Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Ремизов Петр Алексеевич
Должность: декан ЕНФ
Дата подписания: 01.09.2020 13:37:34
Уникальный программный ключ:

# Аннотация к рабочей программе

дисциплины «**Программирование на языках высокого уровня**»

## Цель дисциплины

Формирование у студентов теоретико-прикладных представлений об основах алгоритмизации и проектирования программ, приемов программирования на конкретных алгоритмических языках, основ организации вычислительного процесса на ЭВМ; привитие навыков анализа корректности программ, их отладки и тестирования с использованием современных сред программирования.

**Задачи дисциплины:**

– изучение технологий разработки программ на языках высокого уровня;

– приобретение навыков практического использования языков программирования высокого уровня;

– приобретение навыков хранения и обработки текстовой и числовой информации.

**Компетенции, формируемые в результате освоения дисциплины:**

– способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности (ПК-1);

– способен применять различные технологии разработки программного обеспечения (ПК-4);

– способен создавать программные интерфейсы (ПК-11).

**Разделы дисциплины:**

1. Основные понятия и определения.
2. Программирование линейных алгоритмов на языке C#.
3. Программирование разветвленных алгоритмов на языке C#.
4. Программирование циклических алгоритмов на языке C#.
5. Работа с одномерными массивами на языке C#.
6. Работа с двумерными массивами на языке C#.
7. Функции.
8. Виды строк в C#. Использование библиотеки функций работы со строками в C#.
9. Графический режим работы.
10. Структуры и объединения.
11. Файлы в C#. Использование библиотеки функций работы с файлами в C#.
12. Обработка исключительных ситуаций. Примеры.
13. Объектно-ориентированное программирование, инкапсуляция.
14. Объектно-ориентированное программирование, полиморфизм, наследование.
15. Заключение. Перспективы развития и использования языков программирования

МИНОБРНАУКИ РОССИИ

Юго-Западный государственный университет

УТВЕРЖДАЮ:
И. о. декана факультета
*(наименование ф-та, полностью)*
фундаментальной и прикладной информатики
_____Т.А. Ширабакина
*(подпись, инициалы, фамилия)*
«20» 06 2019 г.

# РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование на языках высокого уровня
*(наименование дисциплины)*

ОПОП ВО 09.03.04 Программная инженерия,
*шифр и наименование направления подготовки (специальности)*

направленность (профиль) «Разработка программно-информационных систем
*наименование направленности (профиля, специализации)*

форма обучения очная
*(очная, очно-заочная, заочная)*

Курск – 2019

Рабочая программа дисциплины составлена в соответствии с ФГОС-3++ – бакалавриат по направлению подготовки (специальности) 09.03.04 «Программная инженерия» на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета (протокол №7 «29» марта 2019г.).

Рабочая программа дисциплины обсуждена и рекомендована к реализации в образовательном процессе для обучения студентов по ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем» на заседании кафедры программной инженерии №13 «20» 06 2019 г.

Зав. кафедрой _____ к.т.н., доцент Малышев А. В.
Разработчик программы _____ к.т.н., доцент Белова Т. М.
Директор научной библиотеки _____ Макаровская В. Г.

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол №7«25»02 2020г., на заседании кафедры программной инженерии «10» 06 2020г., протокол №11.

Зав. кафедрой _____ Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол №6«26»02 2021г., на заседании кафедры программной инженерии «02» 07 2021г., протокол №12.

Зав. кафедрой _____ Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол №7«28»02 2022г., на заседании кафедры программной инженерии «17» 06 2022г., протокол №11.

Зав. кафедрой _____ Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 6 «26» 02 2021 г. на заседании кафедры

ПИ, №11 от 13.06.2023

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 7 «28» 02 2022 г. на заседании кафедры

ПИ, №11 от 10.06.2024

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 9 «31» 03 2025 г. на заседании кафедры

ПИ №12 от 30.06.2025

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 8 «24» 02 2026 г. на заседании кафедры

ПИ №12 от 30.06.2026

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № «» 20 г. на заседании кафедры

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой

# 1 Цель и задачи дисциплины. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы

## 1.1 Цель дисциплины

Формирование у студентов теоретико-прикладных представлений об основах алгоритмизации и проектирования программ, приемов программирования на конкретных алгоритмических языках, основ организации вычислительного процесса на ЭВМ; привитие навыков анализа корректности программ, их отладки и тестирования с использованием современных сред программирования.

## 1.2 Задачи дисциплины

- изучение технологий разработки программ на языках высокого уровня;
- приобретение навыков практического использования языков программирования высокого уровня;
- приобретение навыков хранения и обработки текстовой и числовой информации.

## 1.3 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы

Таблица 1.3 – Результаты обучения по дисциплине

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| ПК-1 | Способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности | ПК-1.3 Осуществляет сбор, анализ, формулировку и документирование требований заинтересованных лиц | ***Знать:*** методы и приемы формализации задач, языки формализации функциональных спецификаций ***Уметь:*** использовать методы и приемы формализации задач, использовать языки формализации функциональных спецификаций ***Владеть:*** составлением формализованных описаний решений поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов |

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| | | ПК-1.4<br>Осуществляет разработку и документирование технического задания на программную систему | ***Знать:***<br>возможности существующей программно-технической архитектуры, возможности современных и перспективных средств разработки программных продуктов, технических средств<br>***Уметь:***<br>использовать возможности существующей программно-технической архитектуры, использовать возможности современных и перспективных средств разработки программных продуктов, технических средств<br>***Владеть:***<br>анализом возможностей реализации требований к программному обеспечению, оценка времени и трудоемкости реализации требований к программному обеспечению, оценкой и согласованием сроков выполнения поставленных задач |
| ПК-4 | Способен применять различные технологии разработки программного обеспечения | ПК-4.2<br>Разрабатывает интерфейсы взаимодействия модулей системы | ***Знать:***<br>синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования<br>***Уметь:***<br>использовать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования<br>***Владеть:***<br>созданием программного кода в соответствии с техническим заданием (готовыми спецификациями) |
| | | ПК-4.3 | ***Знать:*** |

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| | | Разрабатывает интерфейсы взаимодействия с внешней средой | методы и приемы алгоритмизации поставленных задач<br>***Уметь:***<br>использовать методы и приемы алгоритмизации поставленных задач<br>***Владеть:***<br>разработкой алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов |
| | | ПК-4.4<br>Выполняет тестирование программного обеспечения | ***Знать:***<br>методы и приемы отладки программного кода<br>***Уметь:***<br>использовать методы и приемы отладки программного кода, выявлять ошибки в программном коде<br>***Владеть:***<br>проверкой и отладкой программного кода |
| | | ПК-4.5<br>Разрабатывает структуры данных | ***Знать:***<br>типы данных, определяемые пользователем: массивы, строки, структуры; файлы, классы<br>***Уметь:***<br>использовать типы данных, определяемые пользователем: массивы, строки, структуры; файлы, классы<br>***Владеть:***<br>проектированием структур данных |
| ПК-11 | Способен создавать программные интерфейсы | ПК-11.2<br>Разрабатывает алгоритм работы интерфейса | ***Знать:***<br>методы и средства проектирования программных интерфейсов<br>***Уметь:***<br>использовать методы и средства проектирования программных интерфейсов<br>***Владеть:*** |

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| | | | проектированием программных интерфейсов |
| | | ПК-11.3 Разрабатывает и отлаживает программные модули для реализации функций интерфейса | ***Знать:*** правила, алгоритмы и технологии создания тестовых наборов данных ***Уметь:*** использовать правила, алгоритмы и технологии создания тестовых наборов данных ***Владеть:*** практическими навыками разработки программ для решения задач на языках высокого уровня |

## 2 Указание места дисциплины в структуре основной профессиональной образовательной программы

Дисциплина «Программирование на языках высокого уровня» входит в часть, формируемую участниками образовательных отношений блока 1 «Дисциплины (модули)» основной профессиональной образовательной программы – программы бакалавриата 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем». Дисциплина изучается на 1 курсе в 1 и 2 семестрах.

## 3 Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 11 зачетных единиц (ЗЕТ), 396 часов.

Таблица 3 – Объём дисциплины

| Виды учебной работы | Всего, часов |
|---|---|
| Общая трудоемкость дисциплины | 396 |
| Контактная работа обучающихся с преподавателем по видам учебных занятий (всего) | 129,3 |
| в том числе: | |
| лекции | 54 |
| лабораторные занятия | 54, из них практическая |

| Виды учебной работы | Всего, часов |
|---|---|
| | подготовка –12 |
| практические занятия | 18 |
| Самостоятельная работа обучающихся (всего) | 212,7 |
| Контроль (подготовка к экзамену) | 54 |
| Контактная работа по промежуточной аттестации (всего АттКР) | 3,3 |
| в том числе: | |
| зачет | не предусмотрен |
| зачет с оценкой | не предусмотрен |
| курсовая работа (проект) | 1 |
| экзамен (включая консультацию перед экзаменом) | 2,3 |

## 4 Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

### 4.1 Содержание дисциплины

Таблица 4.1.1 – Содержание дисциплины, структурированное по темам (разделам)

| № п/п | Раздел (тема) дисциплины | Содержание |
|---|---|---|
| 1 | 2 | 3 |
| 1 семестр | | |
| 1 | Основные понятия и определения. | Понятие алгоритма. Формы представления алгоритмов. ГОСТы ЕСПД. Интегрированная среда разработки программ на языке C#. Приемы отладки программ. |
| 2 | Программирование линейных алгоритмов на языке C#. | Типы данных. Структура программы. Оператор присваивания. Префиксные и постфиксные формы. |
| 3 | Программирование разветвленных алгоритмов на языке C#. | Условные операторы. Приоритеты выполнения арифметических и логических операций. |
| 4 | Программирование циклических алгоритмов на языке C#. | Операторы цикла. Цикл с предусловием, цикл с постусловием, цикл со счетчиком. |
| 5 | Работа с одномерными массивами на языке C#. | Массивы, ввод и вывод одномерных массивов. Статические и динамические массивы. Поиск минимального и максимального элементов. |
| 6 | Работа с двумерными массивами на языке C#. | Массивы, ввод и вывод двумерных массивов. Статические и динамические массивы. Поиск минимального и максимального элементов. |
| 7 | Функции. | Структура программы с подпрограммами на языке C#. Структура функций. Параметры. Передача массивов в функцию. Параметры по умолчанию. Рекурсия. |
| 2 семестр | | |
| 8 | Строки. | Виды строк в C#. Использование библиотеки функций |

| № п/п | Раздел (тема) дисциплины | Содержание |
|---|---|---|
| 1 | 2 | 3 |
| | | работы со строками в C#. |
| 9 | Графический режим работы. | Понятие канвы. Графические примитивы. |
| 10 | Структуры и объединения. | Описания структур и объединений. Отличия. Примеры использования. |
| 11 | Файлы в C#. | Использование библиотеки функций работы с файлами в C#. |
| 12 | Обработка исключительных ситуаций. | Назначение, варианты использования. Примеры. |
| 13 | Объектно-ориентированное программирование. | Инкапсуляция. Правила построения классов. Примеры. |
| 14 | Объектно-ориентированное программирование. | Полиморфизм, наследование. Раннее связывание и позднее связывание. Простое и множественное наследование. Абстрактные классы. |
| 15 | Заключение. | Перспективы развития и использования языков программирования |

Таблица 4.1.2 - Содержание дисциплины и ее методическое обеспечение

| № п/п | Раздел (тема) дисциплины | Виды деятельности | | | Учебно-методические материалы | Формы текущего контроля успеваемости (*по неделям семестра*). | Компетенции |
|---|---|---|---|---|---|---|---|
| | | лек., час | № лаб. | № пр. | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 семестр | | | | | | | |
| 1. | Основные понятия и определения. | 4 | 1 | – | У1, У2, МУ1 | КО, С | ПК-1, ПК-4, ПК-11 |
| 2. | Программирование линейных алгоритмов на языке C#. | 2 | 2 | – | У1, У2, МУ1 | КО, С | ПК-1, ПК-4, ПК-11 |
| 3. | Программирование разветвленных алгоритмов на языке C#. | 2 | 3 | – | У1, У2, МУ2 | КО, С | ПК-1, ПК-4, ПК-11 |
| 4. | Программирование циклических алгоритмов на языке C#. | 2 | 4 | – | У1, У2, МУ3 | КО, С | ПК-1, ПК-4, ПК-11 |
| 5. | Работа с одномерными массивами на языке C#. | 2 | 5 | – | У1, У2, МУ4 | КО, С | ПК-1, ПК-4, ПК-11 |

| № п/п | Раздел (тема) дисциплины | Виды деятельности | | | Учебно-методические материалы | Формы текущего контроля успеваемости (*по неделям семестра*). | Компетенции |
|---|---|---|---|---|---|---|---|
| | | лек., час | № лаб. | № пр. | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 6. | Работа с двумерными массивами на языке C#. | 2 | 6 | – | У1, У2, МУ5 | КО, С | ПК-1, ПК-4, ПК-11 |
| 7. | Функции. | 4 | 7 | – | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 2 семестр | | | | | | | |
| 8. | Виды строк в C#. Использование библиотеки функций работы со строками в C#. | 2 | 1 | 1 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 9. | Графический режим работы. | 2 | 2 | 2 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 10. | Структуры и объединения. | 2 | 3 | 4 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 11. | Файлы в C#. Использование библиотеки функций работы с файлами в C#. | 2 | 4 | 5 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 12. | Обработка исключительных ситуаций. Примеры. | 2 | 5 | 6 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 13. | Объектно-ориентированное программирование, инкапсуляция. | 2 | 6 | 7 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 14. | Объектно-ориентированное программирование, полиморфизм, наследование. | 2 | 7 | 8 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 15. | Заключение. Перспективы развития и использования языков программирования. | 2 | 7 | 9 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |

КО –контрольный опрос, С – собеседование.

## 4.2 Лабораторные работы и (или) практические занятия

### 4.2.1 Лабораторные работы

Таблица 4.2.1 — Лабораторные работы

| № | Наименование лабораторной работы | Объём, час. |
|---|---|---|
| 1. | Изучение основных режимов работы в интегрированной среде C#. Программирование простого примера. | 4 |

| № | Наименование лабораторной работы | Объём, час. |
|---|---|---|
| 2. | Программирование линейных алгоритмов на языке C#. | 4, из них практическая подготовка – 2 |
| 3. | Программирование разветвленных алгоритмов на языке C#. | 4, из них практическая подготовка – 2 |
| 4. | Программирование циклических алгоритмов на языке C#. | 4 |
| 5. | Программирование с использованием одномерных массивов на C#. | 4 |
| 6. | Программирование с использованием двумерных массивов на языке C#. | 4, из них практическая подготовка – 2 |
| 7. | Программирование с использованием функций на языке C#. | 6, из них практическая подготовка – 2 |
| 8. | Программирование с использованием строк на языке C#. | 4, из них практическая подготовка – 2 |
| 9. | Программирование с использование структур данных на языке C#. | 4 |
| 10. | Программирование в графическом режиме на языке C#. | 4 |
| 11. | Программирование с использованием файлов на языке C#. | 6, из них практическая подготовка – 2 |
| 12. | Программирование с использованием объектно-ориентрованной технологии на языке C# | 6 |
| Итого | | 54 |

### 4.2.2 Практические занятия

Таблица 4.2.2 — Практические занятия

| № | Наименование практического занятия | Объём, час. |
|---|---|---|
| 1. | Программирование линейных и разветвленных алгоритмов на языке C#. | 4 |
| 2. | Программирование циклических алгоритмов на языке C#. | 2 |
| 3. | Программирование с использованием одномерных массивов на C#. | 4 |
| 4. | Программирование с использованием двумерных массивов на языке C#. | 4 |
| 5. | Программирование с использованием функций на языке C#. | 4 |
| Итого | | 18 |

## 4.3 Самостоятельная работа студентов (СРС)

Таблица 4.3 - Самостоятельная работа студентов

| № п.п | Наименование раздела дисциплины | Срок выполнения | Время, затрачиваемое на выполнение СРС, час. |
|---|---|---|---|
| 1 семестр | | | |
| 1 | Основные понятия и определения. | 1,2 неделя | 12 |
| 2 | Программирование линейных алгоритмов на языке C#. | 3,4 неделя | 12 |
| 3 | Программирование разветвленных алгоритмов на языке C#.. | 5,6 неделя | 12 |
| 4 | Программирование циклических алгоритмов на языке C#. | 7,8 недели | 12 |
| 5 | Работа с одномерными массивами на языке C#. | 9,10 неделя | 15 |
| 6 | Работа с двумерными массивами на языке C#. | 11,12 недели | 16 |
| 7 | Функции. | 13–18 недели | 18,85 |
| 2 семестр | | | |
| 8 | Строки. | 1–3недели | 10 |
| 9 | Графический режим работы. | 4–6 недели | 10 |
| 10 | Структуры и объединения. | 10–12 недели | 10 |
| 11 | Файлы в C#. | 12–14 недели | 10 |
| 12 | Обработка исключительных ситуаций. | 14–6 недели | 10 |
| 13 | Объектно-ориентированное программирование. | 16–18 недели | 14,85 |
| 14 | Курсовая работа | 1–17 недели | 50 |
| Итого | | | 212,7 |

## Курсовая работа

Курсовая работа выполняется на первом курсе во 2 семестре.

Отчетность по курсовому проектированию состоит из программной документации и созданного программного продукта в виде исполняемого файла.

Программная документация состоит из 3 разделов: технического задания, технического проекта, рабочего проекта, которые оформляются согласно ЕСПД, ГОСТ 19.701-90 (ISO 5807-85), ГОСТ 2.304-88.

Техническое задание содержит обозначение и наименование программного изделия, краткую характеристику области применения, назначение разработки, назначение разработки, требования заказчика (пользователя). Программное изделие рассматривается как единый функциональный модуль. Примерное количество страниц 10.

На стадии технического проекта производится дальнейшая функциональная декомпозиция программного изделия. При этом функции, определенные на стадии технического задания, разбиваются на модули. Такие модули проектируются на уровне внешнего описания, т.е. для каждого из них определяется функциональность с точки зрения пользователя. Примерное количество страниц 10.

Стадия рабочего проекта является заключительной как в конструировании программного изделия, так и в его изготовлении. В процессе конструирования окончательно определяется структура всего программного изделия. Определяется структура каждого фактического программного модуля в виде схемы его алгоритма. Для всего изделия в целом и для каждого модуля рассчитываются и конструируются тесты. Описание программы представляет собой внутреннее описание программного изделия. Примерное количество страниц 20.

Примерный график выполнения по неделям (с указанием контрольных мероприятий) – 50 часов.

| № недели | Наименование и содержание этапа | Примечание |
|---|---|---|
| 1 | Изучение задания на курсовую работу и его согласование. | |
| 2 | Изучение метода, подлежащего реализации. | |
| 3,4 | Составления технического задания. | |
| 5 | Утверждение технического задания. | Контрольная точка |
| 6 | Разработка технического проекта. Функциональное проектирование. | |
| 7 | Разработка технического проекта. Состав и взаимосвязь функциональных модулей. | |
| 8 | Утверждение и оформление технического проекта. | Контрольная точка. |
| 9 | Разработка рабочего проекта. Определение состава программных модулей и их взаимосвязей. | |
| 10 | Разработка рабочего проекта. Составление схем алгоритмов верхнего уровня. | |
| 11 | Разработка тестов для алгоритмов верхнего уровня. | Контрольная точка |
| 12 | Программирование и отладка программ верхнего уровня. | |
| 13 | Составление схем алгоритмов нижнего уровня. | |
| 14 | Разработка тестов для алгоритмов нижнего уровня. | |
| 15 | Программирование и отладка алгоритмов нижнего уровня. | Контрольная точка |
| 16 | Тестирование и отладка всех модулей. | |
| 17 | Оформление пояснительной записки. | |
| 18 | Защита курсовой работы | |

## 5 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Студенты могут при самостоятельном изучении отдельных тем и вопросов дисциплины пользоваться учебно-наглядными пособиями, учебным оборудованием и методическими разработками кафедры в рабочее время, установленное Правилами внутреннего распорядка работников.

Учебно-методическое обеспечение для самостоятельной работы обучающихся по данной дисциплине организуется:

библиотекой университета:

− библиотечный фонд укомплектован учебной, методической, научной, периодической, справочной и художественной литературой в соответствии с УП и данной РПД;

− имеется доступ к основным информационным образовательным ресурсам, информационной базе данных, в том числе библиографической, возможность выхода в Интернет.

Кафедрой:

− путем обеспечения доступности всего необходимого учебно-методического и справочного материала;

− путем представления сведений о наличии учебно-методической литературы, современных программных средств;

− путем разработки и обеспечения:

- методических рекомендаций, пособий по организации самостоятельной работы студентов;
- заданий для самостоятельной работы;
- тем рефератов и докладов;
- доступа к системе тестирования;
- методических указаний к выполнению лабораторных работ.

Типографией университета:

− помощь авторам в подготовке и издании научной, учебной и методической литературы;

− удовлетворение потребности в тиражировании научной, учебной и методической литературы.

## 6 Образовательные технологии. Практическая подготовка обучающихся. Технологии использования воспитательного потенциала дисциплины

Реализация компетентностного подхода предусматривает широкое использование в образовательном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования универсальных, общепрофессиональных и профессиональных компетенций обучающихся.

Таблица 6.1 – Интерактивные образовательные технологии, используемые при проведении аудиторных занятий

| № | Наименование раздела (лекции, практического или лабораторного занятия) | Используемые интерактивные образовательные технологии | Объем, час. |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 1. | Лекция раздела «Функции» | Разбор конкретных ситуаций | 2 |
| 2. | Программирование с использованием одномерных массивов на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| 3. | Программирование с использованием двумерных массивов на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 4 |
| 4. | Использование подпрограмм и рекурсивного метода на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| 5. | Работа со строками на языке C# (лекция) | Разбор конкретных ситуаций | 2 |
| 6. | Работа со строками на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| 7. | Программирование в графическом режиме на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| 8. | Программирование с использованием структур данных (лабораторная работа). | Творческие задания, работа в малых группах | 2 |

| № | Наименование раздела (лекции, практического или лабораторного занятия) | Используемые интерактивные образовательные технологии | Объем, час. |
|---|---|---|---|
| 9. | Программирование с использованием файлов на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| 10. | Программирование объектно-ориентрованной задачи на C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| Итого: | | | 22 |

Практическая подготовка обучающихся при реализации дисциплины осуществляется путем проведения лабораторных занятий, предусматривающих участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью и направленных на формирование, закрепление, развитие практических навыков и компетенций по направленности (профилю) «Разработка программно-информационных систем» программы бакалавриата.

Практическая подготовка обучающихся при реализации дисциплины организуется в модельных условиях (оборудованных полностью) в лабораториях университета).

Практическая подготовка обучающихся проводится в соответствии с положением П 02.181

Содержание дисциплины обладает значительным воспитательным потенциалом, поскольку в нем аккумулирован исторический и современный научный опыт человечества. Реализация воспитательного потенциала дисциплины осуществляется в рамках единого образовательного и воспитательного процесса и способствует непрерывному развитию личности каждого обучающегося. Дисциплина вносит значимый вклад в формирование профессиональной культуры обучающихся. Содержание дисциплины способствует профессионально-трудовому, культурно-творческому воспитанию обучающихся.

Реализация воспитательного потенциала дисциплины подразумевает:

- целенаправленный отбор преподавателем и включение в лекционный материал, материал для практических и лабораторных занятий содержания, демонстрирующего обучающимся образцы настоящего научного подвижничества создателей и представителей данной отрасли науки, высокого профессионализма ученых, их ответственности за результаты и последствия деятельности человека и общества; примеры подлинной нравственности людей, причастных к развитию науки и производства, а также примеры творческого мышления*;*
- применение технологий, форм и методов преподавания дисциплины, имеющих высокий воспитательный эффект за счет создания условий для взаимодействия обучающихся с преподавателем, другими обучающимися, (командная работа, проектное обучение, разбор конкретных ситуаций и др.)
- личный пример преподавателя, демонстрацию им в образовательной деятельности и общении с обучающимися за рамками образовательного процесса высокой общей и профессиональной культуры.

Реализация воспитательного потенциала дисциплины на учебных занятиях направлена на поддержание в университете единой развивающей образовательной и воспитательной среды. Реализация воспитательного потенциала дисциплины в ходе самостоятельной работы обучающихся способствует развитию в них целеустремленности, инициативности, креативности, ответственности за результаты своей работы – качеств, необходимых для успешной социализации и профессионального становления.

## 7 Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

### 7.1. Перечень компетенций с указанием этапов их формирования в процессе освоения основной профессиональной образовательной программы

Таблица 7.1 – Этапы формирования компетенций

| Код компетенции и наименование компетенции | Этапы формирования компетенций и дисциплины (модули) и практики, при изучении/ прохождении которых формируется данная компетенция | | |
|---|---|---|---|
| | начальный | основной | завершающий |
| 1 | 2 | 3 | 4 |
| ПК – 1 Способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности | Программирование на языках высокого уровня<br>Основы электроники<br>Языки объектно-ориентированного программирования | Производственная практика (научно-исследовательская работа) | Системный анализ<br>Производственная преддипломная практика<br>Выполнение и защита выпускной квалификационной работы |
| ПК – 4 Способен применять различные технологии разработки программного обеспечения | Программирование на языках высокого уровня<br>Языки объектно-ориентированного программирования<br>Конструирование программного обеспечения<br>Проектирование и архитектура программных систем | Системное программное обеспечение<br>Системы реального времени;<br>Офисные технологии<br>Функциональное и логическое программирование<br>Производственная практика (научно-исследовательская работа) | Параллельное программирование;<br>Распределенное программирование;<br>Методы и алгоритмы обработки изображений<br>Тестирование программного обеспечения<br>Web-программирование;<br>Выполнение и защита выпускной квалификационной работы |
| ПК – 11 Способен создавать программные интерфейсы | Программирование на языках высокого уровня | Языки объектно-ориентированного программирования | Web-программирование<br>Проектирование человеко-машинного интерфейса<br>Производственная преддипломная практика |

| | | | |
|---|---|---|---|
| | | | Выполнение и защита выпускной квалификационной работы |

## 7.2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Таблица 7.2 – Показатели и критерии оценивания компетенций, шкала оценивания

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| ПК-1 / начальный | ПК-1.3 Осуществляет сбор, анализ, формулировку и документирование требований заинтересованных лиц<br><br>ПК-1.4 Осуществляет разработку и документирование технического задания на программную систему | **Знать:** в целом сформированные, но неполные знания основных методов и приемов формализации задач, языков формализации функциональных спецификаций, возможностей существующей программно-технической архитектуры, возможностей современных и перспективных средств разработки программных продуктов, технических средств **Уметь:** в целом успешное, но не систематическое умение использовать основные методы и приемы формализации задач, языки формализации функциональных спецификаций, возможности существующей программно-технической архитектуры, | **Знать**: сформированные, но содержащие отдельные пробелы знания основных методов и приемов формализации задач, языков формализации функциональных спецификаций, возможностей существующей программно-технической архитектуры, возможностей современных и перспективных средств разработки программных продуктов, технических средств **Уметь:** успешное, но содержащее отдельные пробелы умение использовать основные методы и инструменты разработки программного обеспечения, возможности существующей программно-технической архитектуры, | **Знать**: сформированные систематические знания основных методов и приемов формализации задач, языков формализации функциональных спецификаций, возможностей существующей программно-технической архитектуры, возможностей современных и перспективных средств разработки программных продуктов, технических средств **Уметь:** сформированное умение применять основные методы и инструменты разработки программного обеспечения, возможности существующей программно-технической архитектуры, использовать возможности |

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| | | использовать возможности современных и перспективных средств разработки программных продуктов, технических средств<br>**Владеть:**<br>в целом успешное, но не полное владение анализом возможностей реализации требований к программному обеспечению, оценкой времени и трудоемкости реализации требований к программному обеспечению, оценкой и согласованием сроков выполнения поставленных задач | использовать возможности современных и перспективных средств разработки программных продуктов, технических средств<br>**Владеть:**<br>успешное, но содержащее отдельные пробелы владение анализом возможностей реализации требований к программному обеспечению, оценкой времени и трудоемкости реализации требований к программному обеспечению, оценкой и согласованием сроков выполнения поставленных задач | современных и перспективных средств разработки программных продуктов, технических средств<br>**Владеть:**<br>сформированное владение навыками использования анализа возможностей реализации требований к программному обеспечению, оценки времени и трудоемкости реализации требований к программному обеспечению, оценки и согласования сроков выполнения поставленных задач |
| ПК-4 / начальный | ПК-4.2<br>Разрабатывает интерфейсы взаимодействия модулей системы<br><br>ПК-4.3<br>Разрабатывает интерфейсы взаимодействия с внешней средой<br><br>ПК-4.4<br>Выполняет тестирование программного обеспечения<br><br>ПК-4.5 | **Знать:**<br>в целом сформированные, но неполные знания синтаксиса выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка программирования, методов и приемов алгоритмизации поставленных задач, отладки программного кода, выявления ошибок в программном коде, типов данных, определяемых пользователем: массивы, строки, | **Знать:**<br>сформированные, но содержащие отдельные пробелы знания синтаксиса выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка программирования, методов и приемов алгоритмизации поставленных задач, отладки программного кода, выявления ошибок в программном коде, типов данных, | **Знать:**<br>сформированные систематические знания синтаксиса выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка программирования, методов и приемов алгоритмизации поставленных задач, отладки программного кода, выявления ошибок в программном коде, типов данных, определяемых |

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| | Разрабатывает структуры данных | структуры; файлы, классы<br>**Уметь:**<br>в целом успешное, но не систематическое умение использовать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования, методы и приемы алгоритмизации поставленных задач, отладку программного кода, выявление ошибок в программном коде, типы данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Владеть:**<br>в целом успешное, но не полное владение созданием программного кода в соответствии с техническим заданием (готовыми спецификациями), разработкой алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов, проверкой и отладкой программного кода, | определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Уметь:**<br>успешное, но содержащее отдельные пробелы умение использовать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования, методы и приемы алгоритмизации поставленных задач, отладку программного кода, выявление ошибок в программном коде, типы данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Владеть:**<br>успешное, но содержащее отдельные пробелы владение навыками создания программного кода в соответствии с техническим заданием (готовыми спецификациями), разработки алгоритмов решения поставленных задач в соответствии с требованиями технического | пользователем: массивы, строки, структуры; файлы, классы<br>**Уметь:**<br>сформированное умение использовать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования, методы и приемы алгоритмизации поставленных задач, отладку программного кода, выявление ошибок в программном коде, типы данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Владеть:**<br>сформированное владение навыками создания программного кода в соответствии с техническим заданием (готовыми спецификациями), разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов, проверки и отладки |

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| | | проектированием структур данных | задания или других принятых в организации нормативных документов, проверки и отладки программного кода, проектирования структур данных. | программного кода, проектирования структур данных. |
| ПК-11/ начальный | ПК-11.2 Разрабатывает алгоритм работы интерфейса<br><br>ПК-11.3 Разрабатывает и отлаживает программные модули для реализации функций интерфейса | Знать:<br>в целом сформированные, но неполные знания методов и средств проектирования программных, правил, алгоритмов и технологий создания тестовых наборов данных<br>**Уметь:**<br>в целом успешное, но не систематическое умение использовать методы и средства проектирования программных, правила, алгоритмы и технологии создания тестовых наборов данных<br>**Владеть:**<br>в целом успешное, но не полное владение навыками проектирования программных интерфейсов, разработки программ для решения задач на языках | Знать:<br>сформированные, но содержащие отдельные пробелы знания методов и средств проектирования программных, правил, алгоритмов и технологий создания тестовых наборов данных<br>**Уметь:**<br>успешное, но содержащее отдельные пробелы умение использовать методы и средства проектирования программных, правила, алгоритмы и технологии создания тестовых наборов данных<br>**Владеть:**<br>успешное, но содержащее отдельные пробелы владения навыками проектирования программных интерфейсов, разработки программ для решения задач на языках | Знать:<br>сформированные систематические знания методов и средств проектирования программных, правил, алгоритмов и технологий создания тестовых наборов данных<br>**Уметь:**<br>сформированное умение использовать методы и средства проектирования программных, правила, алгоритмы и технологии создания тестовых наборов данных<br>**Владеть:**<br>сформированное владение навыками проектирования программных интерфейсов, разработки программ для решения задач на языках |

**7.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения основной профессиональной образовательной программы**

Таблица 7.3 - Паспорт комплекта оценочных средств для текущего контроля успеваемости

| № п/п | Раздел (тема) дисциплины | Код контролируемой компетенции (или ее части) | Технология формирования | Оценочные средства | | Описание шкал оценивания |
|---|---|---|---|---|---|---|
| | | | | наименование | №№ заданий | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 семестр | | | | | | |
| 1 | Основные понятия и определения. | ПК-1, ПК-4, ПК-11 | Лекции СРС | контрольные вопросы к защите лабораторной работы №1 | 1–10 | Согласно табл.7.2 |
| 2 | Программирование линейных алгоритмов на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №2 | 11–20 | Согласно табл.7.2 |
| 3 | Программирование разветвленных алгоритмов на языке C#.. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №3 | 21–30 | Согласно табл.7.2 |
| 4 | Программирование циклических алгоритмов на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №4 | 21–30 | Согласно табл.7.2 |
| 5 | Работа с одномерными массивами на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №5 | 31–40 | Согласно табл.7.2 |
| 6 | Работа с двумерными массивами на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №6 | 31–40 | Согласно табл.7.2 |

| № п/п | Раздел (тема) дисциплины | Код контролируемой компетенции (или ее части) | Технология формирования | Оценочные средства | | Описание шкал оценивания |
|---|---|---|---|---|---|---|
| | | | | наименование | №№ заданий | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 7 | Функции. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №7 | 41–50 | Согласно табл.7.2 |
| 2 семестр | | | | | | |
| 8 | Строки. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №8 | 51–60 | Согласно табл.7.2 |
| 9 | Графический режим работы | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №9 | 61–70 | Согласно табл.7.2 |
| 11 | Структуры и объединения. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №10 | № 81–90 | Согласно табл.7.2 |
| 12 | Файлы в C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №11 | 91–100 101–110 | Согласно табл.7.2 |
| 13 | Обработка исключительных ситуаций. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб СРС | Устный опрос, собеседование | 111–120 | Согласно табл.7.2 |
| 14 | Объектно-ориентированное программирование | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №12 | 121–130 | Согласно табл.7.2 |

| № п/п | Раздел (тема) дисциплины | Код контролируемой компетенции (или ее части) | Технология формирования | Оценочные средства | | Описание шкал оценивания |
|---|---|---|---|---|---|---|
| | | | | наименование | №№ заданий | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 16 | Перспективы использования языков программирования | ПК-1, ПК-4, ПК-11 | Лекции СРС | Устный опрос, собеседование | 141–150 | Согласно табл.7.2 |

Примеры типовых контрольных заданий проведения текущего контроля успеваемости

Вопросы собеседования по разделу (теме) 1. «Основные понятия и определения»

1 Какие этапы работы по созданию приложений можно выполнять с помощью IDE?

2 Для чего предназначен инспектор объектов?

3 Каким образом используется палитра компонентов?

4 Для чего используется панель инструментов?

5 Каким образом переключаются окна проектировщика форм и редактора кода?

6 Каким образом вставляется обработчик событий в код программы?

7 Как создать новый проект?

8 Как сохранить проект?

9 Каким образом открыть существующий проект?

10 Какие основные части содержит простая программа на C#?

11 Каким образом запустить проект только на компиляцию?

12 Каким образом запустить проект на компиляцию и выполнение?

13 Как используются контрольные точки для отладки программ?

14 Как использовать пошаговый режим при отладке программ?

15 Каким образом можно использовать компонент TextBox?

16 Каким образом можно использовать компонент Label?

17 Каким образом можно использовать компонент Button?

Полностью оценочные материалы и оценочные средства для проведения текущего контроля успеваемости представлены в УММ по дисциплине.

Практическая подготовка обучающихся при реализации данной дисциплины организуется, в частности, путем выполнения и защиты курсовой работы (проекта) на одну из предложенных тем

Темы курсовых работ

1. **Программа поиска кратчайшего пути между двумя населенными пунктами в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами. Для двух любых населенных пунктов найти кратчайший путь между ними. Отобразить решение графически, выделив цветом найденный путь.

2. **Программа поиска замкнутого пути между двумя населенными пунктами в системе двусторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. Найти замкнутый путь (с возвратом в тот же населенный пункт) длиной не более 100 км, проходящий через каждую дорогу ровно 1 раз. Отобразить решение графически, выделив цветом найденный путь.
3. **Программа поиска кратчайшего пути между двумя населенными пунктами в системе односторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) односторонними дорогами. Для двух любых населенных пунктов найти кратчайший путь между ними. Отобразить решение графически, выделив цветом найденный путь.
4. **Программа поиска замкнутого пути между двумя населенными пунктами в системе односторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) односторонними дорогами некоторой длины. Найти замкнутый путь (с возвратом в тот же населенный пункт) длиной не более 100 км, проходящий через каждую дорогу ровно 1 раз. Отобразить решение графически, выделив цветом найденный путь.
5. **Программа определения достижимости населенного пункта в системе односторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) односторонними дорогами некоторой длины. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Отобразить решение графически, выделив цветом найденный результат.
6. **Программа определения достижимости населенного пункта в системе двусторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Отобразить решение графически, выделив цветом найденный результат.
7. **Программа определения N-периферии в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. N-периферией называется множество городов, расстояние от которых до выделенного города (столицы) больше N. Определить N-периферию для заданного N. Отобразить решение графически, выделив цветом найденный результат.
8. **Программа определения достижимости города в системе односторонних дорог**. Описание: даны несколько городов, соединенных между собой (произвольным образом) односторонними дорогами некоторой длины. Определить, можно ли проехать из города A в город B таким образом, чтобы посетить город C и не проезжать никакой дороги более одного раза. Отобразить решение графически, выделив цветом найденный результат.
9. **Программа поиска минимальной суммы расстояний между населенными пунктами в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины, причем для любой пары населенных пунктов можно указать соединяющий их путь. Найти такой населенный пункт, для которого сумма расстояний до остальных городов минимальна. Отобразить решение графически, выделив цветом найденный результат.
10. **Программа поиска пути с минимальными затратами между двумя городами в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. За проезд каждой дороги взимается некоторая пошлина. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма пошлин проезжаемых дорог. Отобразить решение графически, выделив цветом найденный результат.
11. **Программа определения достижимости города в изменяемой системе двусторонних дорог.** Описание: даны несколько городов, соединенных между собой (произвольным образом) двусторонними дорогами. Определить, можно ли , построив какие-нибудь 3 новые дороги, из заданного города A добрать до всех других городов. Отобразить решение графически, выделив цветом найденный результат.

12. **Программа определения недостижимости города в изменяемой системе двусторонних дорог**. Описание: даны несколько городов, соединенных между собой (произвольным образом) двусторонними дорогами. Определить, можно ли , закрыв какие-нибудь 3 дороги, добиться того, чтобы нельзя было попасть из города A в город B. Отобразить решение графически, выделив цветом найденный результат.
13. **Программа поиска пути с минимальными затратами между двумя городами в системе односторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) односторонними дорогами некоторой длины. За проезд каждой дороги взимается некоторая пошлина. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма пошлин проезжаемых дорог. Отобразить решение графически, выделив цветом найденный результат.
14. **Программа поиска пути с минимальными затратами между двумя городами в системе двусторонних дорог**. Описание: даны несколько городов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. За проезд каждой дороги взимается некоторая пошлина. Найти путь из города A в город B такой, чтобы величина P была минимальна, где P - сумма пошлин проезжаемых дорог. Если таких путей несколько, то выбрать более короткий путь. Отобразить решение графически, выделив цветом найденный результат.
15. **Программа поиска кратчайшего пути между двумя городами в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Для двух любых населенных пунктов найти кратчайший путь между ними. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
16. **Программа поиска кратчайшего пути между двумя городами в двух системах двусторонних дорог.** Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Для двух любых населенных пунктов найти кратчайший путь между ними. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
17. **Программа поиска пути с минимальными затратами между двумя городами в двух системах односторонних дорог.** Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма денег за проезд дорог. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
18. **Программа поиска пути с минимальными затратами между двумя городами в двух системах двусторонних дорог.** Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма денег за проезд дорог. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
19. **Программа поиска пути с минимальными затратами между двумя городами в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B такой, чтобы величина P была минимальна, где P - сумма денег за проезд дорог. Если таких путей несколько, то выбрать более короткий путь. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
20. **Программа поиска пути с минимальными затратами между двумя городами в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B такой, чтобы величина P была

минимальна, где P - сумма денег за проезд дорог. Если таких путей несколько, то выбрать более короткий путь. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.

21. **Программа определения N-периферии в двух системах двусторонних дорог.** Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. N-периферией называется множество городов, расстояние от которых до выделенного города (столицы) больше N. Определить N-периферию для заданного N. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
22. **Программа определения N-периферии в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. N-периферией называется множество городов, расстояние от которых до выделенного города (столицы) больше N. Определить N-периферию для заданного N. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
23. **Программа поиска минимальной суммы расстояний между населенными пунктами в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов, причем для любой пары населенных пунктов можно указать соединяющий их путь. Найти такой населенный пункт, для которого сумма длин дорог до остальных городов минимальна. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
24. **Программа поиска минимальной суммы расстояний между населенными пунктами в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов, причем для любой пары населенных пунктов можно указать соединяющий их путь. Найти такой населенный пункт, для которого сумма длин дорог до остальных городов минимальна. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат**.**
25. **Программа поиска замкнутого пути между двумя городами в двух системах односторонних дорог.** Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Найти замкнутый путь (с возвратом в тот же населенный пункт), проходящий через каждый город ровно 1 раз. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
26. **Программа поиска замкнутого пути между двумя городами в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Найти замкнутый путь (с возвратом в тот же населенный пункт), проходящий через каждый город ровно 1 раз. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
27. **Программа определения достижимости населенного пункта в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
28. **Программа определения достижимости населенного пункта в двух системах односторонних дорог.** Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
29. **Программа поиска кратчайшего пути шахматной фигуры по клеточному полю.** Найти кратчайший путь передвижения шахматной фигуры (коня, ладьи, слона, ферзя, короля) по

клеточному полю, соединяющий две указанные его клетки. Отобразить решение графически, выделив цветом найденный результат.

30. **Программа определения допустимости хода шахматной фигуры. Описание: задана корректная шахматная позиция**. Определить, является ли допустимым ход шахматной фигуры. Если после хода король атакован, то ход считается не допустимым. Отобразить решение графически, выделив цветом найденный результат.
31. **Программа поиска кратчайшего пути в лабиринте.** Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более трех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. Найти кратчайший путь от входа в лабиринт к его выходу. Отобразить решение графически, выделив цветом найденный результат.
32. **Программа поиска пути в лабиринте. Описание: задан лабиринт, составленный из комнат**. В каждой из комнат имеется не менее одной и не более трех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. Некоторые комнаты объявлены опасными. Найти путь от входа в лабиринт к его выходу, не проходящий через опасные комнаты. Отобразить решение графически, выделив цветом найденный результат.
33. **Программа поиска пути в лабиринте**. Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более четырех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. Одна из комнат объявлена комнатой, приносящей удачу. Определить, есть ли путь от входа в лабиринт к его выходу, не проходящий одну и ту же комнату два раза и проходящий через комнату, приносящую удачу. Отобразить решение графически, выделив цветом найденный результат.
34. **Программа поиска минимального по стоимости пути в лабиринте.** Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более четырех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. За посещение каждой комнаты взимается пошлина в размере от одного до ста рублей. Найти путь от входа в лабиринт к его выходу, чтобы величина P была минимальна, где P - сумма пошлин за посещение комнат лабиринта. Если путей с одинаковыми значениями P несколько, то выбрать более короткий путь. Отобразить решение графически, выделив цветом найденный результат.
35. **Программа поиска пути в лабиринте.** Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более четырех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. В каждой комнате хранятся сокровища: от одной до ста золотых монет. Найти путь от входа в лабиринт к его выходу, не проходящий одну и ту же комнату два раза и позволяющий унести как можно больше сокровищ. Отобразить решение графически, выделив цветом найденный результат.

Требования к структуре, содержанию, объему, оформлению курсовых работ, процедуре защиты, а также критерии оценки определены в:

− стандарте СТУ 04.02.030-2017 «Курсовые работы (проекты). Выпускные квалификационные работы. Общие требования к структуре и оформлению»;

− положении П 02.016-2018 «О балльно-рейтинговой системе оценивания результатов обучения по дисциплинам (модулям) и практикам при освоении обучающимися образовательных программ»;

− методических указаниях по выполнению курсовой работы.

Типовые задания для проведения промежуточной аттестации обучающихся

Промежуточная аттестация по дисциплине проводится в форме экзамена. Экзамен проводится в виде бланкового тестирования.

Примеры типовых заданий для проведения промежуточной аттестации обучающихся

Для тестирования используются контрольно-измерительные материалы (КИМ) – вопросы и задания в тестовой форме, составляющие банк тестовых заданий (БТЗ) по дисциплине, утвержденный в установленном в университете порядке.

Проверяемыми на промежуточной аттестации элементами содержания являются темы дисциплины, указанные в разделе 4 настоящей программы. Все темы дисциплины отражены в КИМ в равных долях (%). БТЗ включает в себя не менее 100 заданий и постоянно пополняется. БТЗ хранится на бумажном носителе в составе УММ и электронном виде в ЭИОС университета.

Для проверки знаний используются вопросы и задания в различных формах:

- закрытой (с выбором одного или нескольких правильных ответов),
- открытой (необходимо вписать правильный ответ).

Результаты практической подготовки (умения, навыки (или опыт деятельности) и компетенции) проверяются с помощью компетентностно-ориентированных задач (ситуационных, производственных) и различного вида конструкторов.

Все задачи являются многоходовыми. Некоторые задачи, проверяющие уровень сформированности компетенций, являются многовариантными. Часть умений, навыков и компетенций прямо не отражена в формулировках задач, но они могут быть проявлены обучающимися при их решении.

В каждый вариант КИМ включаются задания по каждому проверяемому элементу содержания во всех перечисленных выше формах и разного уровня сложности. Такой формат КИМ позволяет объективно определить качество освоения обучающимися основных элементов содержания дисциплины и уровень сформированности компетенций.

Примеры типовых заданий для проведения промежуточной аттестации обучающихся

Задание в закрытой форме:

Оператор множественного выбора switch/case можно заменить набором операторов ( 2 балла) :

Выберите один ответ:

- for
- if else if
- while
- goto

.....

Задание в открытой форме:

В результате работы фрагмента программы ( 2 балла)

```
int a = 1, b = 1;
a + = + + b;
```

Выберите один ответ:

- a=4
- a=1
- a=2
- a=3

.....

Компетентностно-ориентированная задача:

Дан двумерный массив целых чисел. Для каждой строки, упорядоченной по возрастанию или по убыванию, найти максимальный элемент.

.....

Полностью оценочные материалы и оценочные средства для проведения промежуточной аттестации обучающихся представлены в УММ по дисциплине.

### 7.4 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций, регулируются следующими нормативными актами университета:

- Положение П 02.016 – 2018 О балльно-рейтинговой системе оценивания результатов обучения по дисциплинам (модулям) и практикам при освоении обучающимися образовательных программ;
- методические указания, используемые в образовательном процессе, указанные в списке литературы.

Для текущего контроля по дисциплине в рамках действующей в университете балльно-рейтинговой системы применяется следующий порядок начисления баллов:

Таблица 7.4 – Порядок начисления баллов в рамках БРС

| Форма контроля | Минимальный балл | | Максимальный балл | |
|---|---|---|---|---|
| | балл | примечание | балл | примечание |
| 1 семестр | | | | |
| Лекция №1, 2 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторная работа № 1–2 | 4 | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| Лекция № 3, 4 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторные работы № 3–4 | 4 | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| Лекция № 5, 6 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторные работа № 5– 6 | 5 | Выполнил, но «не защитил» | 10 | Выполнил и «защитил» |
| Лекция № 7– 8 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторные работы № 7 | 5 | Выполнил, но «не защитил» | 10 | Выполнил и «защитил» |
| СРС | 2 | | 4 | |
| Итого | 24 | | 48 | |
| Посещаемость | | | 16 | |
| Экзамен | | | 36 | |
| Итого | 24 | | 100 | |
| 2 семестр | | | | |
| Лекция №9 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторная работа № 5 | 4 | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| Лекция № 10 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторные работы № 6, 7 | 4 | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| Лекция № 11 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторная работа № 8 | 5 | Выполнил, но «не защитил» | 10 | Выполнил и «защитил» |
| Лекция № 12 | 1 | Материал усвоен менее чем на 50% | 2 | Материал усвоен более чем на 50% |
| Лабораторная работа № 9 | 5 | Выполнил, но «не защитил» | 10 | Выполнил и «защитил» |
| СРС | 2 | | 4 | |
| Итого | 24 | | 48 | |
| Посещаемость | | | 16 | |
| Экзамен | | | 36 | |
| Итого | 24 | | 100 | |

Для *промежуточной аттестации* обучающихся, проводимой в виде тестирования, используется следующая методика оценивания знаний, умений, навыков и (или) опыта деятельности. В каждом варианте КИМ –16 заданий (15 вопросов и одна задача).

Каждый верный ответ оценивается следующим образом:

– задание в закрытой форме – 2 балла,
– задание в открытой форме – 2 балла,
– решение компетентностно-ориентированной задачи – 6 баллов.

Максимальное количество баллов за тестирование –36 баллов.

## 8 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

### 8.1 Основная учебная литература

1. Белов В. Г. Основы программирования на языке C++ Builder [Текст]: учеб. пособие / В. Г. Белов, Т. М. Белова; Юго-Зап. гос. ун-т. – Курск, 2015. – 160 с.
2. Белов В. Г. Основы программирования на языке C++ Builder [Электронный ресурс]: учеб. пособие / В. Г. Белов, Т. М. Белова; Юго-Зап. гос. ун-т. – Курск, 2015. – 160 с.
3. Иванова Г. С. Технология программирования [Текст] : учебник / Г. С. Иванова. - М. : Кнорус, 2011. - 336 с.

### 8.2 Дополнительная учебная литература

1. Глушаков, С. В. Язык программирования C++ [Текст] : руководство / С. В. Глушаков, А. В. Коваль, С. В. Смирнов. - М. : АСТ, 2004. - 500 с.
2. Фридман, А. Л. Язык программирования Си. Курс лекций [Текст] : учебное пособие / А. Л. Фридман. - 2-е изд., испр. - М. : ИНТУИТ. РУ Интернет-университет Информационных Технологий, 2004. - 264 с.
3. Страуструп Б. Программирование. Принципы и практика использования C# [Текст] / Б. Страуструп. – М.: ИД «Вильямс», 2011. – 1206 с.
4. Лафоре, Роберт . Объектно-ориентированное программирование в C# [Текст] / Р. Лафоре. − 4-е изд. - СПб. [и др.] : Питер, 2012. - 928 с. :

### 8.3 Перечень методических указаний

1. Изучение интегрированной среды MS Visual Studio C# [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплинам «Программирование на языках высокого уровня», «Языки программирования» для студентов направлений подготовки 09.03.04 «Программная инженерия», 10.03.01 «Информационная безопасность», 10.05.02 «Информационная безопасность телекоммуникационных систем»/ Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. Курск: ЮЗГУ, 2019. – 32 с.

2. Программирование линейных алгоритмов на языке C# [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплинам «Программирование на языках высокого уровня», «Языки программирования» для студентов направлений подготовки 09.03.04 «Программная инженерия», 10.03.01 «Информационная безопасность», 10.05.02 «Информационная безопасность телекоммуникационных систем»/ Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2019. – 26 с.

3. Программирование разветвленных алгоритмов на языке C# [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплинам «Программирование на языках высокого уровня», «Языки программирования» для

студентов направлений подготовки 09.03.04 «Программная инженерия», 10.03.01 «Информационная безопасность», 10.05.02 «Информационная безопасность телекоммуникационных систем»/ Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. Курск: ЮЗГУ, 2019. – 21 с.

4. Программирование циклических алгоритмов на языке С++ [Электронный ресурс] : методические указания по выполнению лабораторной работы для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 18 с.
5. Программирование на языках высокого уровня [Электронный ресурс] : методические рекомендации по выполнению курсовой работы для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 20 с.
6. Программирование с использованием одномерных массивов на языке С++ [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплине «Программирование на языках высокого уровня» для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 17 с.
7. Программирование с использованием двумерных массивов на языке С++ [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплине «Программирование на языках высокого уровня» для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 16 с.
8. Программирование с использованием функций на языке С++ [Электронный ресурс]: методические указания по выполнению лабораторной работы по дисциплине «Программирование на языках высокого уровня» для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 24 с.
9. Методические рекомендации по организации самостоятельной работы студентов по дисциплинам учебных планов направлений подготовки 09.03.04 и 09.04.04 [Электронный ресурс] / Юго-Зап. гос. ун-т; сост. Р. А. Томакова. − Курск : ЮЗГУ, 2017. − 55 с.

### 8.4 Другие учебно-методические материалы

Отраслевые научно-технические журналы в библиотеке университета:

- Вестник компьютерных и информационных технологий
- Информационные технологии
- Информационные технологии и вычислительные системы
- Программирование
- Программные продукты и системы
- Искусственный интеллект и принятие решений
- Информатика и её применения

## 9 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- Образовательный сайт Life-prog: http://www.life-prog.ru.
- Электронно-библиотечная система «Университетская библиотека online»: http://www.biblioclub.ru.
- Электронная библиотека ЮЗГУ: http://www.lib.swsu.ru.
- Электронная библиотека: http://www.window.edu.ru
- Интернет-университет информационных технологий http: //www.intuit.ru/
- Виртуальный музей истории вычислительной техники в картинках http://www.computerhistory.narod.ru
- Энциклопедия отечественной информатики
- http://www.computer-museum.ru/
- http://www.i-exam.ru/ – Официальный сайт Федерального интернет-экзамена в сфере профессионального образования НИИ Мониторинга качества образования [Электронный ресурс].

## 10 Методические указания для обучающихся по освоению дисциплины

Основными видами аудиторной работы студента при изучении дисциплины являются лекции, лабораторные и практические занятия. Студент не имеет права пропускать занятия без уважительных причин.

На лекциях излагаются и разъясняются основные понятия темы, связанные с ней теоретические и практические проблемы, даются рекомендации для самостоятельной работы. В ходе лекции студент должен внимательно слушать и конспектировать материал.

Изучение наиболее важных тем или разделов дисциплины завершают лабораторные занятия, которые обеспечивают: контроль подготовленности студента; закрепление учебного материала; приобретение опыта устных публичных выступлений, ведения дискуссии, в том числе аргументации и защиты выдвигаемых положений и тезисов.

Лабораторному занятию предшествует самостоятельная работа студента, связанная с освоением материала, полученного на лекциях, и материалов, изложенных в учебниках и учебных пособиях, а также литературе, рекомендованной преподавателем.

Самостоятельную работу следует начинать с первых занятий. От занятия к занятию нужно регулярно прочитывать конспект лекций, знакомиться с соответствующими разделами учебника, читать и конспектировать литературу по каждой теме дисциплины. Самостоятельная работа дает студентам возможность равномерно распределить нагрузку, способствует более глубокому и качественному усвоению учебного материала. В случае необходимости студенты обращаются за консультацией к преподавателю по вопросам дисциплины «Информатика» с целью усвоения и закрепления компетенций.

По согласованию с преподавателем или по его заданию студенты могут готовить рефераты по отдельным темам дисциплины, выступать на занятиях с докладами. Основу докладов составляет, как правило, содержание подготовленных студентами рефератов.

Качество учебной работы студентов преподаватель оценивает по результатам тестирования, собеседования, защиты отчетов по лабораторным работам, а также по результатам докладов.

## 11 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Windows: MSDN subscriptions, Договор IT000012385 MS Visual Studio Community Edition 2017: Бесплатная, Freeware лицензия. Свободное программное обеспечение: Mozilla Firefox: GNU GPL LibreOffice: GNU LGPL.

## 12 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Для осуществления практической подготовки обучающихся при реализации дисциплины используются оборудование и технические средства обучения:

- учебная аудитория для проведения занятий лекционного типа и лаборатории кафедры программной инженерии, оснащенные учебной мебелью: столы, стулья для обучающихся; стол, стул для преподавателя;
- электронная доска. Мультимедиа центр: ноутбук ASUS X50VL PMD-T2330/14"/1024Mb/160Gb/сумка/проектор inFocus IN24+;
- рабочие станции (ПЭВМ) PremiumP43/ E6300/ 4Гб DDR2/ 320 Гб / DVD RW/ AcerV223HQb с прогр. обеспеч. (27002.40).

## 13 Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

При обучении лиц с ограниченными возможностями здоровья учитываются их индивидуальные психофизические особенности. Обучение инвалидов осуществляется также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).

Для лиц с нарушением слуха возможно предоставление учебной информации в визуальной форме (краткий конспект лекций; тексты заданий, напечатанные увеличенным шрифтом), на аудиторных занятиях допускается присутствие ассистента, а также сурдопереводчиков и тифлосурдопереводчиков. Текущий контроль успеваемости осуществляется в письменной форме: обучающийся письменно отвечает на вопросы, письменно выполняет практические задания. Доклад (реферат) также может быть представлен в письменной форме, при этом требования к содержанию остаются теми же, а требования к качеству изложения материала (понятность, качество речи, взаимодействие с аудиторией и т. д.) заменяются на соответствующие требования, предъявляемые к письменным работам (качество оформления текста и списка литературы, грамотность, наличие иллюстрационных материалов и т.д.). Промежуточная аттестация для лиц с нарушениями слуха проводится в письменной форме, при этом используются общие критерии оценивания. При необходимости время подготовки к ответу может быть увеличено.

Для лиц с нарушением зрения допускается аудиальное предоставление информации, а также использование на аудиторных занятиях звукозаписывающих устройств (диктофонов и т.д.). Допускается присутствие на занятиях ассистента

(помощника), оказывающего обучающимся необходимую техническую помощь. Текущий контроль успеваемости осуществляется в устной форме. При проведении промежуточной аттестации для лиц с нарушением зрения тестирование может быть заменено на устное собеседование по вопросам.

Для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, на аудиторных занятиях, а также при проведении процедур текущего контроля успеваемости и промежуточной аттестации могут быть предоставлены необходимые технические средства (персональный компьютер, ноутбук или другой гаджет); допускается присутствие ассистента (ассистентов), оказывающего обучающимся необходимую техническую помощь (занять рабочее место, передвигаться по аудитории, прочитать задание, оформить ответ, общаться с преподавателем).

**14 Лист дополнений и изменений, внесенных в рабочую программу дисциплины**

| Номер изменения | Номера страниц | | | | Всего стра-ниц | Дата | Основание для изменения и подпись лица, проводившего изменения |
|---|---|---|---|---|---|---|---|
| | изменённых | заменённых | аннулированных | новых | | | |
| 1 | 3, 10, 14, 15 | | | | 4 | | |

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Емельянов Сергей Геннадьевич
Должность: ректор
Дата подписания: 31.12.2020 13:36:24
Уникальный программный ключ:
9ba7d3e34c012eba476ffd2d064cf2781953be730df2374d16f3c0ce536f0fc6

# Аннотация к рабочей программе

дисциплины «**Программирование на языках высокого уровня**»

**Цель дисциплины**

Формирование у студентов теоретико-прикладных представлений об основах алгоритмизации и проектирования программ, приемов программирования на конкретных алгоритмических языках, основ организации вычислительного процесса на ЭВМ; привитие навыков анализа корректности программ, их отладки и тестирования с использованием современных сред программирования.

**Задачи дисциплины:**

– изучение технологий разработки программ на языках высокого уровня;

– приобретение навыков практического использования языков программирования высокого уровня;

– приобретение навыков хранения и обработки текстовой и числовой информации.

**Компетенции, формируемые в результате освоения дисциплины:**

– способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности (ПК-1);

– способен применять различные технологии разработки программного обеспечения (ПК-4);

– способен создавать программные интерфейсы (ПК-11).

**Разделы дисциплины:**

1. Основные понятия и определения.
2. Программирование линейных алгоритмов на языке C#.
3. Программирование разветвленных алгоритмов на языке C#.
4. Программирование циклических алгоритмов на языке C#.
5. Работа с одномерными массивами на языке C#.
6. Работа с двумерными массивами на языке C#.
7. Функции.
8. Виды строк в C#. Использование библиотеки функций работы со строками в C#.
9. Графический режим работы.
10. Структуры и объединения.
11. Файлы в C#. Использование библиотеки функций работы с файлами в C#.
12. Обработка исключительных ситуаций. Примеры.
13. Объектно-ориентированное программирование, инкапсуляция.
14. Объектно-ориентированное программирование, полиморфизм, наследование.
15. Заключение. Перспективы развития и использования языков программирования

МИНОБРНАУКИ РОССИИ

Юго-Западный государственный университет

УТВЕРЖДАЮ:
И. о. декана факультета
*(наименование ф-та, полностью)*
фундаментальной и прикладной информатики
_____Т.А. Ширабакина
*(подпись, инициалы, фамилия)*
«20» 06 2019 г.

# РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование на языках высокого уровня
*(наименование дисциплины)*

ОПОП ВО 09.03.04 Программная инженерия,
*шифр и наименование направления подготовки (специальности)*

направленность (профиль) «Разработка программно-информационных систем
*наименование направленности (профиля, специализации)*

форма обучения заочная
*(очная, очно-заочная, заочная)*

Курск – 2019

Рабочая программа дисциплины составлена в соответствии с ФГОС-3++ – бакалавриат по направлению подготовки (специальности) 09.03.04 «Программная инженерия» на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета (протокол №7 «29» марта 2019г.).

Рабочая программа дисциплины обсуждена и рекомендована к реализации в образовательном процессе для обучения студентов по ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем» на заседании кафедры программной инженерии №13 «20» 06 2019 г.

Зав. кафедрой _____ к.т.н., доцент Малышев А. В.
Разработчик программы _____ к.т.н., доцент Белова Т. М.
Директор научной библиотеки _____ Макаровская В. Г.

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол №7«25»02 2020г., на заседании кафедры программной инженерии «10» 06 2020г., протокол №11.

Зав. кафедрой _____ Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол №6«26»02 2021г., на заседании кафедры программной инженерии «02» 07 2021г., протокол №12.

Зав. кафедрой _____ Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол №7«28»02 2022г., на заседании кафедры программной инженерии «17» 06 2022г., протокол №11.

Зав. кафедрой _____ Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 6 «26» 02 2021 г. на заседании кафедры

ПИ, №11 от 13.06.2023

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 7 «28» 02 2022 г. на заседании кафедры

ПИ, №11 от 10.06.2024

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 9 «31» 03 2025 г. на заседании кафедры

ПИ №12 от 30.06.2025

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № 8 «24» 02 2026 г. на заседании кафедры

ПИ №12 от 30.06.2026

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой Малышев

Рабочая программа дисциплины пересмотрена, обсуждена и рекомендована к реализации в образовательном процессе на основании учебного плана ОПОП ВО 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем», одобренного Ученым советом университета протокол № «» 20 г. на заседании кафедры

*(наименование кафедры, дата, номер протокола)*

Зав. кафедрой

## 1 Цель и задачи дисциплины. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы

### 1.1 Цель дисциплины

Формирование у студентов теоретико-прикладных представлений об основах алгоритмизации и проектирования программ, приемов программирования на конкретных алгоритмических языках, основ организации вычислительного процесса на ЭВМ; привитие навыков анализа корректности программ, их отладки и тестирования с использованием современных сред программирования.

### 1.2 Задачи дисциплины

- изучение технологий разработки программ на языках высокого уровня;
- приобретение навыков практического использования языков программирования высокого уровня;
- приобретение навыков хранения и обработки текстовой и числовой информации.

### 1.3 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы

Таблица 1.3 – Результаты обучения по дисциплине

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| ПК-1 | Способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности | ПК-1.3 Осуществляет сбор, анализ, формулировку и документирование требований заинтересованных лиц | ***Знать:*** методы и приемы формализации задач, языки формализации функциональных спецификаций ***Уметь:*** использовать методы и приемы формализации задач, использовать языки формализации функциональных спецификаций ***Владеть:*** составлением формализованных описаний решений поставленных задач в |

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| | | | соответствии с требованиями технического задания или других принятых в организации нормативных документов |
| | | ПК-1.4 Осуществляет разработку и документирование технического задания на программную систему | ***Знать:*** возможности существующей программно-технической архитектуры, возможности современных и перспективных средств разработки программных продуктов, технических средств ***Уметь:*** использовать возможности существующей программно-технической архитектуры, использовать возможности современных и перспективных средств разработки программных продуктов, технических средств ***Владеть:*** анализом возможностей реализации требований к программному обеспечению, оценка времени и трудоемкости реализации требований к программному обеспечению, оценкой и согласованием сроков выполнения поставленных задач |
| ПК-4 | Способен применять различные технологии разработки программного обеспечения | ПК-4.2 Разрабатывает интерфейсы взаимодействия модулей системы | ***Знать:*** синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования ***Уметь:*** использовать синтаксис выбранного языка программирования, особенности |

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| | | | программирования на этом языке, стандартные библиотеки языка программирования<br>***Владеть:***<br>созданием программного кода в соответствии с техническим заданием (готовыми спецификациями) |
| | | ПК-4.3<br>Разрабатывает интерфейсы взаимодействия с внешней средой | ***Знать:***<br>методы и приемы алгоритмизации поставленных задач<br>***Уметь:***<br>использовать методы и приемы алгоритмизации поставленных задач<br>***Владеть:***<br>разработкой алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов |
| | | ПК-4.4<br>Выполняет тестирование программного обеспечения | ***Знать:***<br>методы и приемы отладки программного кода<br>***Уметь:***<br>использовать методы и приемы отладки программного кода, выявлять ошибки в программном коде<br>***Владеть:***<br>проверкой и отладкой программного кода |
| | | ПК-4.5<br>Разрабатывает структуры данных | ***Знать:***<br>типы данных, определяемые пользователем: массивы, строки, структуры; файлы, классы<br>***Уметь:***<br>использовать типы данных, определяемые пользователем: массивы, строки, структуры; файлы, классы |

| *Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)* | | *Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной* | *Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций* |
|---|---|---|---|
| *код компетенции* | *наименование компетенции* | | |
| | | | ***Владеть:*** проектированием структур данных |
| ПК-11 | Способен создавать программные интерфейсы | ПК-11.2 Разрабатывает алгоритм работы интерфейса | ***Знать:*** методы и средства проектирования программных интерфейсов ***Уметь:*** использовать методы и средства проектирования программных интерфейсов ***Владеть:*** проектированием программных интерфейсов |
| | | ПК-11.3 Разрабатывает и отлаживает программные модули для реализации функций интерфейса | ***Знать:*** правила, алгоритмы и технологии создания тестовых наборов данных ***Уметь:*** использовать правила, алгоритмы и технологии создания тестовых наборов данных ***Владеть:*** практическими навыками разработки программ для решения задач на языках высокого уровня |

## 2 Указание места дисциплины в структуре основной профессиональной образовательной программы

Дисциплина «Программирование на языках высокого уровня» входит в часть, формируемую участниками образовательных отношений блока 1 «Дисциплины (модули)» основной профессиональной образовательной программы – программы бакалавриата 09.03.04 «Программная инженерия», направленность (профиль) «Разработка программно-информационных систем». Дисциплина изучается на 1 курсе в 1, 2 и 3 семестрах.

## 3 Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 11 зачетных единиц (ЗЕТ), 396 часов.

Таблица 3 – Объём дисциплины

| Виды учебной работы | Всего, часов |
|---|---|
| Общая трудоемкость дисциплины | 396 |
| Контактная работа обучающихся с преподавателем по видам учебных занятий (всего) | 33,24 |
| в том числе: | |
| лекции | 12 |
| лабораторные занятия | 12, из них практическая подготовка – 2 |
| практические занятия | 8 |
| Самостоятельная работа обучающихся (всего) | 344,76 |
| Контроль (подготовка к экзамену) | 18 |
| Контактная работа по промежуточной аттестации (всего АттКР) | 1,24 |
| в том числе: | |
| зачет | не предусмотрен |
| зачет с оценкой | не предусмотрен |
| курсовая работа (проект) | 1 |
| экзамен (включая консультацию перед экзаменом) | 2,24 |

## 4 Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

### 4.1 Содержание дисциплины

Таблица 4.1.1 – Содержание дисциплины, структурированное по темам (разделам)

| № п/п | Раздел (тема) дисциплины | Содержание |
|---|---|---|
| 1 | 2 | 3 |
| 1 семестр | | |
| 1 | Основные понятия и определения. | Понятие алгоритма. Формы представления алгоритмов. ГОСТы ЕСПД. Интегрированная среда разработки программ на языке C#. Приемы отладки программ. |
| 2 | Программирование линейных алгоритмов на языке C#. | Типы данных. Структура программы. Оператор присваивания. Префиксные и постфиксные формы. |
| 3 | Программирование разветвленных алгоритмов на языке C#. | Условные операторы. Приоритеты выполнения арифметических и логических операций. |
| 4 | Программирование циклических алгоритмов на языке C#. | Операторы цикла. Цикл с предусловием, цикл с постусловием, цикл со счетчиком. |

| № п/п | Раздел (тема) дисциплины | Содержание |
|---|---|---|
| 1 | 2 | 3 |
| 5 | Работа с одномерными массивами на языке C#. | Массивы, ввод и вывод одномерных массивов. Статические и динамические массивы. Поиск минимального и максимального элементов. |
| 6 | Работа с двумерными массивами на языке C#. | Массивы, ввод и вывод двумерных массивов. Статические и динамические массивы. Поиск минимального и максимального элементов. |
| 2 семестр | | |
| 7 | Функции. | Структура программы с подпрограммами на языке C#. Структура функций. Параметры. Передача массивов в функцию. Параметры по умолчанию. Рекурсия. |
| 8 | Строки. | Виды строк в C#. Использование библиотеки функций работы со строками в C#. |
| 9 | Графический режим работы. | Понятие канвы. Графические примитивы. |
| 10 | Структуры и объединения. | Описания структур и объединений. Отличия. Примеры использования. |
| 11 | Файлы в C#. | Использование библиотеки функций работы с файлами в C#. |
| 3 семестр | | |
| 12 | Обработка исключительных ситуаций. | Назначение, варианты использования. Примеры. |
| 13 | Объектно-ориентированное программирование. | Инкапсуляция. Правила построения классов. Примеры. |
| 14 | Объектно-ориентированное программирование. | Полиморфизм, наследование. Раннее связывание и позднее связывание. Простое и множественное наследование. Абстрактные классы. |
| 15 | Заключение. | Перспективы развития и использования языков программирования |

Таблица 4.1.2 - Содержание дисциплины и ее методическое обеспечение

| № п/п | Раздел (тема) дисциплины | Виды деятельности | | | Учебно-методические материалы | Формы текущего контроля успеваемости (*по неделям семестра*). | Компетенции |
|---|---|---|---|---|---|---|---|
| | | лек., час | № лаб. | № пр. | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 семестр | | | | | | | |
| 1. | Основные понятия и определения. | 4 | 1 | – | У1, У2, МУ1 | КО, С | ПК-1, ПК-4, ПК-11 |
| 2. | Программирование линейных алгоритмов на языке C#. | 2 | 2 | – | У1, У2, МУ1 | КО, С | ПК-1, ПК-4, ПК-11 |

| № п/п | Раздел (тема) дисциплины | Виды деятельности | | | Учебно-методические материалы | Формы текущего контроля успеваемости (*по неделям семестра*). | Компетенции |
|---|---|---|---|---|---|---|---|
| | | лек., час | № лаб. | № пр. | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 3. | Программирование разветвленных алгоритмов на языке C#. | 2 | 3 | – | У1, У2, МУ2 | КО, С | ПК-1, ПК-4, ПК-11 |
| 4. | Программирование циклических алгоритмов на языке C#. | 2 | 4 | – | У1, У2, МУ3 | КО, С | ПК-1, ПК-4, ПК-11 |
| 5. | Работа с одномерными массивами на языке C#. | 2 | 5 | – | У1, У2, МУ4 | КО, С | ПК-1, ПК-4, ПК-11 |
| 6. | Работа с двумерными массивами на языке C#. | 2 | 6 | – | У1, У2, МУ5 | КО, С | ПК-1, ПК-4, ПК-11 |
| 2 семестр | | | | | | | |
| 7. | Функции. | 4 | 7 | – | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 8. | Виды строк в C#. Использование библиотеки функций работы со строками в C#. | 2 | 1 | 1 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 9. | Графический режим работы. | 2 | 2 | | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 10. | Структуры и объединения. | 2 | 3 | | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 11. | Файлы в C#. Использование библиотеки функций работы с файлами в C#. | 2 | 4 | | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 3 семестр | | | | | | | |
| 12. | Обработка исключительных ситуаций. Примеры. | 2 | 5 | 1 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 13. | Объектно-ориентированное программирование, инкапсуляция. | 2 | 6 | 2 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 14. | Объектно-ориентированное программирование, полиморфизм, наследование. | 2 | 7 | 3 | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |
| 15. | Заключение. Перспективы развития и использования языков программирования. | 2 | 7 | | У1, У2, МУ6 | КО, С | ПК-1, ПК-4, ПК-11 |

КО –контрольный опрос, С – собеседование.

## 4.2 Лабораторные работы и (или) практические занятия

### 4.2.1 Лабораторные работы

Таблица 4.2.1 — Лабораторные работы

| № | Наименование лабораторной работы | Объём, час. |
|---|---|---|
| 1. | Изучение основных режимов работы в интегрированной среде C#. Программирование простого примера. | 1 |
| 2. | Программирование линейных алгоритмов на языке C#. | 1 |
| 3. | Программирование разветвленных алгоритмов на языке C#. | 1 |
| 4. | Программирование циклических алгоритмов на языке C#. | 1 |
| 5. | Программирование с использованием одномерных массивов на C#. | 1 |
| 6. | Программирование с использованием двумерных массивов на языке C#. | 1 |
| 7. | Программирование с использованием функций на языке C#. | 1, из них практическая подготовка – 1 |
| 8. | Программирование с использованием строк на языке C#. | 1 |
| 9. | Программирование с использование структур данных на языке C#. | 1 |
| 10. | Программирование в графическом режиме на языке C#. | 1 |
| 11. | Программирование с использованием файлов на языке C#. | 1, из них практическая подготовка – 1 |
| 12. | Программирование с использованием объектно-ориентрованной технологии на языке C# | 1 |
| Итого | | 12 |

### 4.2.2 Практические занятия

Таблица 4.2.2 — Практические занятия

| № | Наименование практического занятия | Объём, час. |
|---|---|---|
| 1. | Программирование с использованием функций на языке C#. | 2 |
| 2. | Программирование с использованием строк на языке C#. | 2 |
| 3. | Программирование с использованием файлов на языке C#. | 2 |
| 4. | Программирование с использованием объектно-ориентрованной технологии на языке C# | 2 |
| Итого | | 8 |

### 4.3 Самостоятельная работа студентов (СРС)

Таблица 4.3 - Самостоятельная работа студентов

| № п.п | Наименование раздела дисциплины | Срок выполнения | Время, затрачиваемое на выполнение СРС, час. |
|---|---|---|---|
| 1 семестр | | | |
| 1 | Основные понятия и определения. | 1,2 неделя | 15 |
| 2 | Программирование линейных алгоритмов на языке C#. | 3,4 неделя | 15 |
| 3 | Программирование разветвленных алгоритмов на языке C#.. | 5,6 неделя | 15 |

| № п.п | Наименование раздела дисциплины | Срок выполнения | Время, затрачиваемое на выполнение СРС, час. |
|---|---|---|---|
| 4 | Программирование циклических алгоритмов на языке C#. | 7,8 недели | 15 |
| 5 | Работа с одномерными массивами на языке C#. | 9,10 неделя | 20 |
| 6 | Работа с двумерными массивами на языке C#. | 11,12 недели | 20 |
| 2 семестр | | | |
| 7 | Функции. | 1–3недели | 25 |
| 8 | Строки. | 4–6 недели | 25 |
| 9 | Графический режим работы. | 10–12 недели | 20 |
| 10 | Структуры и объединения. | 13–18 недели | 20,88 |
| 3 семестр | | | |
| 11 | Файлы в C#. | 1–3недели | 25 |
| 12 | Обработка исключительных ситуаций. | 4–10 недели | 20 |
| 13 | Объектно-ориентированное программирование. | 11–18 недели | 28,88 |
| 14 | Курсовая работа | 1–17 недели | 80 |
| Итого | | | 344,76 |

Курсовая работа

Курсовая работа выполняется на первом курсе во 2 семестре.

Отчетность по курсовому проектированию состоит из программной документации и созданного программного продукта в виде исполняемого файла.

Программная документация состоит из 3 разделов: технического задания, технического проекта, рабочего проекта, которые оформляются согласно ЕСПД, ГОСТ 19.701-90 (ISO 5807-85), ГОСТ 2.304-88.

Техническое задание содержит обозначение и наименование программного изделия, краткую характеристику области применения, назначение разработки, назначение разработки, требования заказчика (пользователя). Примерное количество страниц 10.

На стадии технического проекта производится дальнейшая функциональная декомпозиция программного изделия. При этом функции, определенные на стадии технического задания, разбиваются на модули. Такие модули проектируются на уровне внешнего описания, т.е. для каждого из них определяется функциональность с точки зрения пользователя. Примерное количество страниц 10.

Стадия рабочего проекта является заключительной как в конструировании программного изделия, так и в его изготовлении. В процессе конструирования окончательно определяется структура всего программного изделия. Определяется структура каждого фактического программного модуля в виде схемы его алгоритма. Для всего изделия в целом и для каждого модуля рассчитываются и конструируются тесты. Описание программы представляет собой внутреннее описание программного изделия. Примерное количество страниц 20.

Примерный график выполнения по неделям (с указанием контрольных мероприятий) – 80 часов.

| № недели | Наименование и содержание этапа | Примечание |
|---|---|---|
| 1 | Изучение задания на курсовую работу и его согласование. | |
| 2 | Изучение метода, подлежащего реализации. | |
| 3,4 | Составления технического задания. | |
| 5 | Утверждение технического задания. | Контрольная точка |

| № недели | Наименование и содержание этапа | Примечание |
|---|---|---|
| 6 | Разработка технического проекта. Функциональное проектирование. | |
| 7 | Разработка технического проекта. Состав и взаимосвязь функциональных модулей. | |
| 8 | Утверждение и оформление технического проекта. | Контрольная точка. |
| 9 | Разработка рабочего проекта. Определение состава программных модулей и их взаимосвязей. | |
| 10 | Разработка рабочего проекта. Составление схем алгоритмов верхнего уровня. | |
| 11 | Разработка тестов для алгоритмов верхнего уровня. | Контрольная точка |
| 12 | Программирование и отладка программ верхнего уровня. | |
| 13 | Составление схем алгоритмов нижнего уровня. | |
| 14 | Разработка тестов для алгоритмов нижнего уровня. | |
| 15 | Программирование и отладка алгоритмов нижнего уровня. | Контрольная точка |
| 16 | Тестирование и отладка всех модулей. | |
| 17 | Оформление пояснительной записки. | |
| 18 | Защита курсовой работы | |

## 5 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Студенты могут при самостоятельном изучении отдельных тем и вопросов дисциплины пользоваться учебно-наглядными пособиями, учебным оборудованием и методическими разработками кафедры в рабочее время, установленное Правилами внутреннего распорядка работников.

Учебно-методическое обеспечение для самостоятельной работы обучающихся по данной дисциплине организуется:

библиотекой университета:

− библиотечный фонд укомплектован учебной, методической, научной, периодической, справочной и художественной литературой в соответствии с УП и данной РПД;

− имеется доступ к основным информационным образовательным ресурсам, информационной базе данных, в том числе библиографической, возможность выхода в Интернет.

Кафедрой:

− путем обеспечения доступности всего необходимого учебно-методического и справочного материала;

− путем представления сведений о наличии учебно-методической литературы, современных программных средств;

− путем разработки и обеспечения:

- методических рекомендаций, пособий по организации самостоятельной работы студентов;
- заданий для самостоятельной работы;
- тем рефератов и докладов;
- доступа к системе тестирования;

- методических указаний к выполнению лабораторных работ.

Типографией университета:

− помощь авторам в подготовке и издании научной, учебной и методической литературы;

− удовлетворение потребности в тиражировании научной, учебной и методической литературы.

## 6 Образовательные технологии. Практическая подготовка обучающихся. Технологии использования воспитательного потенциала дисциплины

Реализация компетентностного подхода предусматривает широкое использование в образовательном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования универсальных, общепрофессиональных и профессиональных компетенций обучающихся.

Таблица 6.1 – Интерактивные образовательные технологии, используемые при проведении аудиторных занятий

| № | Наименование раздела (лекции, практического или лабораторного занятия) | Используемые интерактивные образовательные технологии | Объем, час. |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 1. | Лекция раздела «Функции» | Разбор конкретных ситуаций | 2 |
| 2. | Использование подпрограмм и рекурсивного метода на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| 3. | Работа со строками на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| 4. | Программирование с использованием файлов на языке C# (лабораторная работа). | Творческие задания, работа в малых группах | 2 |
| Итого: | | | 8 |

Практическая подготовка обучающихся при реализации дисциплины осуществляется путем проведения лабораторных занятий, предусматривающих участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью и направленных на формирование, закрепление, развитие практических навыков и компетенций по направленности (профилю) «Разработка программно-информационных систем» программы бакалавриата.

Практическая подготовка обучающихся при реализации дисциплины организуется в модельных условиях (оборудованных полностью) в лабораториях университета).

Практическая подготовка обучающихся проводится в соответствии с положением П 02.181

Содержание дисциплины обладает значительным воспитательным потенциалом, поскольку в нем аккумулирован исторический и современный научный опыт человечества. Реализация воспитательного потенциала дисциплины осуществляется в рамках единого образовательного и воспитательного процесса и

способствует непрерывному развитию личности каждого обучающегося. Дисциплина вносит значимый вклад в формирование профессиональной культуры обучающихся. Содержание дисциплины способствует профессионально-трудовому, культурно-творческому воспитанию обучающихся.

Реализация воспитательного потенциала дисциплины подразумевает:

- целенаправленный отбор преподавателем и включение в лекционный материал, материал для практических и лабораторных занятий содержания, демонстрирующего обучающимся образцы настоящего научного подвижничества создателей и представителей данной отрасли науки, высокого профессионализма ученых, их ответственности за результаты и последствия деятельности человека и общества; примеры подлинной нравственности людей, причастных к развитию науки и производства, а также примеры творческого мышления*;*
- применение технологий, форм и методов преподавания дисциплины, имеющих высокий воспитательный эффект за счет создания условий для взаимодействия обучающихся с преподавателем, другими обучающимися, (командная работа, проектное обучение, разбор конкретных ситуаций и др.)
- личный пример преподавателя, демонстрацию им в образовательной деятельности и общении с обучающимися за рамками образовательного процесса высокой общей и профессиональной культуры.

Реализация воспитательного потенциала дисциплины на учебных занятиях направлена на поддержание в университете единой развивающей образовательной и воспитательной среды. Реализация воспитательного потенциала дисциплины в ходе самостоятельной работы обучающихся способствует развитию в них целеустремленности, инициативности, креативности, ответственности за результаты своей работы – качеств, необходимых для успешной социализации и профессионального становления.

## 7 Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

### 7.1. Перечень компетенций с указанием этапов их формирования в процессе освоения основной профессиональной образовательной программы

Таблица 7.1 – Этапы формирования компетенций

| Код компетенции и наименование компетенции | Этапы формирования компетенций и дисциплины (модули) и практики, при изучении/ прохождении которых формируется данная компетенция | | |
|---|---|---|---|
| | начальный | основной | завершающий |
| 1 | 2 | 3 | 4 |
| ПК – 1 Способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности | Программирование на языках высокого уровня<br>Основы электроники<br>Языки объектно-ориентированного программирования | Производственная практика (научно-исследовательская работа) | Системный анализ<br>Производственная преддипломная практика<br>Выполнение и защита выпускной квалификационной работы |
| ПК – 4 Способен применять различные технологии разработки программного обеспечения | Программирование на языках высокого уровня<br>Языки объектно-ориентированного программирования<br>Конструирование программного обеспечения<br>Проектирование и архитектура программных систем | Системное программное обеспечение<br>Системы реального времени;<br>Офисные технологии<br>Функциональное и логическое программирование<br>Производственная практика (научно-исследовательская работа) | Параллельное программирование;<br>Распределенное программирование;<br>Методы и алгоритмы обработки изображений<br>Тестирование программного обеспечения<br>Web-программирование;<br>Выполнение и защита выпускной квалификационной работы |
| ПК – 11 Способен создавать программные интерфейсы | Программирование на языках высокого уровня | Языки объектно-ориентированного программирования | Web-программирование<br>Проектирование человеко-машинного интерфейса<br>Производственная преддипломная практика<br>Выполнение и защита выпускной квалификационной работы |

## 7.2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Таблица 7.2 – Показатели и критерии оценивания компетенций, шкала оценивания

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| ПК-1 / начальный | ПК-1.3 Осуществляет сбор, анализ, формулировку и документирование требований заинтересованных лиц<br><br>ПК-1.4 Осуществляет разработку и документирование технического задания на программную систему | **Знать:** в целом сформированные, но неполные знания основных методов и приемов формализации задач, языков формализации функциональных спецификаций, возможностей существующей программно-технической архитектуры, возможностей современных и перспективных средств разработки программных продуктов, технических средств<br>**Уметь:** в целом успешное, но не систематическое умение использовать основные методы и приемы формализации задач, языки формализации функциональных спецификаций, возможности существующей программно-технической архитектуры, использовать возможности современных и перспективных средств разработки программных продуктов, технических средств<br>**Владеть:** в целом успешное, но не полное владение анализом возможностей реализации требований к | **Знать:** сформированные, но содержащие отдельные пробелы знания основных методов и приемов формализации задач, языков формализации функциональных спецификаций, возможностей существующей программно-технической архитектуры, возможностей современных и перспективных средств разработки программных продуктов, технических средств<br>**Уметь:** успешное, но содержащее отдельные пробелы умение использовать основные методы и инструменты разработки программного обеспечения, возможности существующей программно-технической архитектуры, использовать возможности современных и перспективных средств разработки программных продуктов, технических средств<br>**Владеть:** успешное, но содержащее отдельные пробелы владение анализом возможностей реализации | **Знать:** сформированные систематические знания основных методов и приемов формализации задач, языков формализации функциональных спецификаций, возможностей существующей программно-технической архитектуры, возможностей современных и перспективных средств разработки программных продуктов, технических средств<br>**Уметь:** сформированное умение применять основные методы и инструменты разработки программного обеспечения, возможности существующей программно-технической архитектуры, использовать возможности современных и перспективных средств разработки программных продуктов, технических средств<br>**Владеть:** сформированное владение навыками использования анализа возможностей реализации |

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| | | программному обеспечению, оценкой времени и трудоемкости реализации требований к программному обеспечению, оценкой и согласованием сроков выполнения поставленных задач | требований к программному обеспечению, оценкой времени и трудоемкости реализации требований к программному обеспечению, оценкой и согласованием сроков выполнения поставленных задач | требований к программному обеспечению, оценки времени и трудоемкости реализации требований к программному обеспечению, оценки и согласования сроков выполнения поставленных задач |
| ПК-4 / начальный | ПК-4.2 Разрабатывает интерфейсы взаимодействия модулей системы<br><br>ПК-4.3 Разрабатывает интерфейсы взаимодействия с внешней средой<br><br>ПК-4.4 Выполняет тестирование программного обеспечения<br><br>ПК-4.5 Разрабатывает структуры данных | **Знать:** в целом сформированные, но неполные знания синтаксиса выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка программирования, методов и приемов алгоритмизации поставленных задач, отладки программного кода, выявления ошибок в программном коде, типов данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Уметь:** в целом успешное, но не систематическое умение использовать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования, | **Знать:** сформированные, но содержащие отдельные пробелы знания синтаксиса выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка программирования, методов и приемов алгоритмизации поставленных задач, отладки программного кода, выявления ошибок в программном коде, типов данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Уметь:** успешное, но содержащее отдельные пробелы умение использовать синтаксис выбранного языка программирования, особенности программирования на этом языке, | **Знать:** сформированные систематические знания синтаксиса выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка программирования, методов и приемов алгоритмизации поставленных задач, отладки программного кода, выявления ошибок в программном коде, типов данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Уметь:** сформированное умение использовать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования, методы и |

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| | | методы и приемы алгоритмизации поставленных задач, отладку программного кода, выявление ошибок в программном коде, типы данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Владеть:**<br>в целом успешное, но не полное владение созданием программного кода в соответствии с техническим заданием (готовыми спецификациями), разработкой алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов, проверкой и отладкой программного кода, проектированием структур данных | стандартные библиотеки языка программирования, методы и приемы алгоритмизации поставленных задач, отладку программного кода, выявление ошибок в программном коде, типы данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Владеть:**<br>успешное, но содержащее отдельные пробелы владение навыками создания программного кода в соответствии с техническим заданием (готовыми спецификациями), разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов, проверки и отладки программного кода, проектирования структур данных. | приемы алгоритмизации поставленных задач, отладку программного кода, выявление ошибок в программном коде, типы данных, определяемых пользователем: массивы, строки, структуры; файлы, классы<br>**Владеть:**<br>сформированное владение навыками создания программного кода в соответствии с техническим заданием (готовыми спецификациями), разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов, проверки и отладки программного кода, проектирования структур данных. |
| ПК-11/ начальный | ПК-11.2 Разрабатывает алгоритм работы интерфейса<br><br>ПК-11.3 | **Знать:**<br>в целом сформированные, но неполные знания методов и средств проектирования программных, правил, | **Знать:**<br>сформированные, но содержащие отдельные пробелы знания методов и средств проектирования | **Знать:**<br>сформированные систематические знания методов и средств проектирования программных, |

| Код компетенции / этап | Показатели оценивания компетенций *(индикаторы достижения компетенций, закрепленные за дисциплиной)* | Критерии и шкала оценивания компетенций | | |
|---|---|---|---|---|
| | | Пороговый уровень («удовлетворительно») | Продвинутый уровень («хорошо») | Высокий уровень («отлично») |
| 1 | 2 | 3 | 4 | 5 |
| | Разрабатывает и отлаживает программные модули для реализации функций интерфейса | алгоритмов и технологий создания тестовых наборов данных<br>**Уметь:**<br>в целом успешное, но не систематическое умение использовать методы и средства проектирования программных, правила, алгоритмы и технологии создания тестовых наборов данных<br>**Владеть:**<br>в целом успешное, но не полное владение навыками проектирования программных интерфейсов, разработки программ для решения задач на языках | программных, правил, алгоритмов и технологий создания тестовых наборов данных<br>**Уметь:**<br>успешное, но содержащее отдельные пробелы умение использовать методы и средства проектирования программных, правила, алгоритмы и технологии создания тестовых наборов данных<br>**Владеть:**<br>успешное, но содержащее отдельные пробелы владения навыками проектирования программных интерфейсов, разработки программ для решения задач на языках | правил, алгоритмов и технологий создания тестовых наборов данных<br>**Уметь:**<br>сформированное умение использовать методы и средства проектирования программных, правила, алгоритмы и технологии создания тестовых наборов данных<br>**Владеть:**<br>сформированное владение навыками проектирования программных интерфейсов, разработки программ для решения задач на языках |

## 7.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения основной профессиональной образовательной программы

Таблица 7.3 - Паспорт комплекта оценочных средств для текущего контроля успеваемости

| № п/п | Раздел (тема) дисциплины | Код контролируемой компетенции (или ее части) | Технология формирования | Оценочные средства | | Описание шкал оценивания |
|---|---|---|---|---|---|---|
| | | | | наименование | №№ заданий | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 семестр | | | | | | |

| № п/п | Раздел (тема) дисциплины | Код контролируемой компетенции (или ее части) | Технология формирования | Оценочные средства | | Описание шкал оценивания |
|---|---|---|---|---|---|---|
| | | | | наименование | №№ заданий | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | Основные понятия и определения. | ПК-1, ПК-4, ПК-11 | Лекции СРС | контрольные вопросы к защите лабораторной работы №1 | 1–10 | Согласно табл.7.2 |
| 2 | Программирование линейных алгоритмов на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №2 | 11–20 | Согласно табл.7.2 |
| 3 | Программирование разветвленных алгоритмов на языке C#.. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №3 | 21–30 | Согласно табл.7.2 |
| 4 | Программирование циклических алгоритмов на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №4 | 21–30 | Согласно табл.7.2 |
| 5 | Работа с одномерными массивами на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №5 | 31–40 | Согласно табл.7.2 |
| 6 | Работа с двумерными массивами на языке C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №6 | 31–40 | Согласно табл.7.2 |
| 2 семестр | | | | | | |
| 7 | Функции. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №7 | 41–50 | Согласно табл.7.2 |

| № п/п | Раздел (тема) дисциплины | Код контролируемой компетенции (или ее части) | Технология формирования | Оценочные средства | | Описание шкал оценивания |
|---|---|---|---|---|---|---|
| | | | | наименование | №№ заданий | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | Строки. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №8 | 51–60 | Согласно табл.7.2 |
| 9 | Графический режим работы | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №9 | 61–70 | Согласно табл.7.2 |
| 11 | Структуры и объединения. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №10 | № 81–90 | Согласно табл.7.2 |
| 3 семестр | | | | | | |
| 12 | Файлы в C#. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб. СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №11 | 91–100 101–110 | Согласно табл.7.2 |
| 13 | Обработка исключительных ситуаций. | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб СРС | Устный опрос, собеседование | 111–120 | Согласно табл.7.2 |
| 14 | Объектно-ориентированное программирование | ПК-1, ПК-4, ПК-11 | Лекции Лаб.раб СРС | Устный опрос, собеседование, контрольные вопросы к защите лабораторной работы №12 | 121–130 | Согласно табл.7.2 |
| 16 | Перспективы использования языков программирования | ПК-1, ПК-4, ПК-11 | Лекции СРС | Устный опрос, собеседование | 141–150 | Согласно табл.7.2 |

Примеры типовых контрольных заданий проведения текущего контроля успеваемости

Вопросы собеседования по разделу (теме) 1. «Основные понятия и определения»

1 Какие этапы работы по созданию приложений можно выполнять с помощью IDE?

2 Для чего предназначен инспектор объектов?

3 Каким образом используется палитра компонентов?

4 Для чего используется панель инструментов?

5 Каким образом переключаются окна проектировщика форм и редактора кода?

6 Каким образом вставляется обработчик событий в код программы?

7 Как создать новый проект?

8 Как сохранить проект?

9 Каким образом открыть существующий проект?

10 Какие основные части содержит простая программа на C#?

11 Каким образом запустить проект только на компиляцию?

12 Каким образом запустить проект на компиляцию и выполнение?

13 Как используются контрольные точки для отладки программ?

14 Как использовать пошаговый режим при отладке программ?

15 Каким образом можно использовать компонент TextBox?

16 Каким образом можно использовать компонент Label?

17 Каким образом можно использовать компонент Button?

Полностью оценочные материалы и оценочные средства для проведения текущего контроля успеваемости представлены в УММ по дисциплине.

Практическая подготовка обучающихся при реализации данной дисциплины организуется, в частности, путем выполнения и защиты курсовой работы (проекта) на одну из предложенных тем

Темы курсовых работ

1. **Программа поиска кратчайшего пути между двумя населенными пунктами в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами. Для двух любых населенных пунктов найти кратчайший путь между ними. Отобразить решение графически, выделив цветом найденный путь.
2. **Программа поиска замкнутого пути между двумя населенными пунктами в системе двусторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. Найти замкнутый путь (с возвратом в тот же населенный пункт) длиной не более 100 км, проходящий через каждую дорогу ровно 1 раз. Отобразить решение графически, выделив цветом найденный путь.
3. **Программа поиска кратчайшего пути между двумя населенными пунктами в системе односторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) односторонними дорогами. Для двух любых населенных пунктов найти кратчайший путь между ними. Отобразить решение графически, выделив цветом найденный путь.
4. **Программа поиска замкнутого пути между двумя населенными пунктами в системе односторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой

(произвольным образом) односторонними дорогами некоторой длины. Найти замкнутый путь (с возвратом в тот же населенный пункт) длиной не более 100 км, проходящий через каждую дорогу ровно 1 раз. Отобразить решение графически, выделив цветом найденный путь.

5. **Программа определения достижимости населенного пункта в системе односторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) односторонними дорогами некоторой длины. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Отобразить решение графически, выделив цветом найденный результат.
6. **Программа определения достижимости населенного пункта в системе двусторонних дорог**. Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Отобразить решение графически, выделив цветом найденный результат.
7. **Программа определения N-периферии в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. N-периферией называется множество городов, расстояние от которых до выделенного города (столицы) больше N. Определить N-периферию для заданного N. Отобразить решение графически, выделив цветом найденный результат.
8. **Программа определения достижимости города в системе односторонних дорог**. Описание: даны несколько городов, соединенных между собой (произвольным образом) односторонними дорогами некоторой длины. Определить, можно ли проехать из города A в город B таким образом, чтобы посетить город C и не проезжать никакой дороги более одного раза. Отобразить решение графически, выделив цветом найденный результат.
9. **Программа поиска минимальной суммы расстояний между населенными пунктами в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины, причем для любой пары населенных пунктов можно указать соединяющий их путь. Найти такой населенный пункт, для которого сумма расстояний до остальных городов минимальна. Отобразить решение графически, выделив цветом найденный результат.
10. **Программа поиска пути с минимальными затратами между двумя городами в системе двусторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. За проезд каждой дороги взимается некоторая пошлина. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма пошлин проезжаемых дорог. Отобразить решение графически, выделив цветом найденный результат.
11. **Программа определения достижимости города в изменяемой системе двусторонних дорог.** Описание: даны несколько городов, соединенных между собой (произвольным образом) двусторонними дорогами. Определить, можно ли , построив какие-нибудь 3 новые дороги, из заданного города A добрать до всех других городов. Отобразить решение графически, выделив цветом найденный результат.
12. **Программа определения недостижимости города в изменяемой системе двусторонних дорог**. Описание: даны несколько городов, соединенных между собой (произвольным образом) двусторонними дорогами. Определить, можно ли , закрыв какие-нибудь 3 дороги, добиться того, чтобы нельзя было попасть из города A в город B. Отобразить решение графически, выделив цветом найденный результат.
13. **Программа поиска пути с минимальными затратами между двумя городами в системе односторонних дорог.** Описание: даны несколько населенных пунктов, соединенных между собой (произвольным образом) односторонними дорогами некоторой длины. За проезд каждой дороги взимается некоторая пошлина. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма пошлин проезжаемых дорог. Отобразить решение графически, выделив цветом найденный результат.

14. **Программа поиска пути с минимальными затратами между двумя городами в системе двусторонних дорог**. Описание: даны несколько городов, соединенных между собой (произвольным образом) двусторонними дорогами некоторой длины. За проезд каждой дороги взимается некоторая пошлина. Найти путь из города A в город B такой, чтобы величина P была минимальна, где P - сумма пошлин проезжаемых дорог. Если таких путей несколько, то выбрать более короткий путь. Отобразить решение графически, выделив цветом найденный результат.
15. **Программа поиска кратчайшего пути между двумя городами в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Для двух любых населенных пунктов найти кратчайший путь между ними. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
16. **Программа поиска кратчайшего пути между двумя городами в двух системах двусторонних дорог.** Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Для двух любых населенных пунктов найти кратчайший путь между ними. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
17. **Программа поиска пути с минимальными затратами между двумя городами в двух системах односторонних дорог.** Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма денег за проезд дорог. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
18. **Программа поиска пути с минимальными затратами между двумя городами в двух системах двусторонних дорог.** Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B с минимальной величиной S+P, где S - сумма длин дорог, а P - сумма денег за проезд дорог. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
19. **Программа поиска пути с минимальными затратами между двумя городами в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B такой, чтобы величина P была минимальна, где P - сумма денег за проезд дорог. Если таких путей несколько, то выбрать более короткий путь. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
20. **Программа поиска пути с минимальными затратами между двумя городами в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. За проезд каждой дороги путешественник платит некоторую сумму денег. Найти путь из города A в город B такой, чтобы величина P была минимальна, где P - сумма денег за проезд дорог. Если таких путей несколько, то выбрать более короткий путь. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
21. **Программа определения N-периферии в двух системах двусторонних дорог.** Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. N-периферией называется множество городов, расстояние от которых до выделенного города (столицы) больше N. Определить N-периферию для заданного N. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
22. **Программа определения N-периферии в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством

городов. N-периферией называется множество городов, расстояние от которых до выделенного города (столицы) больше N. Определить N-периферию для заданного N. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.

23. **Программа поиска минимальной суммы расстояний между населенными пунктами в двух системах односторонних дорог**. Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов, причем для любой пары населенных пунктов можно указать соединяющий их путь. Найти такой населенный пункт, для которого сумма длин дорог до остальных городов минимальна. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
24. **Программа поиска минимальной суммы расстояний между населенными пунктами в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов, причем для любой пары населенных пунктов можно указать соединяющий их путь. Найти такой населенный пункт, для которого сумма длин дорог до остальных городов минимальна. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат**.**
25. **Программа поиска замкнутого пути между двумя городами в двух системах односторонних дорог.** Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Найти замкнутый путь (с возвратом в тот же населенный пункт), проходящий через каждый город ровно 1 раз. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
26. **Программа поиска замкнутого пути между двумя городами в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Найти замкнутый путь (с возвратом в тот же населенный пункт), проходящий через каждый город ровно 1 раз. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный путь.
27. **Программа определения достижимости населенного пункта в двух системах двусторонних дорог**. Описание: заданы две системы двусторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
28. **Программа определения достижимости населенного пункта в двух системах односторонних дорог.** Описание: заданы две системы односторонних дорог (железные и шоссейные) с одним и тем же множеством городов. Определить, есть ли населенный пункт, из которого можно добраться до каждого из остальных пунктов, проезжая не более 100 км. Путь может проходить как по железным, так и по шоссейным дорогам. Отобразить решение графически, выделив цветом найденный результат.
29. **Программа поиска кратчайшего пути шахматной фигуры по клеточному полю.** Найти кратчайший путь передвижения шахматной фигуры (коня, ладьи, слона, ферзя, короля) по клеточному полю, соединяющий две указанные его клетки. Отобразить решение графически, выделив цветом найденный результат.
30. **Программа определения допустимости хода шахматной фигуры. Описание: задана корректная шахматная позиция**. Определить, является ли допустимым ход шахматной фигуры. Если после хода король атакован, то ход считается не допустимым. Отобразить решение графически, выделив цветом найденный результат.
31. **Программа поиска кратчайшего пути в лабиринте.** Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более трех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. Найти кратчайший путь от входа в лабиринт к его выходу. Отобразить решение графически, выделив цветом найденный результат.

32. **Программа поиска пути в лабиринте. Описание: задан лабиринт, составленный из комнат**. В каждой из комнат имеется не менее одной и не более трех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. Некоторые комнаты объявлены опасными. Найти путь от входа в лабиринт к его выходу, не проходящий через опасные комнаты. Отобразить решение графически, выделив цветом найденный результат.
33. **Программа поиска пути в лабиринте**. Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более четырех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. Одна из комнат объявлена комнатой, приносящей удачу. Определить, есть ли путь от входа в лабиринт к его выходу, не проходящий одну и ту же комнату два раза и проходящий через комнату, приносящую удачу. Отобразить решение графически, выделив цветом найденный результат.
34. **Программа поиска минимального по стоимости пути в лабиринте.** Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более четырех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. За посещение каждой комнаты взимается пошлина в размере от одного до ста рублей. Найти путь от входа в лабиринт к его выходу, чтобы величина P была минимальна, где P - сумма пошлин за посещение комнат лабиринта. Если путей с одинаковыми значениями P несколько, то выбрать более короткий путь. Отобразить решение графически, выделив цветом найденный результат.
35. **Программа поиска пути в лабиринте.** Описание: задан лабиринт, составленный из комнат. В каждой из комнат имеется не менее одной и не более четырех дверей, соединяющих между собой соседние комнаты. Одна из дверей называется входом в лабиринт, другая – выходом из лабиринта. В каждой комнате хранятся сокровища: от одной до ста золотых монет. Найти путь от входа в лабиринт к его выходу, не проходящий одну и ту же комнату два раза и позволяющий унести как можно больше сокровищ. Отобразить решение графически, выделив цветом найденный результат.

Требования к структуре, содержанию, объему, оформлению курсовых работ, процедуре защиты, а также критерии оценки определены в:

− стандарте СТУ 04.02.030-2017 «Курсовые работы (проекты). Выпускные квалификационные работы. Общие требования к структуре и оформлению»;

− положении П 02.016-2018 «О балльно-рейтинговой системе оценивания результатов обучения по дисциплинам (модулям) и практикам при освоении обучающимися образовательных программ»;

− методических указаниях по выполнению курсовой работы.

Типовые задания для проведения промежуточной аттестации обучающихся

Промежуточная аттестация по дисциплине проводится в форме экзамена. Экзамен проводится в виде компьютерного тестирования.

Примеры типовых заданий для проведения промежуточной аттестации обучающихся

Для тестирования используются контрольно-измерительные материалы (КИМ) – вопросы и задания в тестовой форме, составляющие банк тестовых заданий (БТЗ) по дисциплине, утвержденный в установленном в университете порядке.

Проверяемыми на промежуточной аттестации элементами содержания являются темы дисциплины, указанные в разделе 4 настоящей программы. Все темы дисциплины

отражены в КИМ в равных долях (%). БТЗ включает в себя не менее 100 заданий и постоянно пополняется. БТЗ хранится на бумажном носителе в составе УММ и электронном виде в ЭИОС университета.

Для проверки знаний используются вопросы и задания в различных формах:

- закрытой (с выбором одного или нескольких правильных ответов),
- открытой (необходимо вписать правильный ответ).

Результаты практической подготовки (умения, навыки (или опыт деятельности) и компетенции) проверяются с помощью компетентностно-ориентированных задач (ситуационных, производственных) и различного вида конструкторов.

Все задачи являются многоходовыми. Некоторые задачи, проверяющие уровень сформированности компетенций, являются многовариантными. Часть умений, навыков и компетенций прямо не отражена в формулировках задач, но они могут быть проявлены обучающимися при их решении.

В каждый вариант КИМ включаются задания по каждому проверяемому элементу содержания во всех перечисленных выше формах и разного уровня сложности. Такой формат КИМ позволяет объективно определить качество освоения обучающимися основных элементов содержания дисциплины и уровень сформированности компетенций.

Примеры типовых заданий для проведения промежуточной аттестации обучающихся

Задание в закрытой форме:

Оператор множественного выбора switch/case можно заменить набором операторов ( 2 балла) :

Выберите один ответ:

- for
- if else if
- while
- goto

.....

Задание в открытой форме:

В результате работы фрагмента программы ( 2 балла)

```
int a = 1, b = 1;
a + = + + b;
```

Выберите один ответ:

- a=4
- a=1
- a=2

○ a=3

.....

Компетентностно-ориентированная задача:

Дан двумерный массив целых чисел. Для каждой строки, упорядоченной по возрастанию или по убыванию, найти максимальный элемент.

.....

### 7.4 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций, регулируются следующими нормативными актами университета:

− Положение П 02.016 – 2018 О балльно-рейтинговой системе оценивания результатов обучения по дисциплинам (модулям) и практикам при освоении обучающимися образовательных программ;

− методические указания, используемые в образовательном процессе, указанные в списке литературы.

Для текущего контроля по дисциплине в рамках действующей в университете балльно-рейтинговой системы применяется следующий порядок начисления баллов:

Таблица 7.4 – Порядок начисления баллов в рамках БРС

| Форма контроля | Минимальный балл | | Максимальный балл | |
|---|---|---|---|---|
| | балл | примечание | балл | примечание |
| Лекция №1, 2 | | Материал усвоен менее чем на 50% | 1 | Материал усвоен более чем на 50% |
| Лабораторная работа №1, 2 | | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| Лекция № 3, 4 | | Материал усвоен менее чем на 50% | 1 | Материал усвоен более чем на 50% |
| Лабораторные работы №3 - 4 | | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| Лекция № 5, 6 | | Материал усвоен менее чем на 50% | 1 | Материал усвоен более чем на 50% |
| Лабораторные работа № 5 - 6 | | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| Лекция № 7, 8 | | Материал усвоен менее чем на 50% | 1 | Материал усвоен более чем на 50% |
| Лабораторные работы № 7 | | Выполнил, но «не защитил» | 8 | Выполнил и «защитил» |
| СРС | | | 12 | |
| Итого | | | 36 | |
| Посещаемость | | | 14 | |

| Форма контроля | Минимальный балл | | Максимальный балл | |
|---|---|---|---|---|
| | балл | примечание | балл | примечание |
| Экзамен | | | 60 | |
| Итого | | | 100 | |

Для промежуточной аттестации, проводимой в форме тестирования, используется следующая методика оценивания знаний, умений, навыков и (или) опыта деятельности. В каждом варианте КИМ - количество заданий соответствует количеству тем в содержании дисциплины. Каждое задание оценивается в количестве баллов, зависящем от сложности темы.

Максимальное количество баллов за тестирование - 60 баллов.

## 8 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

### 8.1 Основная учебная литература

1. Белов В. Г. Основы программирования на языке C++ Builder [Текст]: учеб. пособие / В. Г. Белов, Т. М. Белова; Юго-Зап. гос. ун-т. – Курск, 2015. – 160 с.
2. Белов В. Г. Основы программирования на языке C++ Builder [Электронный ресурс]: учеб. пособие / В. Г. Белов, Т. М. Белова; Юго-Зап. гос. ун-т. – Курск, 2015. – 160 с.
3. Иванова Г. С. Технология программирования [Текст] : учебник / Г. С. Иванова. - М. : Кнорус, 2011. - 336 с.

### 8.2 Дополнительная учебная литература

1. Глушаков, С. В. Язык программирования C++ [Текст] : руководство / С. В. Глушаков, А. В. Коваль, С. В. Смирнов. - М. : АСТ, 2004. - 500 с.
2. Фридман, А. Л. Язык программирования Си. Курс лекций [Текст] : учебное пособие / А. Л. Фридман. - 2-е изд., испр. - М. : ИНТУИТ. РУ Интернет-университет Информационных Технологий, 2004. - 264 с.
3. Дубровин, В.В. Программирование на C# [Электронный ресурс]: в 2 ч. / В.В. Дубровин ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего образования «Тамбовский государственный технический университет». – Тамбов : Издательство ФГБОУ ВПО «ТГТУ», 2017. – Ч. 1. – 81 с. : ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=499439

### 8.3 Перечень методических указаний

1. Изучение интегрированной среды MS Visual Studio C# [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплинам «Программирование на языках высокого уровня», «Языки программирования» для студентов направлений подготовки 09.03.04 «Программная инженерия», 10.03.01 «Информационная безопасность», 10.05.02 «Информационная безопасность телекоммуникационных систем»/ Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. Курск: ЮЗГУ, 2019. – 32 с.

2. Программирование линейных алгоритмов на языке C# [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплинам

«Программирование на языках высокого уровня», «Языки программирования» для студентов направлений подготовки 09.03.04 «Программная инженерия», 10.03.01 «Информационная безопасность», 10.05.02 «Информационная безопасность телекоммуникационных систем»/ Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2019. – 26 с.

3. Программирование разветвленных алгоритмов на языке C# [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплинам «Программирование на языках высокого уровня», «Языки программирования» для студентов направлений подготовки 09.03.04 «Программная инженерия», 10.03.01 «Информационная безопасность», 10.05.02 «Информационная безопасность телекоммуникационных систем»/ Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. Курск: ЮЗГУ, 2019. – 21 с.

4. Программирование циклических алгоритмов на языке C++ [Электронный ресурс] : методические указания по выполнению лабораторной работы для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 18 с.
5. Программирование на языках высокого уровня [Электронный ресурс] : методические рекомендации по выполнению курсовой работы для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 20 с.
6. Программирование с использованием одномерных массивов на языке C++ [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплине «Программирование на языках высокого уровня» для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 17 с.
7. Программирование с использованием двумерных массивов на языке C++ [Электронный ресурс] : методические указания по выполнению лабораторной работы по дисциплине «Программирование на языках высокого уровня» для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 16 с.
8. Программирование с использованием функций на языке C++ [Электронный ресурс]: методические указания по выполнению лабораторной работы по дисциплине «Программирование на языках высокого уровня» для студентов направления подготовки 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост. Т. М. Белова, В. Г. Белов. − Курск: ЮЗГУ, 2017. – 24 с.
9. Методические рекомендации по организации самостоятельной работы студентов по дисциплинам учебных планов направлений подготовки 09.03.04 и 09.04.04 [Электронный ресурс] / Юго-Зап. гос. ун-т; сост. Р. А. Томакова. − Курск : ЮЗГУ, 2017. − 55 с.

### 8.4 Другие учебно-методические материалы

Отраслевые научно-технические журналы в библиотеке университета:

- Вестник компьютерных и информационных технологий
- Информационные технологии
- Информационные технологии и вычислительные системы

- Программирование
- Программные продукты и системы
- Искусственный интеллект и принятие решений
- Информатика и её применения

## 9 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- Образовательный сайт Life-prog: http://www.life-prog.ru.
- Электронно-библиотечная система «Университетская библиотека online»: http://www.biblioclub.ru.
- Электронная библиотека ЮЗГУ: http://www.lib.swsu.ru.
- Электронная библиотека: http://www.window.edu.ru
- Интернет-университет информационных технологий http: //www.intuit.ru/
- Виртуальный музей истории вычислительной техники в картинках http://www.computerhistory.narod.ru
- Энциклопедия отечественной информатики
- http://www.computer-museum.ru/
- http://www.i-exam.ru/ – Официальный сайт Федерального интернет-экзамена в сфере профессионального образования НИИ Мониторинга качества образования [Электронный ресурс].

## 10 Методические указания для обучающихся по освоению дисциплины

Основными видами аудиторной работы студента при изучении дисциплины являются лекции, лабораторные и практические занятия. Студент не имеет права пропускать занятия без уважительных причин.

На лекциях излагаются и разъясняются основные понятия темы, связанные с ней теоретические и практические проблемы, даются рекомендации для самостоятельной работы. В ходе лекции студент должен внимательно слушать и конспектировать материал.

Изучение наиболее важных тем или разделов дисциплины завершают лабораторные занятия, которые обеспечивают: контроль подготовленности студента; закрепление учебного материала; приобретение опыта устных публичных выступлений, ведения дискуссии, в том числе аргументации и защиты выдвигаемых положений и тезисов.

Лабораторному занятию предшествует самостоятельная работа студента, связанная с освоением материала, полученного на лекциях, и материалов, изложенных в учебниках и учебных пособиях, а также литературе, рекомендованной преподавателем.

Самостоятельную работу следует начинать с первых занятий. От занятия к занятию нужно регулярно прочитывать конспект лекций, знакомиться с соответствующими разделами учебника, читать и конспектировать литературу по каждой теме дисциплины. Самостоятельная работа дает студентам возможность равномерно распределить нагрузку, способствует более глубокому и качественному усвоению учебного материала. В случае необходимости студенты обращаются за консультацией к преподавателю по вопросам дисциплины «Программирование на языках высокого уровня» с целью усвоения и закрепления компетенций.

По согласованию с преподавателем или по его заданию студенты могут готовить рефераты по отдельным темам дисциплины, выступать на занятиях с докладами. Основу докладов составляет, как правило, содержание подготовленных студентами рефератов.

Качество учебной работы студентов преподаватель оценивает по результатам тестирования, собеседования, защиты отчетов по лабораторным работам, а также по результатам докладов.

## 11 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Windows: MSDN subscriptions, Договор IT000012385 MS Visual Studio Community Edition 2017: Бесплатная, Freeware лицензия. Свободное программное обеспечение: Mozilla Firefox: GNU GPL LibreOffice: GNU LGPL.

## 12 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Для осуществления практической подготовки обучающихся при реализации дисциплины используются оборудование и технические средства обучения:

- учебная аудитория для проведения занятий лекционного типа и лаборатории кафедры программной инженерии, оснащенные учебной мебелью: столы, стулья для обучающихся; стол, стул для преподавателя;
- электронная доска. Мультимедиа центр: ноутбук ASUS X50VL PMD-T2330/14"/1024Mb/160Gb/сумка/проектор inFocus IN24+;
- рабочие станции (ПЭВМ) PremiumP43/ E6300/ 4Гб DDR2/ 320 Гб / DVD RW/ AcerV223HQb с прогр. обеспеч. (27002.40).

## 13 Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

При обучении лиц с ограниченными возможностями здоровья учитываются их индивидуальные психофизические особенности. Обучение инвалидов осуществляется также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).

Для лиц с нарушением слуха возможно предоставление учебной информации в визуальной форме (краткий конспект лекций; тексты заданий, напечатанные увеличенным шрифтом), на аудиторных занятиях допускается присутствие ассистента, а также сурдопереводчиков и тифлосурдопереводчиков. Текущий контроль успеваемости осуществляется в письменной форме: обучающийся письменно отвечает на вопросы, письменно выполняет практические задания. Доклад (реферат) также может быть представлен в письменной форме, при этом требования к содержанию остаются теми же, а требования к качеству изложения материала

(понятность, качество речи, взаимодействие с аудиторией и т. д.) заменяются на соответствующие требования, предъявляемые к письменным работам (качество оформления текста и списка литературы, грамотность, наличие иллюстрационных материалов и т.д.). Промежуточная аттестация для лиц с нарушениями слуха проводится в письменной форме, при этом используются общие критерии оценивания. При необходимости время подготовки к ответу может быть увеличено.

Для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, на аудиторных занятиях, а также при проведении процедур текущего контроля успеваемости и промежуточной аттестации могут быть предоставлены необходимые технические средства (персональный компьютер, ноутбук или другой гаджет); допускается присутствие ассистента (ассистентов), оказывающего обучающимся необходимую техническую помощь (занять рабочее место, передвигаться по аудитории, прочитать задание, оформить ответ, общаться с преподавателем).

**14 Лист дополнений и изменений, внесенных в рабочую программу дисциплины**

| Номер изменения | Номера страниц | | | | Всего стра-ниц | Дата | Основание для изменения и подпись лица, проводившего изменения |
|---|---|---|---|---|---|---|---|
| | изменённых | заменённых | аннулированных | новых | | | |
| 1 | 3, 10, 14, 15 | | | | 4 | | |