не может быть выполнена, должно появляться соответствующее сообщение, если в результате выполнения операции произошло переполнение, должно появиться соответствующее сообщение.

#### Вариант 5

Используя стандартные средства табличного процессора (таблица 3), разработать инструмент сложения целых чисел  $A$  и  $B$ , представленных в дополнительных кодах.

#### Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $A$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $A$ , преобразованного в дополнительный код;
- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $B$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $B$ , преобразованного в дополнительный код;
- на листе табличного процессора должна быть отведена ячейка для вывода суммы в дополнительном коде;
- на листе табличного процессора должна быть отведена ячейка для вывода суммы, преобразованной в прямой код;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина числа  $A$  не должна превосходить 8 разрядов, длина числа  $B$  не должна превосходить 8 разрядов;
- если операция не может быть выполнена, должно появляться соответствующее сообщение, если в результате выполнения опера-

ции произошло переполнение, должно появиться соответствующее сообщение.

#### Вариант 6

Используя стандартные средства табличного процессора (таблица 3), разработать инструмент вычитания целых чисел  $A$  и  $B$ , представленных в дополнительных кодах.

Требования к выполнению задания.

- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $A$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $A$ , преобразованного в дополнительный код;
- на листе табличного процессора должна быть отведена ячейка для ввода двоичного числа  $B$ , представленного в прямом коде;
- на листе табличного процессора должна быть отведена ячейка для отображения двоичного числа  $B$ , преобразованного в дополнительный код;
- на листе табличного процессора должна быть отведена ячейка для вывода разности в дополнительном коде;
- на листе табличного процессора должна быть отведена ячейка для вывода разности, преобразованной в прямой код;
- ячейки должны содержать комментарии, поясняющие их назначение;
- длина числа  $A$  не должна превосходить 8 разрядов, длина числа  $B$  не должна превосходить 8 разрядов;
- если операция не может быть выполнена, должно появляться соответствующее сообщение, если в результате выполнения операции произошло переполнение, должно появиться соответствующее сообщение.

#### Общие требования

При оформлении работы в Excel необходимо использовать шрифт Times New Roman, размер 14, начертание обычное. Размеры ячеек выбираются по контексту, каждая ячейка должна иметь пояснения, какая информация в ней находится.

Составить отчет по результатам выполнения лабораторной работы. Отчет должен содержать:

- титульный лист (Приложение А);

- Содержание;
- Цель работы;
- Задание;
- Словесный подробный алгоритм выполнения задания с указанием функций процессора, используемых для выполнения конкретного действия;
- Скриншот интерфейса инструмента преобразования чисел;
- Скриншот листа выполнения задания с отображением формул и с отображением результатов вычислений;
- Выводы по работе.

Номера страниц проставляются внизу страницы по правому краю (шрифт – Times New Roman 12 пт). Первой страницей является титульный лист, нумерация проставляется, начиная с Цели работы отчета.

Макет Оглавления приведен в Приложении Б.

Для оформления заголовков разделов следует использовать стиль «Заголовок 1». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 12 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

Для оформления текста отчета необходимо использовать стиль «Обычный». Параметры стиля:

- название шрифта – Times New Roman;
- размер шрифта – 14 пт;
- междустрочный интервал – полуторный;
- интервалы перед и после абзаца – 0 пт;
- выравнивание – по ширине;
- отступы слева и справа - 0 см;
- отступ первой строки – 1,25 см.

## Справка по функциям табличного процессора

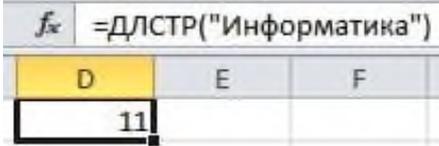
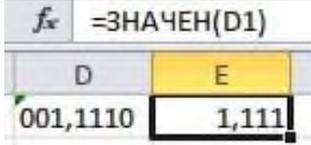
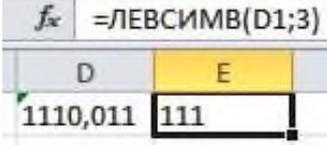
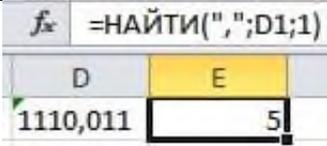
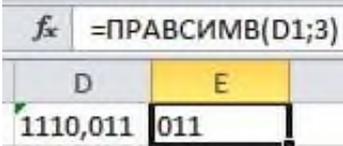
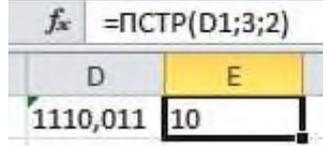
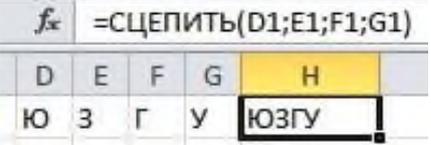
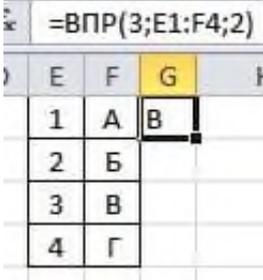
функция	назначение	пример использования
1	2	3
ДЛСТР(текст)	возвращает количество знаков в текстовой строке	
ЗНАЧЕН(текст)	преобразует текстовый аргумент в число	
ЛЕВСИМВ(текст; количество_знаков)	возвращает указанное количество знаков с начала строки текста	
НАЙТИ(искомый_текст; просматриваемый_текст; начальная позиция)	Возвращает позицию начала искомой строки текста в содержащей ее строке текста. Прописные и строчные буквы различаются	
ПРАВСИМВ(текст; число_знаков)	возвращает указанное число знаков с конца строки текста	
ПСТР(текст; начальная_позиция; количество_знаков)	возвращает заданное число знаков из строки текста, начиная с указанной позиции	
СЦЕПИТЬ(текст1; текст2;...)	объединяет несколько текстовых строк в одну	

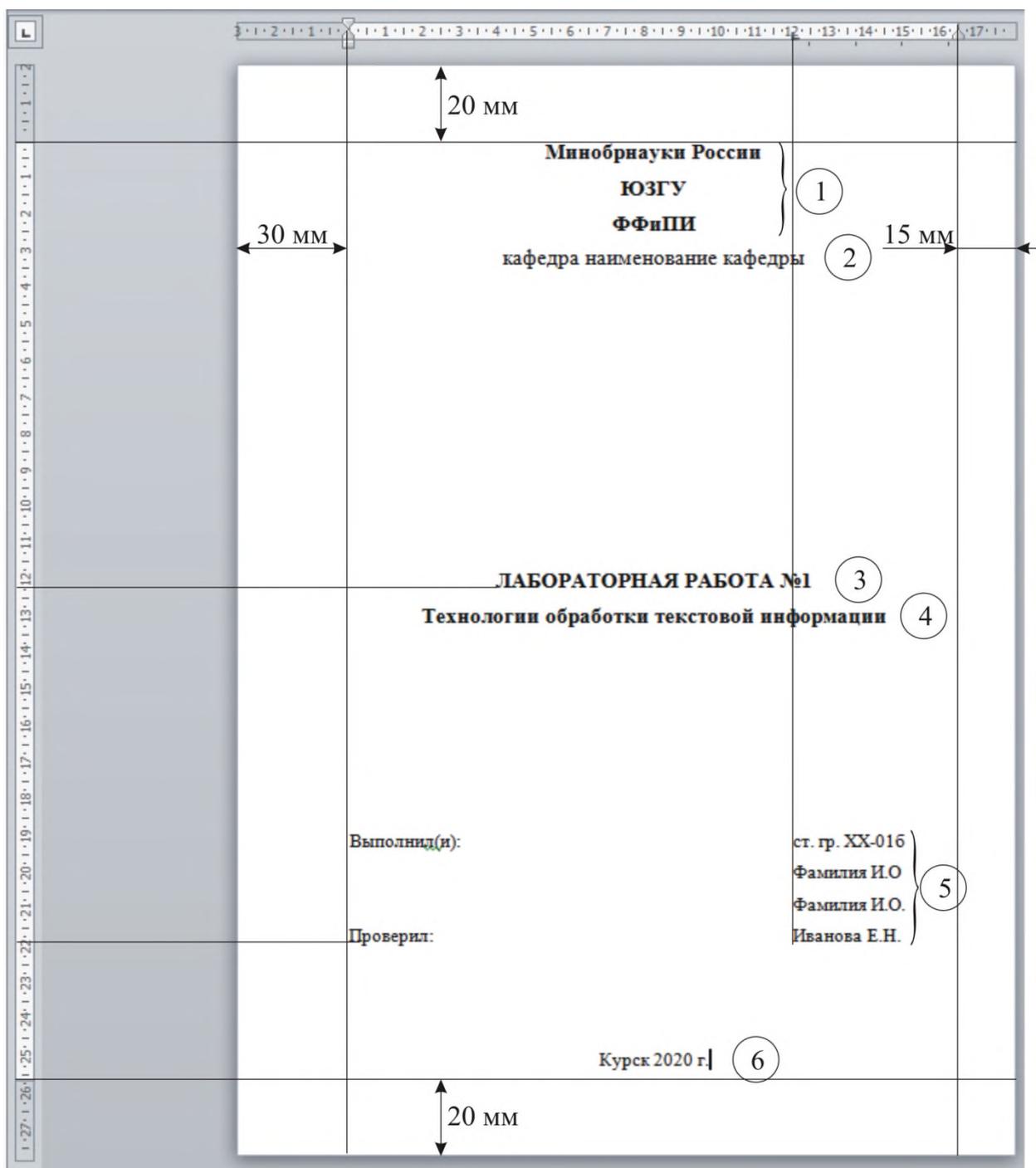
Таблица 3 (продолжение)

1	2	3																				
ВПР(искомое_значение; таблица; номер_столбца; ...)	ищет значение в крайнем левом столбце таблицы и возвращает значение ячейки, находящейся в указанном столбце той же строки	 <p>Formula: =ВПР(3;E1:F4;2)</p> <table border="1" data-bbox="1093 380 1356 604"> <thead> <tr> <th></th> <th>E</th> <th>F</th> <th>G</th> </tr> </thead> <tbody> <tr> <th>1</th> <td>А</td> <td></td> <td>В</td> </tr> <tr> <th>2</th> <td>Б</td> <td></td> <td></td> </tr> <tr> <th>3</th> <td>В</td> <td></td> <td></td> </tr> <tr> <th>4</th> <td>Г</td> <td></td> <td></td> </tr> </tbody> </table>		E	F	G	1	А		В	2	Б			3	В			4	Г		
	E	F	G																			
1	А		В																			
2	Б																					
3	В																					
4	Г																					
ГПР(искомое_значение; таблица; номер_строки; ...)	ищет значение в верхней строке таблицы и возвращает значение ячейки, находящейся в указанной строке того же столбца	 <p>Formula: =ГПР(3;E1:H2;2)</p> <table border="1" data-bbox="1069 772 1380 918"> <thead> <tr> <th></th> <th>E</th> <th>F</th> <th>G</th> <th>H</th> <th>I</th> </tr> </thead> <tbody> <tr> <th>1</th> <td>2</td> <td>3</td> <td>4</td> <td></td> <td>В</td> </tr> <tr> <th>2</th> <td>А</td> <td>Б</td> <td>В</td> <td>Г</td> <td></td> </tr> </tbody> </table>		E	F	G	H	I	1	2	3	4		В	2	А	Б	В	Г			
	E	F	G	H	I																	
1	2	3	4		В																	
2	А	Б	В	Г																		

### Список использованных источников

1. Колокольникова, А.И. Информатика [Текст] : учебное пособие / А.И. Колокольникова. – Москва ; Берлин : Директ-Медиа, 2020. – 289 с. : ил., табл. – Режим доступа: URL: <https://biblioclub.ru/>
2. Информатика [Текст] : лабораторный практикум / авт.-сост. О.В. Вельц. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2018. – 117 с. – Режим доступа: URL: <https://biblioclub.ru/index.php?page=book&id=494706>.

## Оформление титульного листа отчета по лабораторной работе



Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 16 пт, начертание обыч-

ное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

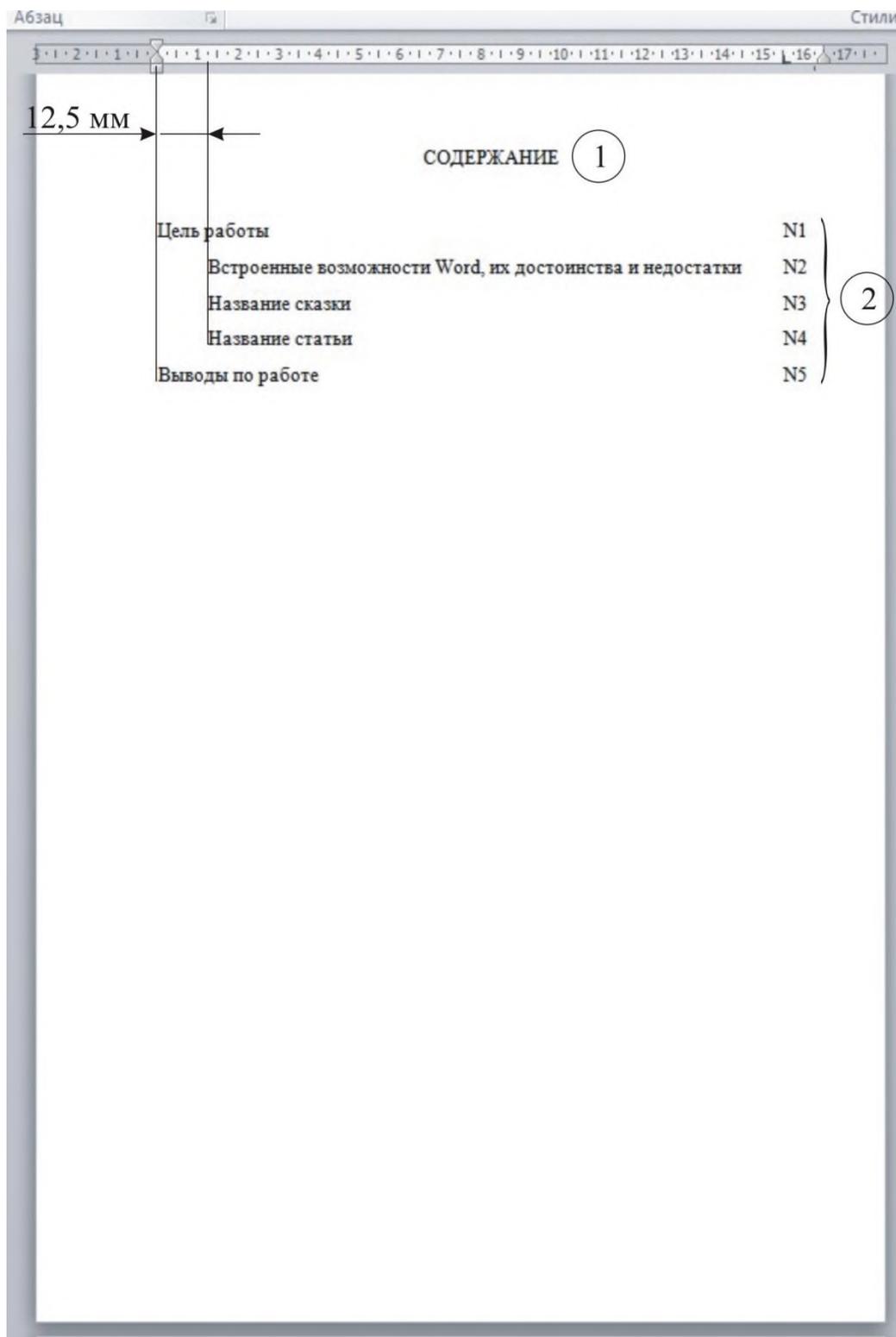
3: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

4: Шрифт Times New Roman, размер 16 пт, начертание полужирное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

5: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по ширине, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки, позиция табуляции 12 см.

6: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

## Оформление Содержания отчета по лабораторной работе



Примечание: в кружках обозначены номера наборов параметров форматирования текста.

1: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет

черный, все прописные. Выравнивание по центру, отступы слева 0, справа 0, красной строки нет, интервалы перед 0, после 0, междустрочный 1,5 строки.

2: Шрифт Times New Roman, размер 14 пт, начертание обычное, цвет черный, видоизменений нет. Выравнивание по левому краю, отступы слева 0, справа 0, интервалы перед 0, после 0, междустрочный 1,5 строки.

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра вычислительной техники

УТВЕРЖДАЮ

Проректор по учебной работе

О.Г. Локтионова

« 16 » 03 2023 г.



**Основы создания сайтов с помощью языка HTML**

Методические указания по выполнению практических и лабораторных работ по дисциплине «Информатика» для направлений подготовки 09.03.00, 10.03.00, 11.03.00, 12.03.00

Курск 2023

УДК 004.4

Составитель Е.А. Коломиец

Рецензент

Кандидат технических наук, доцент Л.В. Стародубцева

**Основы создания сайтов с помощью языка HTML:** методические указания по выполнению практических и лабораторных работ по дисциплине «Информатика» / Юго-Зап. гос. ун-т; сост. Е.А. Коломиец. Курск, 2023. 27 с. Библиогр.: с. 27.

Описываются основы использования языка разметки HTML и создания web-страниц с применением форматирования на основе HTML. Изложены краткие теоретические сведения из теории разработки сайтов, приведены примеры решения задач, а также задачи для самостоятельного решения.

Методические рекомендации предназначены для студентов, обучающихся по направлениям подготовки 09.03.00, 10.03.00, 11.03.00, 12.03.00.

Текст печатается в авторской редакции.

Подписано в печать . Формат 60x84 1/16.

Усл. печ. л. 1,57 п.л. Уч.-изд. л. 1,21. Тираж 100 экз. Заказ. 416 Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

## **Цель работы**

Познакомиться с основами языка разметки HTML и научиться создавать собственные web-страницы с применением форматирования на основе HTML.

## **Задачи лабораторной работы**

После выполнения работы студент должен:

- 1) знать понятия: "HTML", "браузер", "гиперссылка", "тег";
- 2) уметь применять основные теги для оформления web-страниц;
- 3) знать назначение &-последовательностей, комментариев в языке HTML;
- 4) научиться создавать и оформлять web-страницы.

## **Перечень обеспечивающих средств**

Для обеспечения выполнения работы необходимо иметь компьютер с операционной системой и методические указания по выполнению работы.

## **Общие теоретические сведения**

Внимательно ознакомьтесь с теоретическим материалом. Дальнейшие задания будут ссылаться на представленный материал.

### **1. Понятие об HTML**

Гипертекстовые документы создаются на специальном языке, который называется HTML (Hyper Text Markup Language - язык разметки гипертекста). Разработка этого стандарта позволила снять две существенные проблемы, которые возникали у пользователей больших сетей.

Первая из них - время доставки информации, оно существенно зависит от объема передаваемого файла. В стандарте HTML вводится иерархическая структура документов: вся информация по некоторой теме разбивается на небольшие смысловые блоки и посылается пользователю только по мере возникновения необходимости в ней. Это существенно сокращает время ожидания ответа на возникший вопрос и делает работу в сети более комфортной и дешевой. Для организации такого режима работы в языке HTML введено средство *гиперссылка* (см. п. 10).

Вторая проблема - распознавание пришедшей по запросу информации. Способы кодировки одних и тех же сведений в разных текстовых редакторах часто бывают разными. И если компьютер-получатель оснащен не тем текстовым редактором, на котором создавался документ, он не всегда может интерпретировать пришедшую на него информацию. Чтобы преодолеть это, в языке HTML установлены специальные правила.

1. Весь документ набирается только текстовыми символами (их коды понятны всем редакторам), т. е. его можно набрать в любом текстовом редакторе (например, "Блокнот", "WordPad", "NotePad++").

2. Символы и конструкции, которые непонятны *браузеру* (программе, позволяющей просматривать web-страницы), игнорируются (а текстовые редакторы обычно прекращают интерпретировать документ, если встречают в

нем непонятные коды).

3. В текст документа можно добавлять разнообразные команды, которые указывают, как следует изображать на экране тот или иной фрагмент текста. Эти команды называют *тегами*. Они так же, как и документ, набираются текстовыми символами, но отделяются от него угловыми скобками: < >.

4. Файлы с HTML-кодом должны помечаться расширением .html или .htm.

## 2. Структура тегов

Каждый тег обязательно содержит пару угловых скобок < >, между которыми находится *ключевое слово*, обозначающее основную функцию тега. Например, ключевое слово HR в теге <HR> означает, что тег вставляет в документ серую горизонтальную линию (Horizontal Row) по всей ширине страницы и толщиной в один пиксел.

Действие тега детализируется с помощью *атрибутов*. Например, применительно к тегу <HR> можно уточнить, какой длины, толщины и цвета должна быть линия. Каждому ключевому слову сопоставляется свой собственный набор атрибутов и стандартные значения для них.

Атрибуты размещают после ключевого слова и перечисляют в произвольном порядке через пробел. После каждого атрибута ставят знак = и нужное значение. Например, тег <HR WIDTH=50% SIZE=4> означает, что в документ следует вставить горизонтальную линию, которая должна доходить только до середины строки и иметь ширину четыре пиксела. Значения атрибутов в некоторых случаях полагается заключать в двойные кавычки.

Атрибуты, соответствующие одним и тем же свойствам, в разных тегах часто, но не всегда, обозначаются одинаковыми словами. Например, выравнивание текста в пределах строки, ячейки таблицы или около картинки задается одинаковым словом ALIGN. А свойство «цвет шрифта» для документа в целом задается атрибутом TEXT, но для отдельно взятого фрагмента документа - атрибутом COLOR. Соответствующие теги, задающие темно-синий цвет шрифта, выглядят так:

<BODY TEXT=NAVY>, <FONT COLOR=NAVY>.

Многие теги являются парными. Открывающий тег включает какое-нибудь действие, закрывающий прекращает его. В этом случае ключевое слово у обоих тегов одинаковое, но в открывающем теге дополнительно перечисляются все необходимые атрибуты, а в закрывающем атрибутов нет, и перед ключевым словом ставится знак /. Парные теги называются *контейнерами*. Например, контейнер <B>Я изучаю язык HTML</B> означает, что текст «Я изучаю язык HTML» следует изобразить жирным шрифтом, а дальнейшие символы - так, как до него.

Замечания:

1) в ключевых словах и атрибутах строчные и прописные регистры букв не различаются;

2) URL файлов-ссылок заключаются в двойные кавычки, и регистры букв в них учитываются;

3) в последовательности пробелов, идущих подряд, учитывается только

первый, остальные игнорируются;

4) признак начала новой строки игнорируется. Поэтому при составлении кода можно клавишей <Enter> разрывать его на строки так, как код удобно просматривать и редактировать. На виде документа в окне браузера это не отражается. Если нужно сохранить при просмотре через браузер введенные в код HTML пробелы и разрывы строк, следует воспользоваться тегами компоновки текста (см. п. 6).

Списки основных тегов, их функций и атрибутов приведены в табл. 1, 2, 3. Пояснения по основным группам тегов - в п.п. 4-10.

Таблица 1. Основные теги форматирования и компоновки текста

Теги	Действие
<B>...</B>	<b>Полужирный шрифт</b>
<I>...</I>	Курсив
<BIG>...</BIG>	Более крупный шрифт
<SMALL>...</SMALL>	Более мелкий шрифт
<SUB>...</SUB>	Нижний <small>индекс</small>
<SUP>...</SUP>	Верхний <small>индекс</small>
<H1> ... </H1>	Самый крупный размер заголовка
<H2> ... </H2>, ... <H5>...</H5>	<b>Промежуточные размеры заголовков</b>
<H6>...</H6>	<b>Самый мелкий размер заголовка</b>
 	Переход на новую строку
<P>...</P>	Новый абзац
<HR>	Горизонтальная линия
<PRE>...</PRE>	Компоновка текста, как в коде
<OL>...</OL>	Нумерованный список
<UL>...</UL>	Маркированный список
<LI>	Элемент списка
<TABLE> ... </TABLE>	Таблица
<CAPTION> ... </CAPTION>	Заголовок таблицы
<TR>...</TR>	Строка таблицы
<TH>... </TH>	Ячейка заголовка
<TD>...</TD>	Ячейка данных

Таблица 2. Теги внесения ссылок на дополнительную информацию

Теги	Действие
<A HREF="URL файла-ссылки">.. ,</A>	Ссылка на текстовый файл или рисунок
<A NAME "#имя метки">.. ,</A>	Установка метки внутри файла
<IMG SRC="URL файла-изображения">	Вставка изображений

Таблица 3. Основные атрибуты тегов

Атрибут	Возможные значения	Действие атрибута	В каких тегах используется
COLOR=	GRAY (серый), AQUA (аквамарин), BLACK (черный), BLUE (синий),	Задает цвет линий и шрифта в тексте или таблице	<HR>, <FONT>
BGCOLOR=	FUCHSIA (яркий пурпурно-красный), GREEN (зеленый), LIME (зеленоватый),	Задает цвет фона	<TABLE>, <TR>, <TH>, <BODY>
BORDERCOLOR=	MAROON (темнобордовый), NAVY (темносиний), OLIVE (оливковый), PURPLE (пурпурный), RED (красный),	Задает цвет внешнего контура таблицы	<TABLE>
TEXT=	SILVER (серебристый), TEAL, YELLOW (желтый), WHITE (белый). Возможны также те же слова с приставками LIGHT и DARK,	Задает цвет шрифта в документе в целом	<BODY>
LINK=, VLINK=, ALINK=	например LIGHTGREEN (светлозеленый), DARKBLUE (темно-голубой)	Цвета соответственно непосещенных, посещенных и активных ссылок	<BODY>
BACKGROUND=	"URL" файла с изображением для фона	Создает фон-картинку	<TABLE>, <TH>, <TD>, <BODY>
BORDER=	Целое число без размерности	Задает толщину окантовки для изображения или таблицы	<IMG>, <TABLE>
ALIGN=	LEFT, CENTER, RIGHT	Горизонтальное выравнивание текстового фрагмента или таблицы в целом	<P>, <H1>, <H2>, ..., <H6>, <TABLE>, <HR>, <TH>, <TD>
ALIGN=	BOTTOM, TOP	Размещение заголовка над или под таблицей	<CAPTION>
ALIGN=	BOTTOM, LEFT, RIGHT, TOP, MIDDLE	Размещение текстового пояснения к картинке	<IMG>
VALIGN=	MIDDLE, TOP, BOTTOM	Вертикальное выравнивание фрагмента	<TABLE>, <TH>, <TD>

Атрибут	Возможные значения	Действие атрибута	В каких тегах используется
WIDTH=	Целое число без размерности или со знаком %	Длина/высота фрагмента в пикселах или в процентах от ширины/высоты окна. Для всех ячеек, находящихся в строке/столбце, используется максимальное значение из заданных в ее/его ячейках	<TABLE>, <HR>, <TH>, <TD>, <IMG>
HEIGHT=			
SIZE=	Целое число без размерности (по умолчанию 1)	Толщина линии, размер шрифта	<HR>, <FONT>
TYPE=	1, A, a, i, I	Тип нумерации элементов упорядоченного списка	<OL>
START=	Номер первого элемента в выбранном типе нумерации		

Примечание. Первое из перечисленных значений атрибутов тегов - значение по умолчанию.

### 3. &-последовательности, комментарии

Иногда в документе приходится использовать такие знаки, которые имеют в языке HTML специальный смысл. Например, двойные кавычки " " выделяют значение атрибута, и, если их использовать вне тега, браузер не поймет конструкцию и пропустит ее. Чтобы этого не произошло, для отображения служебных символов используют специальные последовательности знаков, начинающиеся с символа & и заканчивающиеся знаком ; (точка с запятой) - так называемые *&-последовательности*. Основные из них:

- последовательность &lt; (от английского "less than") - вместо символа < (меньше);
- последовательность &gt; (от "greater than") - вместо символа > (больше);
- последовательность &quot; (от "quotations mark") - вместо символа " (двойные кавычки);
- последовательность &amp; (от "ampersand") - вместо символа &;
- последовательность &nbsp; (от nonbreaking space - неразрывный пробел) - вместо пробела, который не следует игнорировать при выдаче документа в окно браузера. Эта &-последовательность как бы вставляет в текст невидимую глазу букву, которую браузер воспринимает как реальную и оставляет для нее место на экране. Она используется для заполнения пустых ячеек таблицы, пустых абзацев, для склеивания слов (если набрать TOP&nbsp;100, то у вас никогда не получится, что TOP останется в одной строке, а 100 перенесется на следующую, например,

при изменении шрифта или размера окна браузера).

Внимание: &-последовательности должны набираться только в нижнем регистре, точка с запятой в конце последовательности обязательна. Пример: чтобы пользователь увидел на экране слова крейсер "Аврора", в HTML-документе должен быть следующий код:

крейсер &quot;Аврора&quot;;

Чтобы код HTML было легче понимать и править, полезно вводить в него *комментарии*, которые были бы видны только при работе с кодом, но не отражались бы в окне браузера при просмотре документа. Для этой цели используют специальный контейнер, написанный не по стандартным правилам. Он выглядит так: `<!-- ... -->`

Примеры:

`<!-- Это мой комментарий -->`

`<!-- А этот комментарий занимает в тексте несколько строк -->`

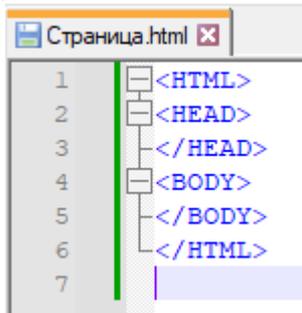
Внимание: в текст комментария не допускается вставлять два и более дефисов подряд: `--` или `----` (разные браузеры по-разному реагируют на это).

#### 4. Структура документа HTML

Каждый HTML-документ состоит из трех главных частей.

1. Объявление HTML-кода - контейнер `<HTML>...</HTML>`
2. Заголовок документа - контейнер `<HEAD> ... </HEAD>`
3. Тело документа - контейнер `<BODY> ... </BODY>`

Заголовок и тело документа вкладываются внутрь объявления HTML по следующей схеме:



В заголовок помещаются теги, определяющие информацию о документе в целом. Наиболее употребительный из них - контейнер `<TITLE>`:

`<TITLE>` Краткая расшифровка содержания документа (не более 64 символов) `</TITLE>`.

Текст, помещенный в `<TITLE>`, заменяет системную информацию в заголовке браузера. При просмотре большого количества файлов это облегчает ориентировку среди них.

В разделе «тело документа» размещается содержание документа, которое выдается в рабочее окно браузера. Атрибуты тега `<BODY>` задают следующие свойства:

- TEXT - цвет текста там, где он не указан специальными средствами. По умолчанию черный.
- BGCOLOR - цвет фона. По умолчанию белый.
- BACKGROUND - фоновое изображение (аналогично рисунку на Рабочем

столе). Значением является URL файла-изображения.

- LINK, VLINK, ALINK - соответственно цвета непосещенных, посещенных и активных в данный момент гиперссылок.

Примеры:

```
<BODY TEXT=SILVER BGCOLOR=NAVY>
```

```
<BODY TEXT=SILVER BACKGROUND="море.jpg">
```

Первый тег задает для базового оформления документа светло-серый шрифт на темно-синем фоне. Все отступления от этого стандарта в дальнейшем надо будет оговаривать дополнительными тегами внутри документа. Вторым в качестве фона использует изображение, находящееся в файле море.jpg, расположенном в том же каталоге, что и вызывающий его документ (подробнее см. п. 8). Цвет шрифта так же, как и в первом примере, - светло-серый.

Основные теги, определяющие вид документа в окне браузера, рассмотрены ниже и приведены в виде справочника в табл. 1, 2, 3.

## 5. Теги форматирования шрифта

В HTML существуют два подхода к шрифтовому оформлению текста - так называемые физические и логические стили. Здесь рассматриваются только физические стили. Под ними подразумевают прямое указание браузеру на изменение текущего шрифта. Теги физических стилей - контейнерные. Например, между тегами <B> и </B> будет жирный шрифт (Bold), а между <I> и </I> - курсив (наклонный - Italic). Основные контейнеры физических стилей (рис.1):

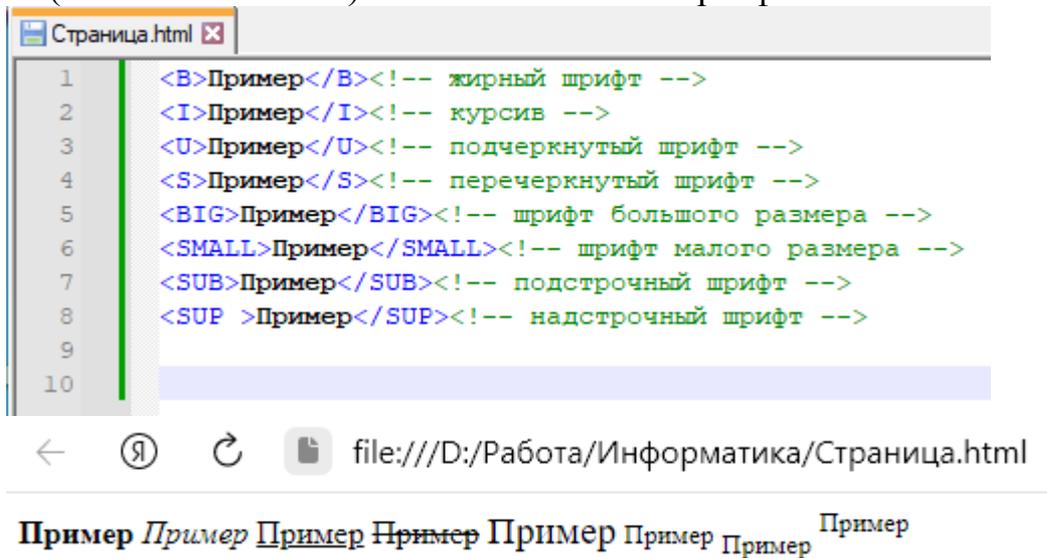


Рисунок 1 – Основные контейнеры физических стилей

Тег <FONT> - изменение типа, цвета и размера шрифта, допускаются атрибуты:

- SIZE - размер букв, возможные значения: 1, 2, 3, ..., 7. По умолчанию 3;
- COLOR - цвет шрифта. Можно указывать шестнадцатеричными кодами или английскими словами. По умолчанию BLACK;
- FACE - изменение типа шрифта, возможные значения: ARIAL, COURIER и т. д. По умолчанию VERDANA.

Теги могут комбинироваться, но при этом надо следить, чтобы контейнеры

имели вложенную структуру (контейнер, открытый позже, должен быть закрыт до того, как будет закрыт внешний).

Помимо перечисленных выше тегов существуют шесть контейнерных тегов, удобных для оформления заголовков: `<H1>...</H1>`, `<H2>...</H2>`, ..., `<H6>...</H6>`. Заголовки отличаются от текста и между собой толщиной и высотой букв (в заголовке `<H1>` самые крупные символы, в `<H6>` - самые мелкие), а также увеличением интервала по вертикали между заголовком и прочей информацией. Атрибут `ALIGN` управляет выравниванием заголовков по горизонтали. По умолчанию он принимает значение `LEFT`. Другие возможные значения: `CENTER`, `RIGHT`.

## 6. Теги выделения смысловых частей текста

Абзацы в тексте выделяются контейнером `<P>...</P>`. Атрибут `ALIGN` задает выравнивание по горизонтали. По умолчанию он принимает значение `LEFT`. Другие возможные значения: `CENTER`, `RIGHT`. При просмотре браузером абзацы отделяются друг от друга пустой строкой. Закрывающий тег `</P>` необязателен. Его можно использовать только тогда, когда было заказано нестандартное выравнивание текста (рис.2).

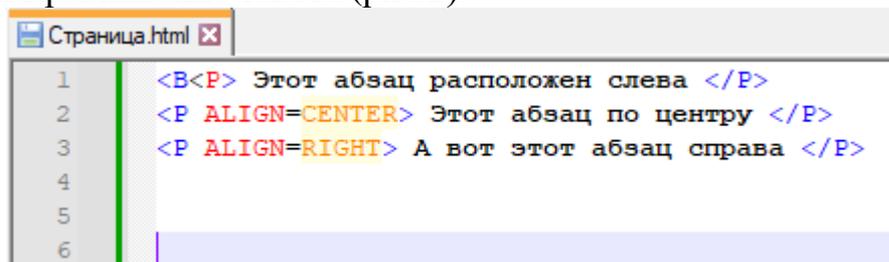


Рисунок 2 – Выравнивание текста

Чтобы перейти на новую строку, не вставляя пустую строку в текст, применяется тег `<BR>`. Закрывающего тега у него нет.

Иногда полезно разделять смысловые части документа горизонтальной чертой. Это делается с помощью одиночного тега `<HR>`. Его атрибуты:

`SIZE` - толщина линии в пикселах. По умолчанию 1 пиксел.

`WIDTH` - длина линии в пикселах или процентах от ширины экрана. По умолчанию 100%.

`ALIGN` - выравнивание линии, возможные значения: `LEFT`, `CENTER`, `RIGHT`. По умолчанию `LEFT`.

`COLOR` - цвет линии. По умолчанию `GRAY`.

`NOSHADA` - линия рисуется без тени. По умолчанию тень есть.

Пример. Код, в котором использованы теги `<BR>` и `<HR>`:

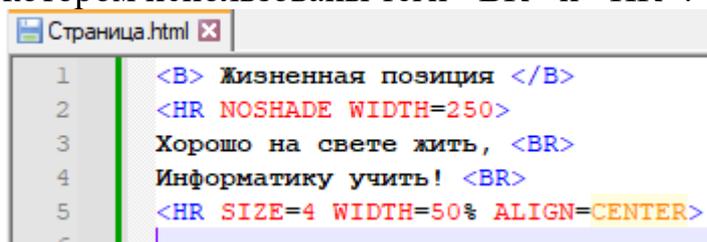


Рисунок 3 – Разделение текста горизонтальной чертой

## 7. Оформление списков

Списки облегчают нахождение нужной информации при перечислении сведений, объединенных общим смыслом: пункты меню, списки товаров, перечень сотрудников и т. п. В HTML предусмотрено несколько вариантов оформления списков, которые чаще всего используются в документации. Здесь будут рассмотрены только маркированные и нумерованные списки.

Для того чтобы оформить фрагмент документа в виде маркированного списка, его помещают в контейнер `<UL>... </UL>` (Unordered List). Для нумерованного - в контейнер `<OL> ... </OL>` (Ordered List). Затем текст в контейнере разбивается на отдельные пункты списка тегами `<LI>` (List Item - элемент списка). Закрывающий тег `</LI>` в конце каждого пункта ставить необязательно. В маркированном списке все пункты автоматически помечаются одинаковыми маркерами, в нумерованном - элементы нумерует сам браузер.

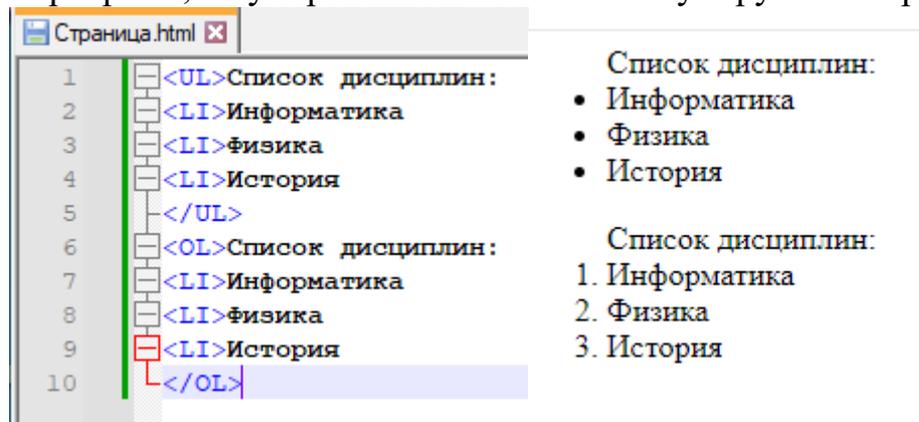


Рисунок 4 – Маркированный список

Можно организовать многоуровневые списки (разумно ограничиться тремя уровнями вложения). Для этого после каждого `<LI>` следует разместить контейнеры `<UL>... </UL>` или `<OL> ... </OL>` с разбивкой информации этого пункта на дополнительные подпункты.

В теге `<OL>` могут использоваться следующие атрибуты:

**TYPE** - определяет стиль оформления меток для элементов списка, подробнее ниже;

**START** - (только для нумерованного списка `<OL>`) определяет значение первого элемента списка, задается всегда числом (по умолчанию 1). В зависимости от значения атрибута **TYPE** этому числу сопоставляются разные символы, подробнее ниже;

**VALUE** - (только для метки `<LI>` списка `<OL>`) устанавливает номер текущего элемента списка. Как и в предыдущем случае, метка пункта может быть нечисловой, хотя значение атрибута - всегда число.

Для нумерованного списка в зависимости от стиля нумерации значением атрибута **TYPE** могут быть:

- 1 - арабские цифры (1,2,3,...)
- a - строчные буквы (a, b, c,...)
- A - прописные буквы (A,B,C,...)
- i - римские цифры строчные (i,ii,iii,...)

I - римские цифры прописные (I,II,III,...)

По умолчанию значения TYPE устанавливаются в зависимости от степени вложенности списка.

Число, указанное в атрибуте START, показывает порядковый номер элемента в ряду знаков, заданных в TYPE. С этого элемента следует начинать нумерацию пунктов списка.

## 8. Создание таблиц

Средства HTML для создания таблиц удобно использовать не только при представлении табличных данных. Их также применяют для точного выравнивания элементов на экране или комбинирования изображений и текстов. В качестве примера рассмотрим код, который создает простейшую таблицу (рис.5):

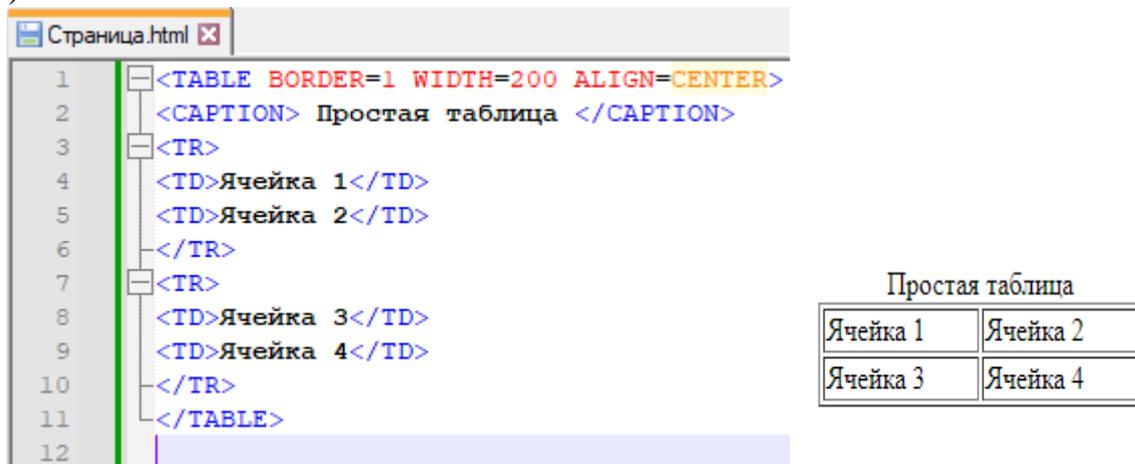


Рисунок 5 – Простая таблица

Как видно из примера, начало и конец таблицы оформляются тегами <TABLE> и </TABLE>. Все остальные теги, создающие структуру таблицы и ее содержание, вкладываются в этот контейнер. Для того, чтобы создать рамку вокруг таблицы, можно задать атрибут BORDER=n, где n - толщина рамки в пикселах. По умолчанию таблицы не имеют рамок, т.е. n = 0. Атрибут WIDTH=n%, где n - целое число, позволяет занять таблицей нужный процент от ширины экрана. Если знак процента опущен, то считается, что n задает размер таблицы в пикселах.

Сразу после тега <TABLE> можно вставить контейнер, создающий заголовок таблицы: <CAPTION> текст заголовка </CAPTION>. Заголовок по умолчанию располагается над таблицей слева. Другое выравнивание задается атрибутом ALIGN:

- LEFT - значение по умолчанию, заголовок над таблицей слева;
- TOP - заголовок над таблицей по центру;
- RIGHT - заголовок над таблицей справа;
- BOTTOM - заголовок под таблицей по центру.

Если таблица имеет сложную структуру и какие-то ячейки занимают несколько столбцов или строк, следует в тегах <TH> или <TD> добавить атрибуты соответственно COLSPAN=n или ROWSPAN=n, где n - количество

столбцов (строк), на которые растянута данная ячейка. Например, COLSPAN=3 означает, что ячейка по ширине будет простираться на 3 колонки, ROWSPAN=2 означает, что ячейка по высоте занимает две строки таблицы.

Пример. Фрагмент кода, создающего показанную слева таблицу (рис.6):

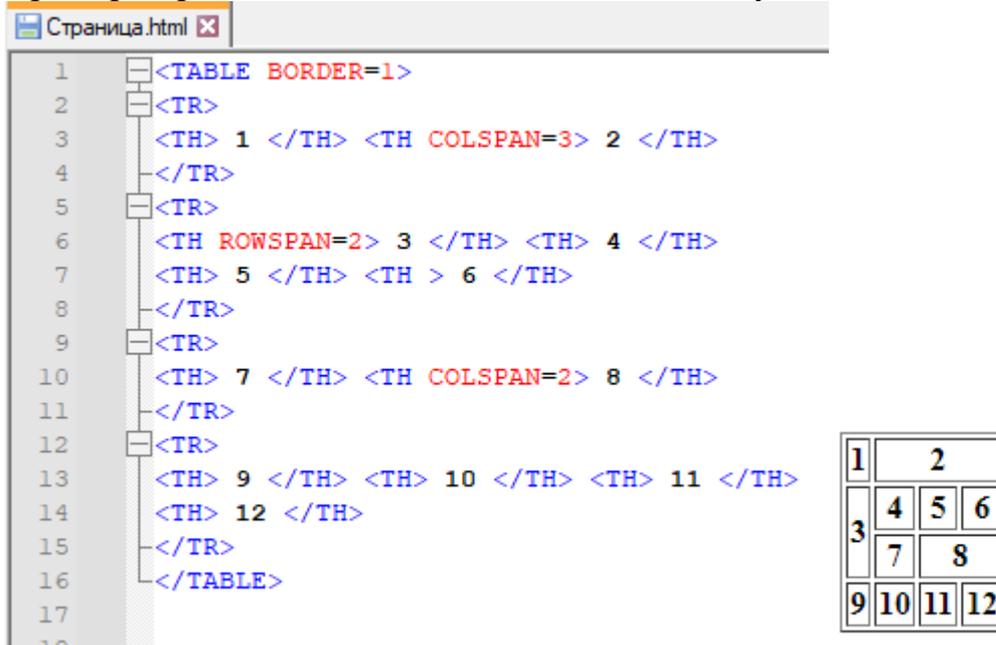


Рисунок 6 – Таблица данных

Таблица создается по строкам. Для каждой строки создается контейнер `<TR>...</TR>`, определяющий, какие ячейки и информация должны находиться в этой строке. Атрибут `ALIGN` со значениями `LEFT`, `CENTER` или `RIGHT` задает горизонтальное выравнивание во всех ячейках строки. Атрибут `VALIGN` со значениями `TOP`, `MIDDLE` или `BOTTOM` - вертикальное выравнивание (по умолчанию `MIDDLE` - по центру).

Для оформления ячеек используются два вида контейнеров: `<TH>...</TH>` (Table Head) и `<TD>...</TD>` (Table Data). Первый свою информацию выравнивает по центру и изображает жирным шрифтом, второй выравнивает по левому краю и использует обычный шрифт. Обычно `<TH>...</TH>` используют для ячеек с заголовками строк или столбцов, а `<TD>...</TD>` - для ячеек с данными таблицы. Можно также оставить ячейку пустой, но, чтобы она была отображена на экране, следует внести в нее последовательность `&nbsp;` (см. п. 3).

Для оформления данных, скомпонованных в виде таблицы, в теги `<TABLE>`, `<TR>`, `<TH>`, `<TD>` добавляют атрибуты.

Основные атрибуты тега `<TABLE>`:

`WIDTH` - определяет ширину таблицы в пикселах или процентах, по умолчанию ширина таблицы определяется содержимым ячеек;

`ALIGN` - определяет расположение таблицы в документе. По умолчанию таблица прижата к левому краю страницы. Допустимые значения атрибута: `LEFT`, `CENTER` и `RIGHT`;

`BORDER` - устанавливает толщину рамки в пикселах. По умолчанию - 0 (таблица рисуется без рамки);

**BORDERCOLOR** - устанавливает цвет рамки, по умолчанию черный. Цвет можно задавать английским словом или шестнадцатеричным числом;

**CELLSPACING** - определяет расстояние между рамками ячеек таблицы в пикселах;

**CELLPADDING** - определяет промежуток в пикселах между рамкой ячейки и текстом внутри нее;

**BGCOLOR** - устанавливает цвет фона для всей таблицы. По умолчанию белый. Цвет можно задавать английским словом или шестнадцатеричным числом;

**BACKGROUND** - заполняет фон таблицы изображением (см. п. 9). Значением атрибута является URL нужного файла.

Тег `<TR>`, открывающий строку таблицы, может иметь такие атрибуты:

**ALIGN** - устанавливает горизонтальное выравнивание текста во всех ячейках строки. Может принимать значения **LEFT** (по умолчанию), **CENTER** и **RIGHT**;

**VALIGN** - устанавливает вертикальное выравнивание текста в ячейках строки. Допустимые значения: **TOP** (выравнивание по верхнему краю), **MIDDLE** (выравнивание по центру - это значение принимается по умолчанию), **BOTTOM** (по нижнему краю);

**BGCOLOR** - устанавливает цвет фона для строки.

Ячейки таблицы начинаются тегами `<TH>` или `<TD>`. Для них предусмотрены следующие атрибуты:

**ALIGN** - устанавливает горизонтальное выравнивание текста в ячейке. Может принимать значение **LEFT**, **CENTER** и **RIGHT**;

**VALIGN** - устанавливает вертикальное выравнивание текста в ячейке. Допустимые значения: **TOP**, **CENTER** (это значение принимается по умолчанию), **BOTTOM**;

**WIDTH** - определяет ширину ячейки в пикселах;

**HEIGHT** - определяет высоту ячейки в пикселах;

**NOWRAP** - присутствие этого атрибута показывает, что текст должен размещаться в одну строку;

**BGCOLOR** - устанавливает цвет фона ячейки;

**BACKGROUND** - заполняет фон ячейки изображением. Значением атрибута является URL нужного файла.

## 9. Использование изображений

Внесение в документ изображений, как правило, существенно улучшает его внешний вид и делает его привлекательным для просмотра. Файлы рисунков можно использовать как фон для основной информации или же включать в документ как самостоятельные объекты. В Интернете обычно используют изображения в формате GIF и JPEG (файлы с расширениями `.gif` и `.jpg`). Современные браузеры понимают и другие форматы графических файлов (например, `.png`, `.svg`). Все они хорошо сжимают изображение, но в то же время прилично сохраняют его цветовую гамму и детали.

Чтобы использовать рисунок в качестве фона, применяют атрибут

BACKGROUND. Его можно указывать в теге <BODY> (см. п. 3) и табличных тегах <TABLE>, <TR>, <TH>, <TD> (см. п. 8). Если размер рисунка больше размера площади, которую следует закрыть фоном, то берется левая верхняя часть изображения, если меньше - рисунок повторяется, пока вся отведенная под фон площадь не будет закрыта.

Рисунок в качестве самостоятельного объекта (логотип фирмы, фотография и т. п.) вставляется тегом <IMG> (IMaGe - изображение). Обязательный атрибут - SRC (SouRCe - источник). Значением этого атрибута является URL файла с изображением.

URL обязательно надо выделять двойными кавычками и следить, чтобы пробелы и регистры букв в названиях папок и файлов в точности соответствовали оригиналу.

Примеры.

<IMG SRC="picture.gif"> - рисунок picture.gif находится в том же каталоге, что и текущий документ;

<IMG SRC="images/picture.gif"> - из текущего каталога перейти в подкаталог images и взять файл picture.gif оттуда;

<IMG SRC="../images/picture.gif"> - подняться в родительский каталог, оттуда перейти в каталог images и взять там рисунок picture.gif;

<IMG SRC "http://www.fortunecity.com/business/artics/practic/myphoto.gif"> - указание полного пути к файлу. Обычно применяется, если рисунок находится на другом сервере или используется файл с другого сайта.

Остальные атрибуты тега <IMG>, перечисленные ниже, оговаривают размеры и положение рисунка на странице, а также то, где должен располагаться поясняющий текст:

BORDER=n - рамка вокруг рисунка, где n - ее толщина в пикселах;

WIDTH=n - задается ширина изображения в пикселах или в процентах от ширины экрана браузера (тогда после n следует знак %);

HEIGHT=n - высота изображения в пикселах или в процентах от высоты экрана;

HSPACE=n - отступ слева и справа от картинке шириной в n пикселей (т.е. свободное пространство между рисунком и текстом или чем-то иным);

VSPACE=n - вертикальный отступ от картинке в пикселах;

ALIGN=БОТТОМ - по умолчанию. Сопровождающий текст выравнивается по нижнему краю рисунка справа от него. Другие значения атрибута ALIGN, приводящие к выравниванию текста справа от рисунка по его высоте: TOP - по верхнему краю рисунка, CENTER или MIDDLE - по центру рисунка. Чтобы текст обтекал рисунок, используются значения LEFT или RIGHT. Рисунок прижимается соответственно к левому или правому краю экрана, а остальное пространство рядом с ним занимает текст;

ALT="какой-то текст" - альтернативный текст. Текст, который появляется вместо картинки, если по какой-либо причине загрузка изображения не состоялась.

Пример:

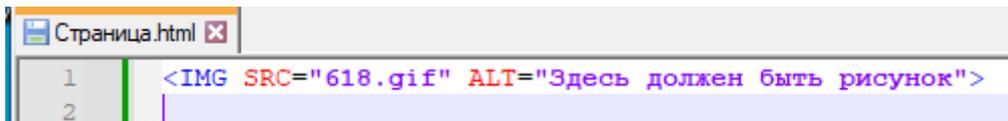


Рисунок picture.gif находится в том же каталоге, что и текущий документ. В том случае, если его не удастся найти или открыть, в рамку, отведенную под рисунок, будет выдан текст: Здесь должен быть рисунок

## 10. Теги гиперссылок

Ссылки - инструмент, позволяющий связывать между собой различные документы или обеспечивать быстрый переход от одной части документа к другой. Браузер обычно выделяет ссылку цветом и подчеркиванием. Курсор при наведении его на ссылку приобретает вид указательного пальца. По двойному щелчку на ссылке браузер включает поисковые средства и раскрывает заказанный документ либо поверх старого, либо в специально оговоренном окне.

Ссылки создаются с помощью контейнера `<A >...</A>` (Add - добавлять). Обязательный атрибут - HREF (Hyper REFerence - гиперссылка). При минимальном наборе атрибутов структура ссылки выглядит так:

```
<A HREF="URL вызываемого файла"> якорь ссылки </A>
```

Примеры.

`<A HREF="filename.html"> нужный документ </A>` - по двойному щелчку на тексте нужный документ браузер раскроет документ filename.html, находящийся в том же каталоге, что и текущий документ;

`<A HREF="folder/filename.html"> нужный документ </A>` - по двойному щелчку на тексте нужный документ браузер раскроет документ filename.html в подкаталоге folder текущего каталога;

`<A HREF="../folder/filename.html"> нужный документ </A>` - по двойному щелчку на тексте нужный документ браузер из каталога, в котором расположен вызываемый документ, поднимется в родительский каталог, затем перейдет в его подкаталог folder и раскроет находящийся там документ filename.html.

`<A HREF="http://www.fortunecity.com/business/fax/339/index.html"> нужный документ </A>` - ссылка с указанием полного адреса файла на другом компьютере. По двойному щелчку на тексте нужный документ установится связь с нужным компьютером и раскроется документ index.html, находящийся там по адресу /business/fax/339/.

Другие атрибуты тега `<A>`:

TITLE = "поясняющий текст" - всплывающая подсказка с пояснением к ссылке;

TARGET="имя окна" - указывает, в каком окне следует раскрывать вызванный документ (подробнее см. п. 11).

Пример (прокомментируйте сами):

```
<A HREF="май/ОтчетПродажиМай.html" TITLE="Майский отчет по продажам" > просмотр отчета </A>
```

Замечание. Якорем ссылки может быть любой объект, видимый в окне браузера, например, рисунок. Для этого разместите `<IMG SRC="...">` между тегами `<A HREF="...">` и `</A>`. Вокруг рисунка появится рамка, которая

показывает, что он является ссылкой. Ширина рамки задается атрибутом BORDER в теге <IMG...>. Если рамка портит внешний вид документа, то укажите в атрибутах рисунка BORDER=0. То, что это гиперссылка, будет видно только по форме курсора, наведенного на рисунок.

Ресурсом, который вызывается по гиперссылке, может быть любой файл, доступный через Интернет.

Примеры:

<A HREF="ftp://servername/directory/file.exe"> нужная программа </A>

Такая ссылка по щелчку на словах нужная программа запустит протокол передачи файлов (ftp) и произведет выгрузку файла file.exe из каталога directory сервера servername на жесткий диск пользователя.

Пишите письма <A HREF="mailto://username@mail.ru" TITLE="Ссылка на почтовый ящик Администратора сайта"> автору </A>

Если навести курсор на слово автору, то появится всплывающая подсказка Ссылка на почтовый ящик Администратора сайта. По двойному щелчку на слове автору будет запущена почтовая программа в режиме создания сообщений, и в поле Кому автоматически будет введен адрес [username@mail.ru](mailto:username@mail.ru)

Ссылки можно делать не только на внешнюю информацию, но и на внутренние места документа. Это облегчает просмотр больших документов. Разберем это на примере. Допустим, электронный вариант этих методических указаний создается по гипертекстовому стандарту. Тогда в начале каждого параграфа необходимо создать метку, на которую будет указывать гиперссылка.

Пример:

<A NAME="п. 10"> </A> или <A NAME="п. 10"> 10. </A>

В первом случае после метки должен идти заголовок параграфа вместе с номером, во втором - просто заголовок, т. е. при организации метки поясняющий текст в контейнере <Л>...</Л> необязателен. Если он вставлен, то он не подчеркивается и выглядит как обычный текст. Теперь Содержание можно оформить как список гиперссылок. Символ # означает, что дальше идет метка внутри документа. Пункт Содержания со ссылкой на п. 10 должен выглядеть так:

<LI><A HREF="#п.10"> 10. Теги гиперссылок </A>

На эту же метку можно будет перейти и из любого другого места методических указаний. Например, ссылку со стр. 1 (п. 1) следует оформить так:

<A HREF="#п.10"> (см. п. 10) </A>

Если гиперссылка указывает на внешний документ, то по умолчанию в окно браузера выдается его начало. Если же нужная в нем информация расположена дальше и обозначена меткой, то можно сразу же открыть документ на этом месте, указав после имени файла метку (так же, как мы открываем книгу с закладкой).

Обратите внимание, что браузеры чувствительны к регистру ссылок. Если Вы вместо "http://webs.web.com" наберете "http://webs.Web.com", то получите нерабочую ссылку. Та же ситуация и с расширениями: если у Вас ссылка на файл "index.htm", а там находится "index.html", то результат тот же - нерабочая ссылка.

С дополнительным материалом по теме можно ознакомиться в литературе [3, 5].

## Задание 1

Составить свое резюме по следующему плану (в общей сложности 4-5 строчек, вместо многоточий и пояснений в скобках вставить свои данные):

Я, ... (ФИО), родился ... (дата и год рождения) в городе ... (место, где родились).

В 20... году окончил ... школу (лицей, гимназию) с углубленным изучением ..., средний балл аттестата ...

В 20. году поступил на специальность .... Средний балл по первой сессии (аттестациям) ...

В свободное от учебы время увлекаюсь ... (перечислить не менее трех увлечений помимо учебы).

Порядок выполнения.

1. Прочтите общие сведения о языке HTML (п.п. 1, 2 теоретических сведений об HTML).

2. Подготовьте бланк для HTML-кода Вашего резюме. Для этого следует:

- выполнить команду Пуск → Программы → Стандартные → Блокнот → Файл → Сохранить как.;

- задать имя файла Резюме\_Фамилия.html (вместо Фамилия должна стоять Ваша фамилия), сохранить файл в своей рабочей папке.

Или с помощью программы NotePad++:

- выполнить команду Пуск → Программы → NotePad++ → Файл → Сохранить как.;

- задать имя файла Резюме\_Фамилия.html (вместо Фамилия должна стоять Ваша фамилия), сохранить файл в своей рабочей папке.

Если все сделано правильно, то в папке Вашей группы появится этот файл со значком обозревателя, в противном случае - со значком блокнота. Тогда следует удалить созданный файл и повторить все снова.

3. Введите в созданный документ стандартные теги:

<HTML>

<HEAD>

</HEAD>

<BODY>

</BODY>

</HTML>

Теги набираются с помощью латинских букв, регистр (прописные или строчные) значения не имеет. Подробнее об этих тегах см. п. 4.

4. Внутри контейнера <BODY> напечатайте свое резюме. Сохраните файл.

5. Посмотрите, как выглядит набранный текст без стилей текста. Для этого следует открыть созданный файл из папки своей группы.

6. Проведите первую коррекцию кода. Для этого следует:

- не закрывая браузер, перейти в окно Блокнота или NotePad++;

- вставить в контейнер <HEAD> ... </HEAD> контейнер <TITLE> Резюме ФИО </TITLE> (вместо ФИО должна стоять Ваша фамилия);

- сохранить файл, перейти в окно браузера, нажать кнопку Обновить (или F5).

Посмотрите, как повлиял тег <TITLE> на информацию в заголовке браузера.

7. Проведите вторую коррекцию кода. Для этого следует:

- в открывающий тег <BODY> вставить атрибуты TEXT и BGCOLOR, значения цветов выбрать самим. Подробнее об этих атрибутах в п. 4;
- сохранить файл, перейти в окно браузера, нажать кнопку Обновить (или F5).

Посмотрите, как изменился вид информации в окне браузера.

Если какой-то атрибут «не сработал», значит, допущены ошибки в написании английских слов или пропущены пробелы между атрибутами. Следует найти ошибки и повторить цикл коррекции кода.

Итак, один цикл коррекции кода включает следующие действия: внесение и сохранение исправлений в окне блокнота, переход в окно браузера и обновление его информации кнопкой Обновить (F5). В дальнейших пунктах инструкции будут просто указываться изменения, которые следует внести в код, а все остальные действия перечисляться не будут.

8. Проведите третью коррекцию кода. Для этого следует:

- после раздела «ФИО, где и когда родился», поставить тег абзаца <P>;
- после раздела «какую школу кончил, средний балл аттестата» - тег разрыва строки <BR>;
- перед разделом «увлечения помимо учебы» вставить в код тег горизонтальной линии <HR>.

Подробнее об этих тегах см. п. 6.

Посмотрите через браузер, в чем разница действий тегов <P> и <BR>, как действует тег <HR> со значениями атрибутов по умолчанию.

9. Оформите линию нестандартным способом. Для этого следует внести в тег <HR> атрибуты длины, толщины, цвета и выравнивания линии (см. п. 6). Задайте длину 20 % - 50 % от ширины окна, выравнивание по центру, толщину 4 - 10 пикселей, цвет - любой.

Посмотрите через браузер, как действуют атрибуты на линию, созданную тегом <HR>.

10. Измените вид шрифтов в разных частях резюме. Для этого следует контейнером <H1> выделить свою фамилию, контейнером <I> - город, где Вы родились, контейнером <H6> - информацию о школе, контейнером <FONT> с атрибутами SIZE=7 COLOR=RED - информацию о факультете, контейнером <FONT> с атрибутами SIZE=1 COLOR=GREEN - информацию о друзьях, контейнером <B> выделить какое-нибудь увлечение. Подробнее об этих тегах см. п. 5.

Посмотрите через браузер, как действуют эти теги.

11. Оформите часть текста в виде списка. Для этого следует контейнером <OL> выделить блок об увлечениях, разбить его на пункты тегами <LI> (подробнее см. п. 7). Аналогично контейнером <UL> и тегами <LI> оформить в виде маркированного списка перечень Ваших друзей.

Посмотрите, как выглядят стандартные списки.

12. Измените нумерацию в списке. Для этого с помощью атрибутов TYPE и START поменяйте нумерацию в списке увлечений (подробнее см. п. 7).

Посмотрите, как изменится стандартный нумерованный список при новых атрибутах.

## Задание 2

Создать HTML-документ, в котором представлены сведения о гостиницах для размещения участников некоторой конференции. Эти сведения оформить в виде таблицы:

Таблица. Характеристики гостиниц

Название	Адрес	Тип номера	Цена за день
Центральная	Пр. Мира, д. 4	Люкс одноместный	\$150
		Люкс двухместный	\$250
		Одноместный	\$80
Спортивная	Бульвар Профсоюзов, д. 15	Одноместный	\$50
		Двухместный	\$80
Вокзальная	Привокзальная пл., д. 3	Четырехместный	\$50

Порядок выполнения.

1. Подготовьте бланк для HTML-документ и сохраните файл в своей рабочей папке под именем Таблица\_Фамилия.html (вместо Фамилия должна стоять Ваша фамилия).

2. Введите в созданный документ стандартные теги:

```

1 <HTML>
2 <HEAD>
3 <TITLE> Таблица ФИО </TITLE>
4 </HEAD>
5 <BODY>
6 </BODY>
7 </HTML>

```

Вместо ФИО в контейнере <TITLE> должна стоять Ваша фамилия.

3. Внутри контейнера <BODY> вставьте скелетные теги таблицы (см. п. 8):

```

1 <HTML>
2 <HEAD>
3 <TITLE> Таблица ФИО </TITLE>
4 </HEAD>
5 <BODY>
6 <TABLE>
7 <CAPTION>
8   Характеристики гостиниц
9 </CAPTION>
10 <TR>
11 </TR>
12 <TR>
13 </TR>
14 <TR>
15 </TR>
16 </TABLE>
17 </BODY>
18 </HTML>

```

Количество контейнеров для строк рассчитывается так же, как в редакторе Word (по самому мелкому дроблению). В нашем примере - 7.

Отцентрируйте заголовок таблицы. Расположите ее на цветном фоне. Выберите цвет для шрифта.

4. Заполните первую строку таблицы. Так как это «шапка» с названиями столбцов, для ячеек используйте контейнеры `<TH> ... </TH>`. Окончательно контейнер первой строки выглядит так:

```

1 <HTML>
2 <HEAD>
3 <TITLE> Таблица ФИО </TITLE>
4 </HEAD>
5 <BODY>
6 <TABLE>
7 <CAPTION>
8   Характеристики гостиниц
9 </CAPTION>
10 <TR>
11 <TH> Название </TH>
12 <TH> Адрес </TH>
13 <TH> Тип номера </TH>
14 <TH> Цена за день </TH>
15 </TR>
16 <TR>
17 </TR>
18 <TR>
19 </TR>
20 </TABLE>
21 </BODY>
22 </HTML>
23

```

5. Заполните вторую строку таблицы. Так как это уже содержание таблицы, для ячеек используйте контейнеры `<TD> ... </TD>`. Первая и вторая ячейки по высоте занимают 3 строки. Чтобы объединить их, добавьте атрибуты

ROWSPAN=3. Окончательно контейнер второй строки выглядит так:

```

1 <HTML>
2 <HEAD>
3 <TITLE> Таблица ФИО </TITLE>
4 </HEAD>
5 <BODY>
6 <TABLE>
7 <CAPTION>
8   Характеристики гостиниц
9 </CAPTION>
10 <TR>
11 <TH> Название </TH>
12 <TH> Адрес </TH>
13 <TH> Тип номера </TH>
14 <TH> Цена за день </TH>
15 </TR>
16 <TR>
17 <TD ROWSPAN=3> Центральная </TD>
18 <TD ROWSPAN=3> Пр. Мира, д. 4</TD>
19 <TD> Люкс одноместный </TD>
20 <TD> $150 </TD>
21 </TR>
22 <TR>
23 </TR>
24 </TABLE>
25 </BODY>
26 </HTML>
  
```

6. Заполните контейнер третьей строки таблицы. Для ячеек по-прежнему используйте контейнеры `<TD> ... </TD>`. Первая и вторая ячейки входят в объединенные раньше блоки, их объявлять не надо. А третья и четвертая ячейки содержат новую информацию. Скопируйте контейнеры соответствующих ячеек из второй строки и поменяйте в них содержимое. Окончательно контейнер этой строки выглядит так:

```

1 <HTML>
2 <HEAD>
3 <TITLE> Таблица ФИО </TITLE>
4 </HEAD>
5 <BODY>
6 <TABLE>
7 <CAPTION>
8   Характеристики гостиниц
9 </CAPTION>
10 <TR>
11 <TH> Название </TH>
12 <TH> Адрес </TH>
13 <TH> Тип номера </TH>
14 <TH> Цена за день </TH>
15 </TR>
16 <TR>
17 <TD ROWSPAN=3> Центральная </TD>
18 <TD ROWSPAN=3> Пр. Мира, д. 4 </TD>
19 <TD> Люкс одноместный </TD>
20 <TD> $150 </TD>
21 </TR>
22 <TR>
23 <TD> Люкс двухместный </TD>
24 <TD> $250 </TD>
25 </TR>
26 </TABLE>
27 </BODY>
28 </HTML>

```

7. Аналогично п. 6 заполните контейнер четвертой строки. Сохраните документ и просмотрите его через браузер. В таблице пока нет разлиновки, но, если все сделано правильно, структура расположения информации должна просматриваться.

8. Добавьте в таблицу сведения о гостинице Спортивная. Они располагаются в пятой и шестой строках. Похожие строки находятся во второй и третьей строках таблицы, и теги, создающие их, уже сформированы. Скопируйте теги из второй и третьей строк соответственно в контейнеры пятой и шестой строк, замените содержание ячеек. Так как здесь первая и вторая ячейки по высоте занимают только 2 строки, поменяйте значение атрибута ROWSPAN. Получаем для новых строк:

```

<TR>
<TD ROWSPAN=2> Спортивная </TD>
<TD ROWSPAN=2> Бульвар Профсоюзов, д. 15 </TD>
<TD> одноместный </TD>
<TD> $50 </TD>
</TR>
<TR>
<TD> двухместный </TD>
<TD> $80 </TD>
</TR>

```

9. Добавьте в таблицу сведения о гостинице Вокзальная. Они располагаются в седьмой строке. В ней нет объединения ячеек, все четыре ячейки содержат новую информацию. Можно скопировать теги второй строки, убрать в

них атрибуты ROWSPAN и просто заменить информацию в контейнерах ячеек:

```
<TR>
<TD> Вокзальная </TD>
<TD> Привокзальная пл., д. 3 </TD>
<TD> Четырехместный </TD>
<TD> $50 </TD>
</TR>
```

Сохраните документ и просмотрите его через браузер.

10. Форматирование таблицы. Добавьте в тег <TABLE> атрибуты BORDER=3 WIDTH=70% ALIGN-CENTER BGCOLOR=YELLOW BORDERCOLOR=MAROON (можно взять свои значения атрибутов).

Посмотрите через браузер, как действуют эти атрибуты на вид таблицы. Добавьте атрибут CELLSPACING=0, потом измените его на CELLSPACING=10. Посмотрите через браузер, за что отвечает этот атрибут. Подробнее см. п. 8.

11. Форматирование ячеек. Выделите сведения, относящиеся к разным гостиницам, разной заливкой. Выровняйте по центру ячеек цены номеров. Необходимые для этого сведения - в п. 8.

### Задание 3

В документ Таблица\_Фамилия.html вставить файл-картинку в виде фона для одной ячейки таблицы, затем для таблицы в целом, затем для документа в целом. После таблицы вставить гиперссылку на администратора конференции.

В документ Резюме\_Фамилия.html вставить файл-фотографию, проверить разные варианты обтекания изображения текстом.

Порядок выполнения.

1. Подготовьте необходимые для работы файлы. Для этого нужно:

- подготовьте три файла с изображениями пейзажей и один с изображением человека, скопировать их в свою рабочую папку;
- в своей папке создать новую папку с именем Сайт\_Фамилия;
- открыть в папке своей группы файл Таблица\_Фамилия.html;
- в окне браузера выполнить команду Вид → Показать код HTML.

2. Создайте фон в разных частях документа. Для этого нужно:

- в открывающий тег какой-нибудь ячейки таблицы вставить атрибут BACKGROUND-'имя файла-пейзажа с расширением'. В результате тег должен выглядеть примерно так:

```
<TD BACKGROUND="море.jpg">
```

- сохранить код с исправлением и посмотреть его в окне браузера. В ячейке в виде фона отображен левый верхний угол картинки;

- скопировать атрибут BACKGROUND-"..." в открывающий тег <TABLE>, заменить название файла на файл другого пейзажа. Сохранить код с исправлением и посмотреть его в окне браузера. Теперь вся таблица изображается на фоне второй картинки, а в ячейке по-прежнему сохраняется предыдущий фон;

- скопировать атрибут BACKGROUND-'...' в открывающий тег <BODY>, заменить название файла на файл третьего пейзажа. Сохранить код с исправлением и посмотреть его в окне браузера. Фон, указанный в <BODY>, заполняет все окно браузера, а фон, заказанный во внутренних тегах,

распространяется только на область их действия.

Вывод: внутренние назначения заменяют внешние.

3. Проанализируйте способы указания адресов нужных файлов (URL). Для этого сделайте следующие манипуляции:

- перенесите в папку Сайт\_Фамилия все файлы пейзажей, нажмите кнопку Обновить в окне браузера. Фон пропал;

- в атрибутах BACKGROUND-"..." добавьте перед именами файлов название папки, в которой они находятся. В результате получим примерно это:

BACKGROUND-" Сайт\_Иванов\море .jpg"

Сохраните изменения и просмотрите их через браузер. Фон появился. Прочтите п. 6 (фрагмент про относительный URL) и просмотрите примеры п. 9 при разных ссылках на файлы. В нашем примере мы сделали ссылку на файл в дочерней папке;

- поменяйте местами файл с таблицей и файлы пейзажей (файл с таблицей - в папку Сайт., а файлы пейзажей - в папку своей группы), закройте окно браузера и откройте таблицу снова из папки Сайт. Фон опять пропал;

- в атрибутах BACKGROUND-". " замените название дочерней папки на значок родительской. В результате получим примерно это:

BACKGROUND-"..\море.jpg"

Сохраните изменения и просмотрите их через браузер. Фон появился. В этом примере мы сделали ссылку на файл в родительской папке.

Вывод: самый простой URL получается, если все файлы, необходимые для просмотра документа, находятся в одной папке. В противном случае надо указывать пути к ним.

4. Добавьте первую гиперссылку. Для этого нужно:

- в код документа с таблицей после закрывающего тега </TABLE> добавить текст.

Ответственный за размещение гостей <A HREF="..\Резюме ...html"> ... (Ваша фамилия) </A>

Внимание! В атрибуте HREF при указании пути и названия файла Вашего резюме следует соблюдать все регистры букв и пробелы. Сохраните изменения и просмотрите их через браузер;

- сделайте двойной щелчок на появившейся гиперссылке. Если гиперссылка не раскрыла Ваше резюме, проверьте правильность составления пути и названия файла в атрибуте HREF контейнера <A> ...</A>. Чтобы вернуться к документу с таблицей, следует нажать кнопку Назад в окне браузера или вставить в код резюме другую гиперссылку примерно такого вида:

<A HREF="Сайт...(Ваша фамилия) \Таблица ...(Ваша фамилия).html"> Вернуться к таблице? </A>

Подробнее о гиперссылках и их атрибутах см. п. 10.

5. Добавьте гиперссылку на почтовый ящик e-mail. Для этого нужно в код документа с таблицей после контейнера с первой гиперссылкой добавить строку:

Сообщите нам о нужном Вам номере по <A HREF="mailto://adm@mail.ru"> e-mail </A>

Сохраните изменение и просмотрите результат. Сделайте двойной щелчок

на этой гиперссылке. Просмотрите окно создания сообщения. Прочтите в п. 6 о структуре URL и разберите в п. 10 примеры различных гиперссылок. Добавьте во все гиперссылки атрибут TITLE с какими-нибудь поясняющими надписями.

б. Вставьте изображение как самостоятельный объект. Для этого откройте файл с Вашим резюме. После сведений о том, где Вы родились, вставьте тег <IMG SRC="..."> (вместо многоточия укажите URL своего файла с изображением человека). Сохраните и просмотрите изменение. Добавьте в тег <IMG> атрибуты размеров картинки, сделайте ей окантовку в виде толстой рамки, поэкспериментируйте с вариантами разного расположения текста около нее. Необходимые для этого сведения см. в п. 9.

### **Варианты задания**

Создать сайт из 5 документов на выбранную тему. Минимальный набор средств языка HTML, который должен быть использован в документах:

- разные приемы форматирования текста;
- оформление списков;
- оформление таблиц;
- гиперссылки;
- использование изображений.

Возможные темы сайтов: любимая вокальная группа, театральный коллектив, спортивная команда, автор любимых книг, Ваш город, какой-нибудь курорт, вид спорта, отдыха, реклама какой-нибудь фирмы, себя любимого и т. п.

В документе Приветствие.html можно использовать в качестве самостоятельного объекта изображение логотипа фирмы, фотографию на тему сайта, а также разместить краткое пояснение того, чему посвящен Ваш сайт в целом, как двигаться по нему.

В документах Пункт1.html, Пункт2.html, Пункт3.html, Пункт4.html можно разместить разные группы информации по выбранной теме. Например, если выбрана тема «спортивная команда», Пункт1.html может содержать текстовые сведения об истории ее создания, Пункт2.html - список игроков, Пункт3.html - таблицу игр.

Документ Приветствие.html должен содержать гиперссылки на Пункт1.html, Пункт2.html, Пункт3.html, Пункт4.html.

### **Содержание отчета**

После выполнения работы составить отчет, включающий исходный код html-страниц, описание основных использованных тегов и снимки экрана (скриншоты).

### **Технология выполнения работы**

В данной работе с помощью языка разметки HTML должны быть созданы 5 web-страниц, оформленных с помощью основных html-тегов.

### **Вопросы для защиты работы**

1. Дайте определение понятиям "HTML", "гиперссылка", "браузер", "тег".
2. Общая структура тегов.

3. В каких случаях используются &-последовательности и комментарии?
4. Расскажите об общей структуре html-документа.
5. Перечислите теги форматирования шрифта.
6. Как оформляются смысловые части текста и списки?
7. Создание таблиц с помощью языка HTML.
8. Приведите примеры использования графических изображений в html-документе.
9. Как создать гиперссылку?

### **Список использованных источников**

1. Беликова, С. А. Основы HTML и CSS: проектирование и дизайн веб-сайтов: учебное пособие по курсу «Web-разработка» : [16+] / С. А. Беликова, А. Н. Беликов ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2020. – 176 с. Режим доступа: – URL: <https://biblioclub.ru/index.php?page=book&id=598663>
2. Титов, В. А. Разработка WEB-сайта средствами языка HTML : учебное пособие / В. А. Титов, Г. И. Пещеров. – Москва : Институт мировых цивилизаций, 2018. – 184 с.: ил., табл. – Режим доступа: – URL: <https://biblioclub.ru/index.php?page=book&id=598475>