

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Локтионова Оксана Геннадьевна  
Должность: проректор по учебной работе  
Дата подписания: 02.09.2021 10:08:59  
Уникальный программный ключ:  
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

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

Кафедра программной инженерии

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Локтионова  
« 18 » 09 2019 г.



**ПРОГРАММИРОВАНИЕ РАЗВЕТВЛЕННЫХ АЛГОРИТМОВ  
НА ЯЗЫКЕ C#**

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»

Курск 2019

УДК 681.3.06(071.8)

Составители: Т.М. Белова, В.Г. Белов

Рецензент

Кандидат технических наук, доцент кафедры программной инженерии ЮЗГУ И.Н. Ефремова

**Программирование разветвленных алгоритмов на языке С#:** 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, 2019. – 23 с.

Содержат основные теоретические положения и приемы разработки разветвленных программ на языке С#, пример решения типовой задачи, индивидуальные задания и контрольные вопросы к защите лабораторной работы.

Методические указания соответствуют требованиям рабочих программ по дисциплинам «Программирование на языках высокого уровня», «Языки программирования».

Предназначены для студентов дневной и заочной форм обучения направлений подготовки 09.03.04 «Программная инженерия», 10.03.01 «Информационная безопасность», 10.05.02 «Информационная безопасность телекоммуникационных систем».

Текст печатается в авторской редакции.

Подписано в печать . Формат 60x84 1/16.

Усл. печ. л. . Уч.-изд. л. . Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет  
305040, Курск, ул.50 лет Октября, 94.

## ПРОГРАММИРОВАНИЕ РАЗВЕТВЛЕННЫХ АЛГОРИТМОВ НА ЯЗЫКЕ C#

**Цель работы** – изучение и приобретение навыков программирования разветвленных алгоритмов, освоение логических операций, условных операторов, оператора выбора.

### Основные понятия

*Составной оператор* представляет собой группу операторов, отделенных друг от друга точкой с запятой, и ограниченную операторными скобками { }.

Пример:

```
{  
оператор 1;  
оператор 2;  
...  
оператор n;  
}
```

Пример:

```
{  
a+=12;  
c=a*b+2;  
}
```

Составной оператор воспринимается как единое целое и может находиться в любом месте программы.

На языке *C#* имеются две формы условного оператора – полный и неполный.

### Полный условный оператор

Рассмотрим схему алгоритма полного условного оператора (рисунок 1). Ключевые слова *if*, *else* означают, соответственно, «если», «иначе». Условие – выражение логического типа. При записи условия могут использоваться операции отношения =, !=, >, >=, <, <= и логические операции &&, ||, !.

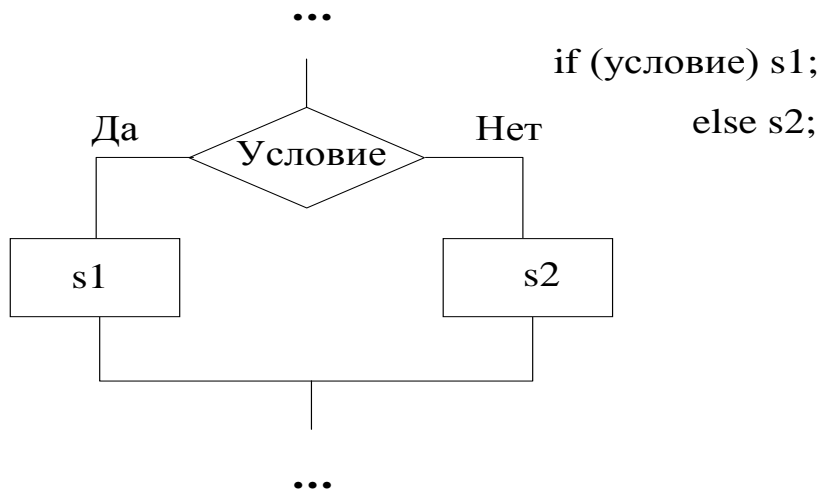


Рисунок 3.1 – Схема алгоритма полного условного оператора:  
s1, s2 – один простой или составной оператор

Если после условия или ключевого слова *else* следует выполнить не один, а несколько операторов, то необходимо использовать составной оператор. Результатом вычисления условия может являться истина (*true*) или ложь (*false*). Если условие истинно, то выполняется оператор *s1*, иначе – оператор *s2*.

Приоритеты выполнения операций (таблица 1):

Таблица 1 – Приоритеты операций

Приоритет	Категория	Операции	Описание
0	Первичные	. ( ) [ ] ++ -- new typeof sizeof checked unchecked ->	выбор члена (класса или объекта) вызов метода или делегата доступ по индексу (индексирование) постфиксный инкремент постфиксный декремент создание объекта ( создание экземпляра) идентификация типа определение размера операнда (только в опасном коде) контроль за переполнениями в выражениях отмена контроля за переполнениями в выражениях доступ к члену ( объекта ) по указателю (только в опасном коде)

Приоритет	Категория	Операции	Описание
1	Унарные	+ - ++ -- ~ ! (тип) & *	унарный плюс (здание знака) унарный минус (здание знака) префиксный инкремент префиксный инкремент поразрядное отрицание логическое отрицание приведение к заданному типу получение адреса (только в опасном коде) разыменование указателя (только в опасном коде)
2	Мультипликативные (Умножение)	* / %	умножение деление получение остатка при целочисленном делении
3	Аддитивные (Сложение)	+ -	сложение вычитание
4	Сдвиг	>> <<	поразрядный сдвиг вправо поразрядный сдвиг влево
5	Отношения, проверка типов	< > <= >= is as	меньше больше меньше или равно больше или равно сравнение типов ( возвращает логическое значение) проверка типов (возвращает тип или null)
6	Эквивалентность	== != ??	сравнение на равенство сравнение на неравенство сравнение с null (поглощение null)
7	Поразрядные операции	&   ^	& побитовое (поразрядное) И побитовое (поразрядное ) ИЛИ побитовое (поразрядное) исключающее ИЛИ
8	Логические бинарные операции	&   ^ && 	конъюнкция (логическое И) дизъюнкция (логическое ИЛИ) исключающая дизъюнкция условная конъюнкция условная дизъюнкция

Приоритет	Категория	Операции	Описание
9	Тернарная операция	?:	условное выражение
10	Присваивание	= += -= *= /= %= &= ^=  = >>= <<=	присваивание сложение с присваиванием вычитание с присваиванием умножение с присваиванием деление с присваиванием получение остатка от деления с присваиванием поразрядное И с присваиванием поразрядное исключающее ИЛИ с присваиванием поразрядное ИЛИ с присваиванием поразрядный сдвиг вправо с присваиванием поразрядный сдвиг влево с присваиванием

Пример:

```
int a=1, b=2, c=0;
```

```
if (a>b)
```

```
    c=a;
```

```
else c=b;
```

Так как выражение  $a > b$  ложно, будет выполняться оператор после *else*, т. е.  $c = b$ .

### Неполный условный оператор

Рассмотрим схему алгоритма неполного условного оператора (рисунок 2).

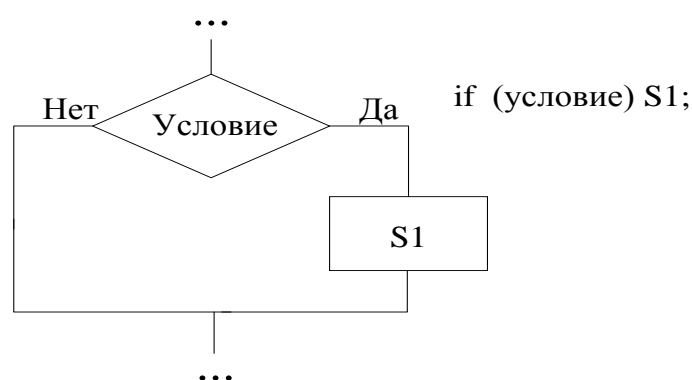


Рисунок 2 – Схема алгоритма неполного условного оператора

Если результат условия – истина, то выполняется оператор  $s1$ . Если ложь, то выполняется оператор, непосредственно следующий за условным оператором  $if$ .

Пример:

```

int a=2, b=8, c=0;
if (a>b) c=a+b;
c=12;
  
```

Результатом выполнения данного примера будет  $c=12$ , так как выражение  $a>b$  ложно и оператор  $c=a+b$  не будет выполняться. Если после условия следует выполнить не один, а несколько операторов, то необходимо использовать составной оператор.

Можно воспользоваться сокращенным условным оператором, который часто называется тернарный условный оператор:

*Выражение1*? *Выражение2*: *Выражение3*,  
 где *Выражение1* должно относиться к типу [bool](#), а *Выражение2* и *Выражение3* – к одному и тому же типу.

Сначала вычисляется логическое выражение и возвращается результат вычисления одного из двух выражений, в зависимости от того, чему равно значение логического выражения: true или false. Тогда пример выше будет выглядеть так.

```

int a=2, b=8, c=0;
c = (a>b) ? a+b : 12;
  
```

## Оператор выбора

Оператор *switch* часто используется вместо конструкции *if-else*, если одно выражение проверяется на соответствие трем или больше условиям.

```
switch <выражение - селектор> {
    case значение 1 : оператор 1; break;
    case значение 2 : оператор 2; break;
    ...
    case значение n : оператор n; break;
    default : оператор; break;
}
```

Оператор *switch* работает следующим образом. Сначала вычисляется значение выражения-селектора, затем обеспечивается реализация того оператора, константа выбора которого равна текущему значению селектора. Если значение выражения-селектора совпало со значением, указанным в одной из меток *case*, то выполняется оператор, записанный после этой метки, после чего будут выполняться все последующие операторы остальных меток, если не записан оператор *break*. Он прерывает выполнение структуры *switch*, и управление передается следующему за ней оператору. Если ни одна из констант не равна текущему значению селектора, выполняется оператор, стоящий за словом *default*. Слово *default* может отсутствовать. Если слово *default* отсутствует, то активизируется первый оператор за границей *switch*.

Селектор может иметь тип:

- *char*;
- *string*;
- *bool*;
- целочисленное значение, например *int* или *long*;
- значение перечисления *enum*.

Начиная с C# 7.0 выражение соответствия может быть любым выражением, отличным от *NULL*. Тип констант в любом случае должен совпадать с типом селектора.

### Примеры записи оператора *switch*:

1. Селектор целого типа *int k*:

```
switch (k)
```



```

{
    case 1: z=k+10; break;
    case 2: z=k+100; break;
    case 3: case 4: case 5: z=k+1000; break;
}

```

2. Селектор символьного типа *char n* (рисуное 3).

```

switch (n)

```

```

{
    case 'A': case 'a': Console.WriteLine("арбуз"); break;
    case 'Д': case 'д': Console.WriteLine("дыня"); break;
case 'C': case 'c': Console.WriteLine("слива"); break;
    default: Console.WriteLine("ошибка"); break;
}

```

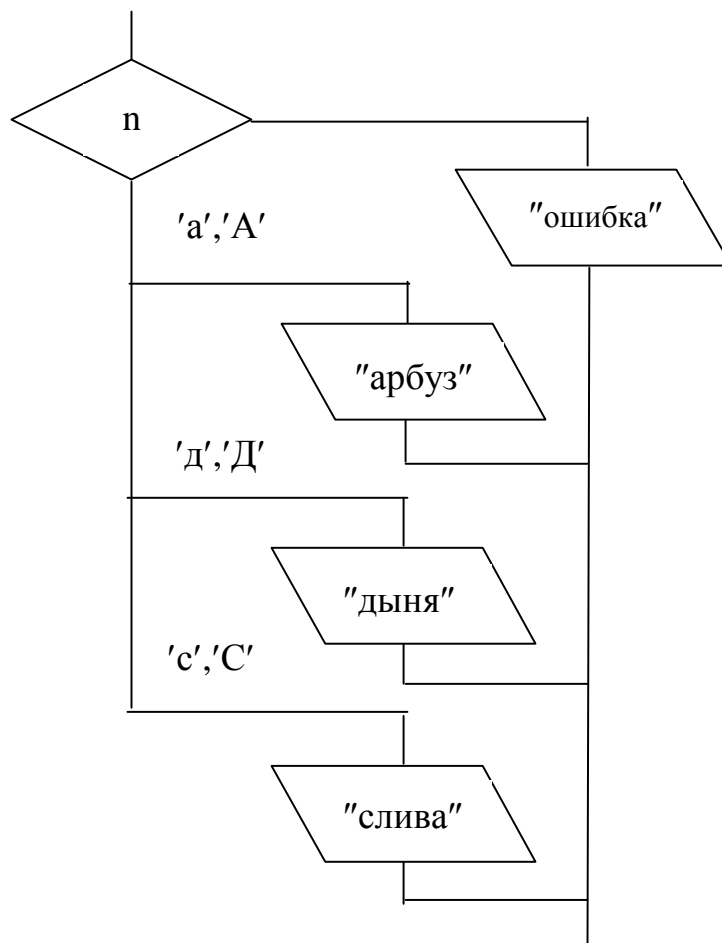


Рисунок 3 – Схема алгоритма оператора выбора с селектором СИМВОЛЬНОГО типа

В первом и во втором примерах показан способ задания размещения подряд нескольких меток.

### Пример программирования разветвленного алгоритма

**Задание.** Вычислите значение функции

$$f = \begin{cases} \sin^2 x, & \text{если } x < 0, \\ \cos x^2, & \text{если } 0 \leq x < 1.5, \\ x, & \text{если } x \geq 1.5. \end{cases}$$

1. На рисунке 4 – разработка алгоритма:

- входные данные:  $x$  – действительная переменная;
- выходные данные:  $f$  – действительная переменная.

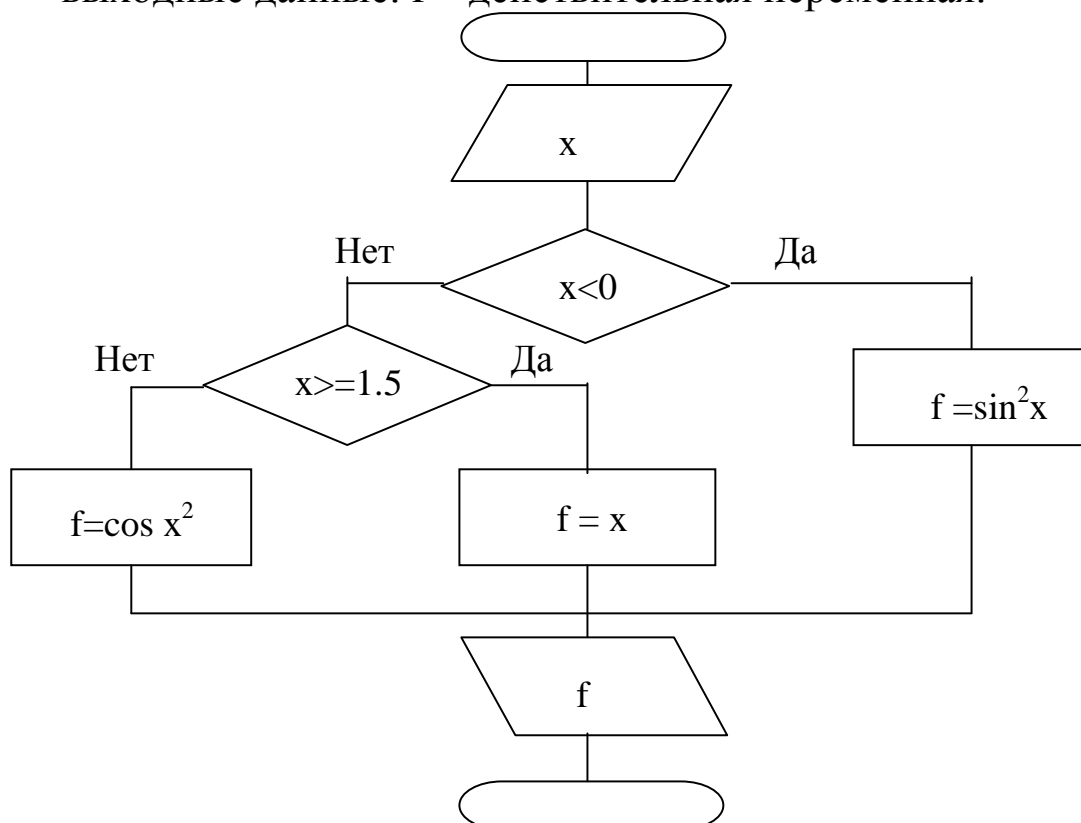


Рисунок 4 – Схема алгоритма программы

2. Разработка формы – таблица 2, рисунок 5.

Таблица 2 – Используемые компоненты

Имя компонента	Настраиваемое свойство	Значение
1. Form1	Text	Пример лабораторной работы № 3

Имя компонента	Настраиваемое свойство	Значение
2. TextBox1	Text	
3. TextBox2	Text	
4. Label1	Text	Введите x
5. Label2	Text	Результат
6. Button1	Text	Высчитать

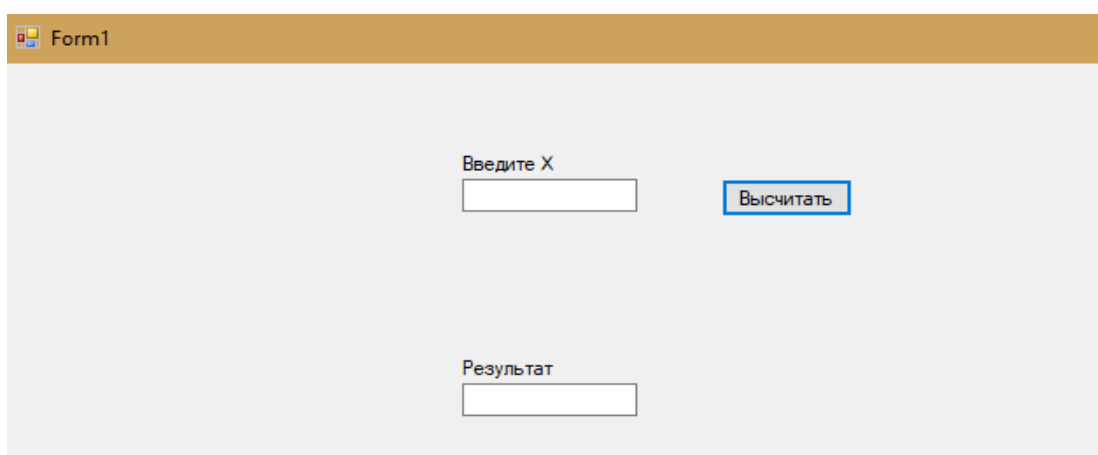


Рисунок 3.5 — Внешний вид формы

## 3. Текст программы:

```
//-----
using System;
using System.Windows.Forms;
namespace IfElse
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        //-----
        private void Button1_Click(object sender, EventArgs e)
        {
            double x = Convert.ToDouble (TextBox1.Text);
            if (x < 0)
                TextBox2.Text = Convert.ToString (Math.Pow(Math.Sin(x), 2));
            else if (x < 1.5)
```

```

        TextBox2.Text = Convert.ToString (Math.Cos(Math.Pow(x,2)));
    else
        TextBox2.Text = Convert.ToString (x);
    }
}
}

```

## Индивидуальные задания

Решите две задачи первого и второго уровней сложности.

### Задачи первого уровня сложности

1. Дано четырехзначное положительное целое число. Напишите программу, которая печатает true или false в зависимости от того, выполняется или нет указанное условие: сумма 1 и 4 цифры равна произведению 2 и 3 цифры.

2. Даны действительные числа  $x$ ,  $y$ . Если  $x$  и  $y$  отрицательны, то каждое значение замените его модулем; если отрицательно только одно из них, то оба значения увеличьте на 5.5; в остальных случаях оба значения уменьшите в 10 раз.

3. Даны действительные числа  $x$ ,  $y$ . Если  $x$  и  $y$  – положительны, то каждое значение уменьшить в 5 раз; если оба значения лежат в диапазоне  $[-3, 9]$ , то оба значения заменить на 0; в остальных случаях  $x$  и  $y$  увеличьте на 3.

4. Даны действительные числа  $x$ ,  $y$  ( $x \diamond y$ ). Меньшее из этих двух чисел замените их полусуммой, а большее – их удвоенным произведением.

5. Если сумма двух различных действительных чисел  $x$ ,  $y$  меньше 1, то наименьшее из этих двух чисел замените произведением этих чисел; в противном случае замените большее из  $x$  и  $y$  полусуммой этих чисел.

6. Определите, является ли шестизначное целое число счастливым. (Число называется счастливым, если сумма первых трех цифр равна сумме его последних трех цифр.)

7. Для решения следующей задачи написать программу, которая печатает true или false в зависимости от того, выполняется или нет указанное условие: для произвольных вещественных чисел

$a$ ,  $b$ ,  $c$  определите, имеет ли уравнение  $ax^2+bx+c=0$  хотя бы одно вещественное решение.

8. Для решения следующей задачи написать программу, которая печатает `true` или `false` в зависимости от того, выполняется или нет указанное условие: равна ли сумма двух первых цифр заданного четырехзначного числа сумме двух его последних чисел.

9. В некотором учебном заведении действуют следующие правила приема. Абитуриенты сдают два экзамена, которые оцениваются по 100 баллов каждый. Если абитуриент набирает не менее 150 баллов, то это дает право поступать на дневное отделение, от 100 до 149 – на вечернее отделение; ниже 100 баллов означает отказ в приеме на учебу. Напишите программу, которая в зависимости от суммы набранных баллов сообщает абитуриенту его права на поступление.

10. Напишите программу, позволяющую получить словесное описание отметок (1 – «плохо», 2 – «неудовлетворительно», 3 – «удовлетворительно», 4 – «хорошо», 5 – «отлично»).

11. Для решения следующей задачи напишите программу, которая печатает `true` или `false` в зависимости от того, выполняется или нет указанное условие. Определите, есть ли среди цифр заданного трехзначного числа одинаковые.

12. Напишите программу, которая печатает `true`, если дата  $d1$ ,  $m1$  предшествует (в рамках года) дате  $d2$ ,  $m2$  и значение `false` в противном случае.

13. Напишите программу с использованием оператора выбора, которая по первой букве имени выводит на экран имя полностью (Иван, Петр, Николай, Владимир, Георгий).

14. Напишите программу с использованием оператора выбора, которая по введенной цифре  $0 \leq k \leq 5$  печатает название этой цифры на русском и английском языках.

15. Дано целое число  $k$  ( $1 \leq k \leq 365$ ). Определите, каким будет  $k$ -й день года – выходным или рабочим, если 1 января – понедельник.

16. Дано действительное число  $x$ . Вычислите  $f$ , если:

$$f = \begin{cases} 5 & \text{при } x < 0, \\ x^2 - x & \text{при } 0 \leq x < 1, \\ x^{11} - \sin \pi x^2 & \text{в остальных случаях.} \end{cases}$$

17. Напишите программу с использованием оператора выбора, которая по первой букве названия города выводит на экран название города полностью (Курск, Москва, Тула, Новгород, Воронеж).

18. Напишите программу с использованием оператора выбора, которая по первой букве фамилии выводит на экран фамилию полностью (Иванов, Петров, Сидоров, Мышкин, Шишкин).

19. Даны координаты центра и радиус окружности. Определите принадлежность некоторой точки, имеющей координаты  $(x, y)$ , окружности.

20. Даны координаты левого верхнего и правого нижнего углов прямоугольника, расположенного на плоскости. Определите принадлежность некоторой точки, имеющей координаты  $(x, y)$ , прямоугольнику.

21. Даны координаты левого верхнего и правого нижнего углов квадрата, расположенного на плоскости. Определите принадлежность некоторой точки, имеющей координаты  $(x, y)$ , квадрату.

22. Напишите программу с использованием оператора выбора, которая по введенной первой букве названия животного печатает полностью название животного (собака, кошка, жираф, лошадь, обезьяна).

23. Даны два различных числа  $a, b$ . Упорядочьте эти числа по возрастанию таким образом, чтобы  $a$  соответствовало наименьшее число,  $b$  – наибольшее.

24. Напишите программу с использованием оператора выбора, которая по номеру (от 1 до 7) выводит название дня недели.

25. Напишите программу с использованием оператора выбора, которая по номеру (от 1 до 12) выводит название месяца года.

26. Напишите программу с использованием оператора выбора, которая по введенной первой букве названия сезона года печатает полностью название сезона года.

27. Дано пятизначное положительное целое число. Напишите

программу, которая печатает true или false в зависимости от того, выполняется или нет указанное условие: сумма 1, 3 и 5 цифры больше произведения 2 и 4 цифры.

28. Дано четырехзначное положительное целое число. Напишите программу, которая печатает true или false в зависимости от того, выполняется или нет указанное условие: среднее арифметическое 1 и 4 цифры больше суммы 2 и 3 цифры.

29. Даны цифра и трехзначное число в десятичной системе счисления. Если цифра совпадает со старшей цифрой числа, то вычислите сумму цифр числа. Если цифра совпадает со средней цифрой числа, то вычислите произведение цифр числа. Если цифра совпадает с младшей цифрой числа, то вычислите среднее арифметическое цифр числа. Если ни одно из условий не выполняется, то вычислите среднее геометрическое цифр числа.

30. Даны цифра и трехзначное число в десятичной системе счисления. Если цифра совпадает со старшей цифрой числа, то вычислите среднее арифметическое цифр числа. Если цифра совпадает со средней цифрой числа, то вычислите среднее геометрическое цифр числа. Если цифра совпадает с младшей цифрой числа, то вычислите сумму цифр числа. Если ни одно из условий не выполняется, то вычислите произведение цифр числа.

### **Задачи второго уровня сложности**

1. Дано целое двузначное положительное число в троичной системе счисления. Напишите программу, которая выводит названия цифр числа на русском языке, если старшая цифра числа больше младшей, иначе – на английском языке.

2. Дано целое двузначное положительное число в четверичной системе счисления. Напишите программу, которая выводит названия цифр числа на английском языке, если старшая цифра числа больше младшей, иначе – на русском языке.

3. Дано целое двузначное положительное число в пятеричной системе счисления. Напишите программу, которая выводит названия цифр числа на английском языке, если старшая цифра числа больше младшей ровно в 2 раза, иначе – на русском языке.

4. Дано целое двузначное положительное число в троичной

системе счисления. Напишите программу, которая выводит названия цифр числа на русском языке, если старшая цифра числа меньше младшей, иначе – на английском языке.

5. Дано целое двузначное положительное число в пятеричной системе счисления. Напишите программу, которая выводит названия цифр числа на русском языке.

6. Дано целое двузначное положительное число в четверичной системе счисления. Напишите программу, которая выводит названия цифр числа на русском и английском языках.

7. Дано целое двузначное положительное число в четверичной системе счисления. Напишите программу, которая выводит, если старшая цифра меньше младшей цифры, названия цифр числа на английском языке, а иначе – на русском языке.

8. Дано целое двузначное положительное число в семеричной системе счисления. Напишите программу, которая выводит, если цифры расположены по убыванию, названия цифр числа на английском языке, а иначе – на русском языке.

9. Даны две цифры в десятичной системе счисления. Напишите программу, которая выводит названия на английском языке цифр суммы этих цифр.

10. Даны две цифры в десятичной системе счисления. Напишите программу, которая выводит названия на русском языке цифр произведения этих цифр.

11. Даны две цифры в десятичной системе счисления. Напишите программу, которая выводит название на русском языке цифр модуля разности этих цифр.

12. Даны две цифры в десятичной системе счисления. Напишите программу, которая выводит названия цифр на английском языке частного и остатка целочисленного деления этих чисел.

13. Дано целое двузначное положительное число в шестеричной системе счисления. Напишите программу, которая выводит, если старшая цифра больше в три раза младшей цифры, названия цифр числа на английском языке, а иначе – на русском языке.

14. Дано целое двузначное положительное число в шестеричной системе счисления. Напишите программу, которая выво-



дит название цифры числа, имеющей большее значение, на английском языке, а название цифры числа, имеющей меньшее значение, – на русском языке.

15. Дано целое двузначное положительное число в семеричной системе счисления. Напишите программу, которая выводит название цифры числа, имеющей большее значение, на английском языке, а название цифры числа, имеющей меньшее значение, – на русском языке.

16. Дано целое двузначное положительное число в шестеричной системе счисления. Напишите программу, которая выводит названия цифр числа на английском языке.

17. Даны две цифры в троичной системе счисления. Напишите программу, которая выводит названия цифр суммы этих чисел на русском языке.

18. Даны две цифры в двоичной системе счисления. Напишите программу, которая выводит названия цифр суммы этих чисел на английском языке.

19. Дано целое двузначное положительное число в пятеричной системе счисления. Напишите программу, которая выводит, если цифры расположены по возрастанию, названия цифр числа на английском языке, а иначе – на русском языке.

20. Дано целое двузначное положительное число в шестеричной системе счисления. Напишите программу, которая выводит, если старшая цифра больше в пять раз младшей цифры, названия цифр числа на русском языке, а иначе – на английском языке.

21. Даны две цифры в восьмеричной системе счисления. Напишите программу, которая выводит названия на русском языке цифры модуля разности этих цифр.

22. Даны две цифры в восьмеричной системе счисления. Напишите программу, которая выводит названия на русском языке цифр суммы этих двух цифр.

23. Дано целое двузначное положительное число в семеричной системе счисления. Напишите программу, которая выводит, если старшая цифра больше в три раза младшую, названия цифр числа на русском языке, а иначе – на английском языке.

24. Дано целое двузначное положительное число в восьме-

ричной системе счисления. Напишите программу, которая выводит название цифры числа, имеющей большее значение, на русском языке, а название цифры числа, имеющей меньшее значение, □ на английском языке.

25. Дано целое двузначное положительное число в семеричной системе счисления. Напишите программу, которая выводит название цифр числа, имеющих значение меньше 4, на русском языке, а название цифр числа, имеющих значение большее или равное 4, – на английском языке.

26. Дано целое двузначное положительное число в шестеричной системе счисления. Напишите программу, которая выводит название цифр числа, имеющих значение меньше 3, на русском языке, а название цифр числа, имеющих значение большее или равное 3, – на английском языке.

27. Даны две первые буквы (без учета регистра, как строчные, так и прописные) названия месяца года. Напишите программу, которая выводит полное название месяца года на русском языке.

28. Даны две первые буквы (без учета регистра, как строчные, так и прописные) названия дня недели. Напишите программу, которая выводит полное название дня недели на русском языке.

29. В старояпонском календаре был принят 60-летний цикл, состоявший из пяти 12-летних подциклов. Подциклы обозначались названиями цвета: зеленый, красный, желтый, белый и черный. Внутри каждого подцикла годы носили названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи (1984 год – год зеленой крысы – был началом очередного цикла). Напишите программу, которая вводит номер некоторого года нашей эры и печатает его название по старояпонскому календарю.

30. Даны две цифры в четверичной системе счисления. Напишите программу, которая выводит названия цифр суммы этих чисел на русском языке.

### **Содержание отчета**

Отчет по лабораторной работе включает:

– титульный лист;

- условие задания;
- алгоритм решения задачи;
- текст программы;
- результаты тестирования программы.

### Контрольные вопросы к защите лабораторной работы

1. Вычислите значения выражений:

$a \parallel b \ \&\& \ !a$  при  $a=true$  и  $b=false$ ;

$t \ \&\& \ (p \% 3 == 0)$  при  $t=true$ ,  $p=1011$ ;

$(x*y) \ \&\& \ (y>x)$  при  $x=2$ ,  $y=1$ ;

$(x*y!=0) \parallel (y>x)$  при  $x=2$ ,  $y=1$ ;

$!(a \ \&\& \ b)$  при  $a=false$ ,  $b=true$ .

2. Запишите на **C#** выражение, истинное при выполнении данного условия:

$x$  принадлежит отрезку  $[0,1]$ ;

$x$  лежит вне отрезка  $[0,1]$ ;

$x$  принадлежит отрезку  $[2,5]$  или  $[-1,1]$ ;

$x$  лежит вне отрезка  $[2,5]$  или  $[-1,1]$ .

3. Поясните структуру и правила выполнения условных операторов.

4. Запишите указанное действие в виде одного условного оператора:

$$y = \begin{cases} \cos^2 x, & \text{при } 0 < x < 2; \\ 1 - \sin^2 x, & \text{иначе.} \end{cases}$$

5. Какое значение будет иметь переменная  $z$  после выполнения операторов?

```
int z=0, x=1, y=-1;
```

```
if (x>0) if (y>0) z=1; else z=2;
```

6. Какое значение будет иметь переменная  $z$  после выполнения операторов?

```
int z=0, x=-1, y=1;
```

```
if (x) { if (y>0) z=1;} else z=2;
```

7. Какое значение будет иметь переменная z после выполнения операторов?

```
int z=0, x=0, y=1;
if (x) { if (y>0) z=1;} else z=2;
```

8. Какое значение будет иметь переменная z после выполнения операторов?

```
int z=0, x=3, y=1;
if (x) { if (y) z=1;} else z=2;
```

9. Укажите ошибки в следующих операторах:

```
if (1<x<2) x=x+1; y:=0;
else x=0; y+= 1;
```

10. Какое значение будет иметь переменная z после выполнения операторов?

```
int z=0, x=0, y=1;
if (!x) { if (!(y-1)) z=1;} else z=2;
```

11. Какое значение будет иметь переменная z после выполнения операторов:

```
int z=0, x=1, y=1;
if (!x) { if (!(y-1)) z=1;} else z=2;
```

12. Найдите ошибки во фрагменте программы:

```
{int n,x;
switch (k)
  case + : x:=x-4 break;
  case '-','*': x=5;
}
```

Какие правила нарушены и как можно исправить ошибки?

13. Какие типы можно использовать для выражения–селектора и констант в операторе switch?

14. Как устанавливается соответствие между if и else при вложениях условных операторов?

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Подбельский В.В. Язык С#. Базовый курс. [Текст]/ В.В. Подбельский. – М.: Финансы и статистика, 2013. – 427 с.
2. Троелсен, Эндрю. Язык программирования С# 5.0 и платформа .NET 4.5, 6-е изд. [Текст]/ Эндрю Троелсен – М.: ООО «И.Д. Вильямс», 2013. – 1312 с.
3. Макконнелл, Стив. Совершенный код. Мастер-класс. [Текст]/ Пер. с англ. – М. : Издательство «Русская редакция», 2010. – 896 с.
4. Техническая документация, материалы по API и примеры кода [Электронный ресурс]// Режим доступа – <https://docs.microsoft.com> (дата обращения: 26.05.2019).
5. METANIT.COM – Сайт о программировании [Электронный ресурс]// Режим доступа – <https://metanit.com> (дата обращения: 6.06.2019).
6. MSDN – сеть разработчиков Microsoft [Электронный ресурс]// Режим доступа – <https://msdn.microsoft.com> (дата обращения: 27.05.2019).
7. Шилдт, Г. С# 4.0: полное руководство. Пер. с англ. [Текст]/ Герберт Шилдт. – М.: ООО "И.Д. Вильямс", 2011. - 1056 с.
8. Википедия – свободная энциклопедия [Электронный ресурс]// Режим доступа – [https://ru.wikipedia.org/wiki/ Заглавная\\_страница](https://ru.wikipedia.org/wiki/Заглавная_страница) (дата обращения: 24.05.2019).