

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»
Филиал
«Минский радиотехнический колледж»

Учебный предмет
«Конструирование программ и языки программирования»

Инструкция
по выполнению лабораторной работы №4
«Разработка, отладка и испытание программ с методами»

Минск 2024 г.

Лабораторная работа № 4

Тема работы: «Разработка, отладка и испытание программ с методами»

1 Цель работы

Обучить разработке и перегрузке методов с параметрами различного статуса, использованию встроенных методов

2 Задание

Задание 1. Используя методы создайте программу согласно варианту.

1. Дано натуральное число n . Выясните, можно ли представить данное число в виде произведения трех последовательных натуральных чисел.
2. Дано натуральное число m . Укажите все тройки натуральных чисел x , y и z , удовлетворяющие следующему условию: $m = x^3 + y^3 + z^3$.
3. Среди всех четырехзначных номеров машин, определите количество номеров, содержащих только три одинаковые цифры.
4. Среди всех четырехзначных номеров машин, определите количество номеров, содержащих три или более одинаковых цифры.
5. Среди всех четырехзначных номеров машин, определите количество номеров машин, содержащих только две одинаковые цифры.
6. Среди всех четырехзначных номеров машин, определите количество номеров машин, содержащих только две или более одинаковых цифры.
7. Напишите программу нахождения, следующего за данным совершенного числа. Совершенным называется число, сумма делителей которого, не считая самого числа, равна этому числу. Первое совершенное число 6 ($6 = 1 + 2 + 3$).
8. Проверьте, является ли данное натуральное число простым.
9. Дано натуральное число P . Напишите программу нахождения всех натуральных чисел, не превосходящих P , которые можно представить в виде произведения двух простых чисел.
10. Дано натуральное число P , заданное в восьмеричной системе счисления. Напишите программу перевода этого числа в двоичную систему счисления. Встроенный метод перевода не использовать.
11. Дано натуральное число P , заданное в шестнадцатеричной системе счисления. Переведите его в двоичную систему счисления. Встроенный метод перевода не использовать.
12. Дано натуральное число P . Найдите все «совершенные» числа, не превосходящие P . Совершенным, называется число, сумма делителей которого, не считая самого числа, равна этому числу. Первое совершенное число 6 ($6 = 1 + 2 + 3$).
13. Дано натуральное n -значное число P . Проверьте, является ли данное число палиндромом (перевертышем).
14. Дано натуральное число P . Проверьте, кратно ли P трем, используя признак делимости на 3.

15. Дано натуральное число P . Проверьте, кратно ли P одиннадцати, используя признак делимости на 11 (знакопеременная сумма его цифр делится на 11).
16. Дано натуральное число P . Найдите все простые числа, не превосходящие числа P .
17. Дано натуральное число P . Найдите все делители числа P .
18. Дано натуральное число P . Найдите сумму цифр числа P .
19. Дано натуральное число P . Выбросите из записи числа P цифры 0, оставив прежним порядок остальных цифр.
20. Дано натуральное число P . Проверьте, кратно ли число P девяти, используя признак делимости на 9.
21. Дана обыкновенная дробь $\frac{m}{n}$. Сократите данную дробь.
22. Напишите программу сложения двух обыкновенных несократимых дробей $\frac{m}{n}$ и $\frac{p}{q}$. Результат представить в виде несократимой дроби.
23. Напишите программу вычитания двух обыкновенных несократимых дробей $\frac{m}{n}$ и $\frac{p}{q}$. Результат представить в виде несократимой дроби.
24. Напишите программу умножения двух обыкновенных несократимых дробей $\frac{m}{n}$ и $\frac{p}{q}$. Результат представить в виде несократимой дроби.
25. Напишите программу деления двух обыкновенных несократимых дробей $\frac{m}{n}$ и $\frac{p}{q}$. Результат представить в виде несократимой дроби.
26. Данное натуральное число N переведите из десятичной системы счисления в двоичную. Встроенный метод перевода не использовать.
27. Данное натуральное число N замените суммой квадратов его цифр. Произведите K таких замен.
28. Дано натуральное число n . Найдите все меньшие n числа Мерсена. Простое число называется числом Мерсена, если оно может быть представлено в виде $2^p - 1$, где p – тоже простое число.
29. Дано натуральное число N . Найдите все составные натуральные числа, меньшие N .

Задание 2. Используя перегрузку методов, создайте программу согласно варианту.

1.
 - а) для сложения целых чисел;
 - б) для сложения комплексных чисел.

2.

- а) для сложения вещественных чисел;
 - б) для сложения комплексных чисел.
- 3.
- а) для умножения целых чисел;
 - б) для умножения комплексных чисел.
- 4.
- а) для вычитания целых чисел;
 - б) для вычитания комплексных чисел.
- 5.
- а) для умножения вещественных чисел;
 - б) для умножения комплексных чисел.
- 6.
- а) для вычитания вещественных чисел;
 - б) для вычитания комплексных чисел.
- 7.
- а) для деления целых чисел;
 - б) для деления комплексных чисел.
- 8.
- а) по номеру года выдает его название по старояпонскому календарю;
 - б) по названию месяца выдает знак Зодиака.
- 9.
- а) для сложения десятичных дробей;
 - б) для сложения обыкновенных дробей.
- 10.
- а) для вычитания десятичных дробей;
 - б) для вычитания обыкновенных дробей.
- 11.
- а) для умножения десятичных дробей;
 - б) для умножения обыкновенных дробей.
- 12.
- а) для деления десятичных дробей;
 - б) для деления обыкновенных дробей.
- 13.
- а) для преобразования десятичной дроби в обыкновенную;
 - б) для преобразования обыкновенной дроби в десятичную.
- 14.
- а) для вычисления натурального логарифма;
 - б) для вычисления десятичного логарифма.
- 15.
- а) целые числа возводит в степень n ;
 - б) из десятичных чисел извлекает корень степени n .
- 16.
- а) для перевода часов и минут в минуты;
 - б) для перевода минут в часы и минуты.

17.

- а) для последовательности целых чисел находит среднее арифметическое;
- б) для строки находит количество букв, содержащихся в ней.

18.

- а) для последовательности целых чисел находит максимальный элемент;
- б) для строки находит длину самого длинного слова.

19.

- а) для последовательности целых чисел находит минимальный элемент;
- б) для строки находит длину самого короткого слова.

20.

- а) для последовательности целых чисел находит количество четных элементов;
- б) для строки находит количество слов, начинающихся на букву «а».

21.

- а) для последовательности целых чисел находит количество отрицательных элементов;
- б) для строки находит количество слов, заканчивающихся и начинающихся на одну и ту же букву.

22.

- а) для последовательности целых чисел находит количество нечетных элементов;
- б) для строки находит количество слов в ней.

23.

- а) для последовательности, начинающейся на четное число, выполняет циклический сдвиг влево на количество элементов равное первому элементу последовательности.
- б) для последовательности, начинающейся на нечетное число, выполняет циклический сдвиг вправо на количество элементов равное последнему элементу последовательности.

24.

- а) для последовательности целых чисел удаляет все четные элементы из последовательности;
- б) для строки удаляет все четные слова.

25.

- а) для последовательности удаляет все четные элементы;
- б) для последовательности удаляет все элементы, заключенные между двумя нулевыми элементами.

26.

- а) для последовательности целых чисел находит количество простых чисел;
- б) для строки находит количество слов, заканчивающихся и начинающихся на одну и ту же букву.

27.

- а) для последовательности целых чисел находит количество нечетных элементов;

б) для строки находит количество пробелов в ней.

28.

а) для последовательности, начинающейся на четное число, выполняет обрезку всех элементов после четного числа.

б) для последовательности, начинающейся на нечетное число, выполняет обрезку всех элементов, начиная с 4-го символа после нечетного числа.

29.

а) для последовательности целых чисел удаляет все четные элементы из последовательности;

б) для строки удаляет все нечетные слова.

3 Оснащение работы

ПК, среда Visual Studio 2019, MSword.

4 Основные теоретические сведения

Метод — это именованный блок кода, который выполняет некоторые действия.

Общее определение методов выглядит следующим образом:

```
[модификаторы]           тип_возвращаемого_значения
название_метода ( [параметры] )
{
    // тело метода
}
```

Модификаторы и параметры необязательны.

Чтобы использовать методы в программе, нам надо вызвать их в методе Main.

Для вызова метода указывается его имя, после которого в скобках идут значения для его параметров (если метод принимает параметры).

```
название_метода (значения_для_параметров_метода) ;
```

Преимуществом методов является то, что их можно повторно и многократно вызывать в различных частях программы.

Метод может возвращать значение, какой-либо результат.

Если метод имеет любой другой тип, отличный от void, то такой метод обязан вернуть значение этого типа.

Для этого применяется оператор return, после которого идет возвращаемое значение:

```
return возвращаемое значение ;
```

После оператора return также можно указывать сложные выражения, которые возвращают определенный результат.

При этом методы, которые в качестве возвращаемого типа имеют любой тип, отличный от void, обязательно должны использовать оператор return для возвращения значения.

Также между возвращаемым типом метода и возвращаемым значением после оператора return должно быть соответствие. Например, в следующем

случае возвращаемый тип - int, но метод возвращает строку (тип string), поэтому такое определение метода некорректно:

```
static int GetSum()
{
    int x = 2;
    int y = 3;
    return "5"; // ошибка - надо возвращать число
}
```

Результат методов, который возвращают значение, мы можем присвоить переменным или использовать иным образом в программе.

Оператор return не только возвращает значение, но и производит выход из метода. Поэтому он должен определяться после остальных инструкций.

```
static string GetHello()
{
    return "Hello";
    Console.WriteLine("After return");
}
```

С точки зрения синтаксиса данный метод корректен, однако его инструкция Console.WriteLine("After return") не имеет смысла - она никогда не выполнится, так как до ее выполнения оператор return возвратит значение и произведет выход из метода.

Однако мы можем использовать оператор return и в методах с типом void. В этом случае после оператора return не ставится никакого возвращаемого значения (ведь метод ничего не возвращает).

Типичная ситуация - в зависимости от определенных условий произвести выход из метода:

```
static void SayHello()
{
    int hour = 23;
    if(hour > 22)
    {
        return;
    }
    else
    {
        Console.WriteLine("Hello");
    }
}
```

Если метод в качестве тела определяет только одну инструкцию, то мы можем сократить определение метода. Например, допустим у нас есть метод:

```
static void SayHello()
{
    Console.WriteLine("Hello");
}
```

Можно его сократить следующим образом:

```
static void SayHello() =>
Console.WriteLine("Hello");
```

Подобным образом мы можем сокращать методы, которые возвращают значение.

Параметры позволяют передать в метод некоторые входные данные.

Значения, которые передаются параметрам, еще называются аргументами.

Иногда можно встретить такие определения как формальные параметры и фактические параметры.

Формальные параметры — это собственно параметры метода (в данном случае *x* и *y*), а фактические параметры - значения, которые передаются формальным параметрам. То есть фактические параметры — это и есть аргументы метода.

C# также позволяет использовать необязательные параметры. Для таких параметров нам необходимо объявить значение по умолчанию. Также следует учитывать, что после необязательных параметров все последующие параметры также должны быть необязательными.

Существует два способа передачи параметров в метод в языке C#: по значению и по ссылке.

Наиболее простой способ передачи параметров представляет передача по значению, по сути, это обычный способ передачи параметров.

При передаче параметров по ссылке перед параметрами используется модификатор `ref`.

Параметры могут быть также выходными. Чтобы сделать параметр выходным, перед ним ставится модификатор `out`:

```
static void Sum(int x, int y, out int a)
{
    a = x + y;
}
```

Как и в случае с `ref` ключевое слово `out` используется как при определении метода, так и при его вызове.

Кроме выходных параметров с модификатором `out` метод может использовать входные параметры с модификатором `in`. Модификатор `in` указывает, что данный параметр будет передаваться в метод по ссылке, однако внутри метода его значение параметра нельзя будет изменить.

Перегрузка метода как раз заключается в том, что методы имеют разную сигнатуру, в которой совпадает только название метода. То есть методы должны отличаться по:

- количеству параметров;
- типу параметров;
- порядку параметров;
- Модификаторам параметров.

Для создания встроенного метода существует два способа:

- через использование анонимного метода;

– с помощью лямбда-выражения.

Их функциональность практически идентична, различия касаются формы записи.

5. Порядок выполнения работы

1. Выделить ключевые моменты задачи.
2. Построить алгоритм и теоретическую объектную модель решения задачи.
3. Запрограммировать полученные алгоритмы и объектную модель.

6. Форма отчета о работе

Лабораторная работа № _____

Номер учебной группы _____

Фамилия, инициалы учащегося _____

Дата выполнения работы _____

Тема работы: _____

Цель работы: _____

Оснащение работы: _____

Результат выполнения работы: _____

7. Контрольные вопросы и задания

1. Что такое метод?
2. Назовите типы методов?
3. Для чего применяется оператор return?
4. В каком случае можно сократить запись метода?
5. Что такое аргументы?
6. Формальные параметры – это...
7. Фактические параметры – это...
8. Для чего служит перегрузка методов?
9. Назовите способы создания встроенных методов.
10. Для чего служат операторы ref, out, in?

8. Рекомендуемая литература

1. Рихтер, Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Дж. Рихтер. СПб. : Изд-во Питер, 2021. 896 с.

2. Прайс, М. Дж. C# 10 и .NET 6. Современная кросс-платформенная разработка / М. Дж. Прайс. СПб : Изд-во Питер, 2023. 848 с.

3. Васильев, А.Н. Программирование на C# для начинающих. Особенности языка / А.Н. Васильев. М. : Эксмо, 2022. 528 с.

4. Фримен, А. ASP.NET Core 3 с примерами на C# для профессионалов / А. Фримен. СПб. : Изд-во Вильямс, 2021. 1184 с.