

# Прикладное программирование

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ  
РОСТУНЦЕВ САВВА ДМИТРИЕВИЧ

## Оглавление

Критерий оценок .....	4
Правила оформления.....	5
Структура отчетов .....	5
Общие требования.....	5
Построение пояснительной записки.....	5
Нумерация страниц .....	6
Нумерация разделов, подразделов, пунктов, подпунктов пояснительной записки .....	6
Перечисления.....	7
Иллюстрации .....	8
Таблицы.....	9
Примечания.....	10
Формулы и уравнения.....	11
Ссылки.....	11
Приложения .....	11
Лабораторная работа №1 .....	13
Цель и задачи .....	13
Описание задания.....	13
Написание echo-приложение с анализом сообщений. ....	13
Написать приложение для решения ряда задач.....	13
Технические требования.....	14
Варианты индивидуальных заданий.....	15
Теория и примеры реализаций .....	22
Создание консольного приложения.....	22
Пример создания отдельного класса .....	23
Работа в Debug-режиме.....	24
Лабораторная работа №2 .....	26
Цель и задачи .....	26
Описание задания.....	26
Выделение части строки по двум разделителям используя класс String .....	26
Нахождение суммы и среднего значения, мин. и макс. целочисленного списка.....	26
Работа со списками .....	26
2. Словарь.....	27
Лабораторная работа №3 .....	28
Цель и задачи .....	28
Описание задания.....	28
Технические требования.....	28
Теория и примеры реализации .....	29

Создания проекта и описание класса службы .....	29
Добавление установщиков в службу .....	31
Лабораторная работа №4 .....	32
Описание задания.....	32
Технические требования.....	32
Лабораторная работа №5 .....	33
Описание задания.....	33
Технические требования.....	33
ПРИЛОЖЕНИЕ А .....	34

## Критерий оценок

Критерий оценок практических занятий по дисциплине «прикладное программное обеспечение»

Тип занятия	Контрольное мероприятие	Критерии оценки	Количество баллов
Лабораторные работы	Посещение лабораторных занятий (18)	Присутствие на всем протяжении лабораторного занятия	1 посещение лабораторной работы = 1 балл
	Выполнение лабораторной работы № 1–4	Работа сдана в указанный срок (в указанный преподавателем срок)	3
		Правильно выполненная работа в соответствии с требованиями в методическом материале	3
		Правильные ответы на вопросы по выполненной работе	5
		Оформление отчета по лабораторным работам в соответствии с методическими указаниями	5
	Выполнение лабораторной работы № 5	Работа сдана в указанный срок (в указанный преподавателем срок)	3
		Правильно выполненная работа в соответствии с требованиями в методическом материале	5
		Правильные ответы на вопросы по выполненной работе	5
		Оформление отчета по лабораторным работам в соответствии с методическими указаниями	5

## Правила оформления

### Структура отчетов

- титульный лист (см. приложение А);
- содержание;
- введение (цели и задачи);
- основная часть:
  - формулирование индивидуального задания;
  - описание библиотек, классов и основных методов, использованных при разработке приложения;
  - результаты разработанного приложения в виде скриншотов;
  - анализ полученных результатов;
- заключение (выводы о проделанной работе);

приложения (текст программы с комментариями).

### Общие требования

Отчет должен быть выполнен любым печатным способом на пишущей машинке или с использованием компьютера и принтера на одной стороне листа белой бумаги формата А4 через полтора интервала. Цвет шрифта должен быть черным, высота букв, цифр и других знаков — не менее 1,8 мм (кегель не менее 14 пт), шрифт Times New Roman.

Текст отчета следует печатать, соблюдая следующие размеры полей: правое – 10 мм, верхнее – 20 мм, левое – 30 мм и нижнее – 25 мм.

Текст пояснительной записки следует размещать, соблюдая следующие размеры:

- расстояние от рамки и в конце строк не менее 3 мм;
- расстояние от текста до верхней или нижней рамки должно быть не менее 10 мм;
- расстояние между заголовком и текстом должно быть равно 15 мм (при рукописном способе) или 2 интервала (при машинном способе);
- расстояние между заголовками раздела и подраздела 8 мм или 1 интервал;
- абзацы в тексте начинают отступом, равным 15–17 мм
- абзацы в тексте выравнивают по ширине страницы.

### Построение пояснительной записки

Наименования структурных элементов пояснительной записки «Реферат», «Содержание», «Нормативные ссылки», «Определения», «Обозначения и сокращения», «Введение», «Заключение», «Список использованных источников», «Приложение А» и так далее служат заголовками структурных элементов пояснительной записки.

Основную часть записки следует делить на разделы (уровень заголовков структурных элементов), подразделы, пункты и, при необходимости, на подпункты. Они должны начинаться с абзацного отступа и иметь нумерацию арабскими цифрами.

Точка между номером раздела (подраздела, пункта, подпункта) и названием не ставится. Например, «1 Постановка задачи», «1.3 Характеристика объекта», «1.5.2 Требования к надежности», «1.5.2.3 Требования к надежности технических средств и программного обеспечения».

Таким образом, нижние уровни содержат информацию о соответствующем номере верхнего уровня. Номера нижнего и верхнего уровня разделяются точкой.

Разделы и подразделы должны иметь заголовки. Пункты могут не иметь заголовков. Если заголовок состоит из двух предложений, их разделяют точкой. Переносы слов в заголовках не допускаются.

Если раздел состоит из одного подраздела, то подраздел не нумеруется. Если подраздел состоит из одного пункта, то пункт не нумеруется. И так далее. Таким образом, наличие одного подраздела в разделе эквивалентно фактическому отсутствию других.

Внутри пунктов и подпунктов могут быть приведены перечисления. Перед каждым перечислением следует ставить дефис или, при необходимости ссылки в тексте документа на одно из перечислений, строчную букву (за исключением ё, з, й, о, ч, ь, ы, ъ), после которой ставится скобка.

Для дальнейшей детализации перечислений необходимо использовать арабские цифры, после которых ставится круглая скобка. Запись в этом случае производится с дополнительным абзацным отступом, подчеркивающим иерархию:

Пример – 1, 2, 3 и т. д.

Номер подраздела или пункта включает номер раздела и порядковый номер подраздела или пункта, разделенные точкой.

Пример – 1.1, 1.2, 1.3 и т. д.

Номер подпункта включает номер раздела, подраздела, пункта и порядковый номер подпункта, разделенные точкой.

Пример – 1.1.1.1, 1.1.1.2, 1.1.1.3 и т. д.

Если раздел или подраздел имеет только один пункт или пункт имеет один подпункт, то нумеровать его не следует.

Разделы, подразделы должны иметь заголовки. Пункты, как правило, заголовков не имеют. Заголовки должны четко и кратко отражать содержание разделов, подразделов.

Заголовки разделов и подразделов следует печатать с абзацного отступа с прописной буквы без точки в конце, не подчеркивая.

Если заголовок состоит из двух предложений, их разделяют точкой.

### Нумерация страниц

Страницы пояснительной записки следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту отчета. Номер страницы проставляют в центре нижней части листа без точки.

Титульный лист включают в общую нумерацию страниц отчета. Номер страницы на титульном листе не проставляют.

Иллюстрации и таблицы, расположенные на отдельных листах, включают в общую нумерацию страниц отчета.

Иллюстрации и таблицы на листе формата А3 учитывают как одну страницу.

### Нумерация разделов, подразделов, пунктов, подпунктов пояснительной записки

Разделы должны иметь порядковые номера в пределах всего документа, обозначенные арабскими цифрами без точки и записанные с абзацного отступа. Подразделы должны иметь нумерацию в пределах каждого раздела. Номер подраздела состоит из номеров раздела и подраздела,

разделенных точкой. В конце номера подраздела точка не ставится. Разделы, как и подразделы, могут состоять из одного или нескольких пунктов.

Если документ не имеет подразделов, то нумерация пунктов в нем должна быть в пределах каждого раздела и номер пункта должен состоять из номеров раздела и пункта, разделенных точкой. В конце номера пункта точка не ставится.

#### Пример

##### *1 Описание систем-аналогов*

- 1.1
  - 1.2
  - 1.3
- } Нумерация пунктов первого раздела документа

##### *2 Техническое задание*

- 2.1
  - 2.2
  - 2.3
- } Нумерация пунктов второго раздела документа

Если документ имеет подразделы, то нумерация пунктов должна быть в пределах подраздела и номер пункта должен состоять из номеров раздела, подраздела и пункта, разделенных точками.

#### Пример

##### *3 Методика тестирования*

##### *3.1 Описание набора тестов*

- 3.1.1
  - 3.1.2
  - 3.1.3
- } Нумерация пунктов первого подраздела третьего раздела документа

##### *3.2 Подготовка к тестированию*

- 3.2.1
  - 3.2.2
  - 3.2.3
- } Нумерация пунктов второго подраздела третьего раздела документа

Если раздел состоит из одного подраздела, то подраздел не нумеруется. Если подраздел состоит из одного пункта, то пункт не нумеруется. Наличие одного подраздела в разделе эквивалентно фактическому отсутствию других.

Если текст отчета подразделяется только на пункты, то они нумеруются порядковыми номерами в пределах всего отчета.

Пункты, при необходимости, могут быть разбиты на подпункты, которые должны иметь порядковую нумерацию в пределах каждого пункта, например: 4.2.1.1, 4.2.1.2, 4.2.1.3 и т. д.

#### Перечисления

Внутри пунктов или подпунктов могут быть приведены перечисления.

Перед каждым перечислением следует ставить дефис или, при необходимости ссылки в тексте документа на одно из перечислений, строчную букву (за исключением ё, з, о, г, ь, й, ы, ъ), после которой ставится скобка.

Для дальнейшей детализации перечислений необходимо использовать арабские цифры, после которых ставится скобка, а запись производится с абзацного отступа, как показано в примере.

Пример

а) \_\_\_\_\_  
б) \_\_\_\_\_  
1) \_\_\_\_\_  
2) \_\_\_\_\_  
в) \_\_\_\_\_

Если отчет состоит из двух и более частей, каждая часть должна иметь свой порядковый номер. Номер каждой части следует проставлять арабскими цифрами на титульном листе под указанием вида отчета, например «Часть 2».

Каждый структурный элемент ПЗ (каждую главу) следует начинать с нового листа (страницы).

Нумерация страниц отчета и приложений, входящих в состав ПЗ, должна быть сквозная.

### Иллюстрации

Иллюстрации (чертежи, графики, схемы, компьютерные распечатки, диаграммы, фотоснимки) следует располагать в ПЗ непосредственно после текста, в котором они упоминаются впервые, или на следующей странице.

Иллюстрации могут быть в компьютерном исполнении, в том числе и цветные.

На все иллюстрации должны быть даны ссылки в ПЗ.

Чертежи, графики, диаграммы, схемы, иллюстрации, приводимые в ПЗ, должны соответствовать требованиям государственных стандартов Единой системы конструкторской документации (ЕСКД).

Допускается выполнение чертежей, графиков, диаграмм, схем посредством использования компьютерной печати.

Фотоснимки размером меньше формата А4 должны быть наклеены на стандартные листы белой бумаги.

Иллюстрации, за исключением иллюстрации приложений, следует нумеровать арабскими цифрами сквозной нумерацией.

Если рисунок один, то он обозначается «Рисунок 1». Слово «рисунок» и его наименование располагают посередине строки.

Допускается нумеровать иллюстрации в пределах раздела. В этом случае номер иллюстрации состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой. Например, Рисунок 1.1.

Иллюстрации при необходимости могут иметь наименование и пояснительные данные (подрисуночный текст). Слово «Рисунок» и наименование помещают после пояснительных данных и располагают следующим образом: Рисунок 1 — Детали прибора.

Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Например, Рисунок А.3.



При ссылках на иллюстрации следует писать «... в соответствии с рисунком 2» при сквозной нумерации и «... в соответствии с рисунком 1.2» при нумерации в пределах раздела.

Пример оформления рисунка в ПЗ приведен в прил. Ж.

## Таблицы

Таблицы применяют для лучшей наглядности и удобства сравнения показателей. Название таблицы, при его наличии, должно отражать ее содержание, быть точным, кратким. Название таблицы следует помещать над таблицей слева, без абзацного отступа в одну строку с ее номером через тире.

При переносе части таблицы название помещают только над первой частью таблицы, нижнюю горизонтальную черту, ограничивающую таблицу, не проводят.

Таблицу следует располагать в отчете непосредственно после текста, в котором она упоминается впервые, или на следующей странице.

На все таблицы должны быть ссылки в отчете. При ссылке следует писать слово «таблица» с указанием ее номера.

Таблицу с большим количеством строк допускается переносить на другой лист (страницу). При переносе части таблицы на другой лист (страницу) слово «Таблица» и номер ее указывают над первой частью таблицы, над другими частями пишут слово «Продолжение» и указывают номер таблицы, например: «Продолжение таблицы 1». При переносе таблицы на другой лист (страницу) заголовок помещают только над ее первой частью.

Таблицу с большим количеством граф допускается делить на части и помещать одну часть под другой в пределах одной страницы. Если строки и графы таблицы выходят за формат страницы, то в первом случае в каждой части таблицы повторяется головка, во втором случае — боковик.

Если повторяющийся в разных строках графы текст состоит из одного слова, то его после первого написания допускается заменять кавычками; если из двух и более слов, то при первом повторении его заменяют словами «То же», а далее — кавычками. Ставить кавычки вместо повторяющихся цифр, марок, знаков, математических и химических символов не допускается. Если цифровые или иные данные в какой-либо строке таблицы не приводят, то в ней ставят прочерк.

Цифровой материал, как правило, оформляют в виде таблиц. Пример оформления таблицы приведен на рис. 1.



Рисунок 1 – оформление таблицы

Таблицы, за исключением таблиц приложений, следует нумеровать арабскими цифрами сквозной нумерацией.

Допускается нумеровать таблицы в пределах раздела. В этом случае номер таблицы состоит из номера раздела и порядкового номера таблицы, разделенных точкой.

Таблицы каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения.

Если в документе одна таблица, то она должна быть обозначена «Таблица 1» или «Таблица В.1», если она приведена в приложении В.

Заголовки граф и строк таблицы следует писать с прописной буквы в единственном числе, а подзаголовки граф — со строчной буквы, если они составляют одно предложение с заголовком, или с прописной буквы, если они имеют самостоятельное значение. В конце заголовков и подзаголовков таблиц точки не ставят.

Таблицы слева, справа и снизу, как правило, ограничивают линиями. Допускается применять размер шрифта в таблице меньший, чем в тексте.

Разделять заголовки и подзаголовки боковика и граф диагональными линиями не допускается.

Горизонтальные и вертикальные линии, разграничивающие строки таблицы, допускается не проводить, если их отсутствие не затрудняет пользование таблицей.

Заголовки граф, как правило, записывают параллельно строкам таблицы. При необходимости допускается перпендикулярное расположение заголовков граф.

Головка таблицы должна быть отделена линией от остальной части таблицы.

Пример оформления таблиц приведен в прил. И.

### Примечания

Слово «Примечание» следует печатать с прописной буквы с абзаца и не подчеркивать.

Примечания приводят в документах, если необходимы пояснения или справочные данные к содержанию текста, таблиц или графического материала. Примечания не должны содержать требований.

Примечания следует помещать непосредственно после текстового, графического материала или в таблице, к которым относятся эти примечания. Если примечание одно, то после слова «Примечание» ставится тире и примечание печатается с прописной буквы. Одно примечание не нумеруют. Несколько примечаний нумеруют по порядку арабскими цифрами без проставления точки. Примечание к таблице помещают в конце таблицы над линией, обозначающей окончание таблицы.

Пример

**Примечание -** \_\_\_\_\_

Несколько примечаний нумеруются по порядку арабскими цифрами.

Пример

Примечания

1 \_\_\_\_\_

2 \_\_\_\_\_

3 \_\_\_\_\_

## Формулы и уравнения

Уравнения и формулы следует выделять из текста в отдельную строку. Выше и ниже каждой формулы или уравнения должно быть оставлено не менее одной свободной строки. Если уравнение не умещается в одну строку, то оно должно быть перенесено после знака равенства (=), или после знаков плюс (+), минус (-), умножения (x), деления (/), или других математических знаков, причем знак в начале следующей строки повторяют. При переносе формулы на знаке, символизирующем операцию умножения, применяют знак «X».

Пояснение значений символов и числовых коэффициентов следует приводить непосредственно под формулой в той же последовательности, в которой они даны в формуле.

Формулы в отчете следует нумеровать порядковой нумерацией в пределах всего отчета арабскими цифрами в круглых скобках в крайнем правом положении на строке.

Пример

$$N = 100 \times \frac{E_n \times S_n \times Z \cdot K_z}{KПД \times \Phi_{св} \times K_m} . \quad (3)$$

Одну формулу обозначают (1).

Формулы, помещаемые в приложениях, должны нумероваться отдельной нумерацией арабскими цифрами в пределах каждого приложения с добавлением перед каждой цифрой обозначения приложения, например формула (В.1).

Ссылки в тексте на порядковые номера формул дают в скобках. Пример –... в формуле (1).

Допускается нумерация формул в пределах раздела. В этом случае номер формулы состоит из номера раздела и порядкового номера формулы, разделенных точкой, например (3.1).

Порядок изложения в ПЗ математических уравнений такой же, как и формул.

В ПЗ допускается выполнение формул и уравнений рукописным способом черными чернилами.

## Ссылки

В ПЗ допускаются ссылки на данный документ, стандарты, технические условия и другие документы при условии, что они полностью и однозначно определяют соответствующие требования и не вызывают затруднений в пользовании документом.

Ссылаться следует на документ в целом или его разделы и приложения. Ссылки на подразделы, пункты, таблицы и иллюстрации не допускаются, за исключением подразделов, пунктов, таблиц и иллюстраций данного документа.

При ссылках на стандарты и технические условия указывают только их обозначение, при этом допускается не указывать год их утверждения при условии полного описания стандарта в списке использованных источников в соответствии с ГОСТ 7.1-2003.

Ссылки на использованные источники следует приводить в квадратных скобках.

## Приложения

Приложение оформляют как продолжение данного документа на последующих его листах или выпускают в виде самостоятельного документа.

В тексте документа на все приложения должны быть даны ссылки. Приложения располагают в порядке ссылок на них в тексте документа, за исключением справочного приложения «Библиография», которое располагают последним.

Каждое приложение следует начинать с новой страницы с указанием наверху посередине страницы слова «Приложение», его обозначения и степени.

Приложение должно иметь заголовок, который записывают симметрично относительно текста с прописной буквы отдельной строкой.

Приложения обозначают заглавными буквами русского алфавита, начиная с А, за исключением букв Ё, З, Й, О, Ч, Ь, Ы, Ъ. После слова «Приложение» следует буква, обозначающая его последовательность.

Допускается обозначение приложений буквами латинского алфавита, за исключением букв I и O.

В случае полного использования букв русского и латинского алфавитов допускается обозначать приложения арабскими цифрами.

Если в документе одно приложение, оно обозначается «Приложение А».

Текст каждого приложения при необходимости может быть разделен на разделы, подразделы, пункты, подпункты, которые нумеруют в пределах каждого приложения. Перед номером ставится обозначение этого приложения.

Приложения должны иметь общую с остальной частью документа сквозную нумерацию страниц.

При необходимости такое приложение может иметь пункт «Содержание».

Приложениям или частям, выпущенным в виде самостоятельного документа, обозначение присваивают как части документа с указанием в коде документа ее порядкового номера.

# Лабораторная работа №1

## Цель и задачи

Написание простого консольного приложения для вычисления ряда логических (математических) задач и для ознакомления с языком программирования.

## Описание задания

Написание echo-приложение с анализом сообщений.

Необходимо создать консольное приложение позволяющее принимать сообщение с консоли обрабатывать его, то есть подсчитывать количество символов и слов в сообщении и выводить всю информацию на консоли.

### Детализация

При загрузке приложения, пользователю дается информация о возможностях приложения, например:

«Данное приложение выполняет функцию echo-приложения, все введенные символы в консоли будут обрабатываться приложением и выводиться на консоли с указанием количества символов в строке. Для выхода из приложения наберите команду: exit или 0.»

Приложение должно получать информацию с консоли анализировать ее, то есть подсчитывать количество символов и количество отдельных слов в строке.

После чего информация должна быть выведена на консоль:

- само сообщение;
- через пробел и любой разделяющий символ количество символов;
- дальше через пробел и любой разделяющий символ количество слов.

Для выхода из приложения необходимо предусмотреть обработку ключевых слов (символов). То есть если из консоли придет сообщение exit или 0, приложение должно прекратить свою работу.

### Пример работы:

Пользователь вводит фразу: «Уральский федеральный университет»

В ответ должен получить: «Уральский федеральный университет – 33 – 3»

## Написать приложение для решения ряда задач

Задачи которые необходимо реализовать в приложении:

- а. Определение является ли заданный год високосным с помощью класса DateTime. Пользователь вводит год интересующий его, программа должна определить является ли она високосным (366 дней).
- б. Анализ генерации случайных чисел. Пользователь вводит 3 параметра: число определяющее сколько раз должно сгенерироваться число, минимальная граница генерации чисел и максимальная граница генерации чисел. После чего должно сгенерироваться n-ое количество чисел в заданных границах. Далее необходимо вычислить сколько раз было сгенерировано каждое из чисел и вывести эту информацию пользователю.  
Пример: пользователь вводит количество чисел равное 100, минимальную границу 1, максимальную границу 2. В результате будет сгенерировано 100 чисел в пределах от 1 до 2, нужно подсчитать сколько сгенерировалось цифр «1» и «2» по отдельности. В идеальном случае результат будет 50 значений «1» и 50 значений «2».
- в. Расчет разницы между заданной датой и текущей используя класс DateTime.

- d. Вычисление площади треугольника по заданным: стороне и высоте, трем сторонам, двум сторонам и углу между ними. Пользователь выбирает, какую именно хочет решать задачу сейчас, далее вводит необходимую информацию. Используя пакет System.Math.
- e. Дана последовательность натуральных чисел, разделенных пробелами, в виде строки. Необходимо вычислить их сумму.
- f. Выполнение индивидуального задания.

### Технические требования

Каждая задача должна быть оформлена в отдельный класс, содержать свой конструктор с инициализацией входных данных, а также функцию вычисления (возврата) результата.

При загрузке приложения должна выводиться информация с пунктами «меню». То есть если вы хотите рассчитать арифметическое выражение то нажимаете соответствующую пункту цифру, например: 5. После чего вводите необходимую информацию для вычисления. Осуществляется с помощью функции switch – case.

Так же должен быть предусмотрен механизм выхода из приложения, например на команду exit.

После выполнения вычисления, и вывода результата на консоль, происходит возврат в главное меню. То есть необходимо сделать бесконечный цикл, до того момента пока пользователь не пожелает выйти из него прописав соответствующую команду.

## Варианты индивидуальных заданий

Данные задания были взяты из другого курса по программированию.

Вариант №1	
№	Описание задачи
1	Вычислить значение функции. $F(x) = \begin{cases} \cos 3x + \sin 5x, & \text{если } x > 0 \\ \sin 2x, & \text{иначе} \end{cases}$
2	Написать программу, которая на ввод знака препинания выводит его название. Например, на ввод точки выводится сообщение: «Это – точка».
3	Задан массив S, размерностью n, содержащий большое количество нулевых элементов. Найти длину самой длинной последовательности подряд идущих элементов, равных 0. Вывести также индекс начала этой последовательности.
4	Написать программу, которая определяет номер строки прямоугольной матрицы случайных действительных чисел A[N][M], сумма элементов которой максимальна. Вывести на экран эту строку отдельно.
5	Подсчитать общее количество цифр и знаков '+', '-', и '*', входящих во вводимую с клавиатуры строку.

Вариант №2	
№	Описание задачи
1	Вычислить значение функции. $F(x) = \begin{cases} x^2, & \text{если } x - \text{четно} \\  x ^3, & \text{иначе} \end{cases}$
2	Написать программу, которая на ввод времени суток выводит соответствующее пожелание <u>доброе у</u> тра, <u>добр</u> ого дня, <u>добр</u> ого вечера или <u>спокойной</u> ночи.
3	В массиве натуральных случайных чисел A[N] осуществите сдвиг на K < N элементов вправо без использования дополнительных массивов. Первые K позиций при этом становятся нулевыми, последние K позиций теряются.
4	Написать программу, проверяющую, является ли введенная с клавиатуры квадратная матрица A[N][N] магическим квадратом. Матрица является магическим квадратом, если сумма элементов в каждой строке, сумма элементов в каждом столбце и сумма элементов по диагоналям одна и та же.
5	Вводится текст из цифр и строчных латинских букв, за которым следует точка. Определить, каких букв – гласных (a, e, i, o, u) или согласных – больше в этом тексте.

Вариант №3	
№	Описание задачи
1	Вычислить значение функции. $F(x) = \begin{cases}  x ^3 + x^2, & \text{если } x > 0 \\  x - 10 , & \text{иначе} \end{cases}$
2	По введенной дате (число, месяц, год) вычислить, сколько дней осталось до конца года.
3	Дан массив действительных случайных чисел a размером n=100. Вывести на экран те элементы массива, индексы которых являются степенями двойки 2, 4, 8, 16
4	В заданном массиве случайных целых чисел A[N][M] найдите суммы элементов в каждой строке. Сформируйте одномерный массив S[N], содержащий полученные суммы.
5	Вводится текст, за которым следует точка. В алфавитном порядке напечатать (по разу) все строчные русские буквы, входящие в этот текст.

Вариант №4	
№	Описание задачи
1	<p>Вычислить значение функции:</p> $F(x, y) = \begin{cases}  x  +  y , & \text{если } x > y \\  y  -  x , & \text{иначе} \end{cases}$
2	Ввести две даты. Вывести сообщение: «Неправильно», если вторая дата предшествует первой, и - сообщение «Правильно» в противном случае
3	Дан массив А из n целых случайных чисел. Ввести число М. Выбрать из массива А все числа, меньшие М.
4	В матрице действительных чисел D размером MxN поменять местами строки с заданными номерами.
5	Зашифровать введенную с клавиатуры строку заменой исходных символов на символы с кодом, большим на три единицы. Провести дешифровку

Вариант №5	
№	Описание задачи
1	<p>Вычислить значение функции:</p> $F(x) = \begin{cases} tg^2 x + ctg^3 x, & \text{если } x > 0 \\ tg 2x, & \text{иначе} \end{cases}$
2	Ввести три числа <i>a</i> , <i>b</i> , <i>c</i> . Упорядочить их так, чтобы переменной <i>a</i> соответствовало наименьшее число, <i>b</i> - среднее, <i>c</i> – максимальное.
3	Дан массив F из n целых случайных чисел. Выбрать из него в массив W все четные элементы, а в массив S – все нечетные.
4	Дана матрица случайных действительных чисел D, размерностью NxM. В каждой строке найти максимальный элемент. Вывести на экран минимальный из всех найденных максимальных элементов с указанием его индексов.
5	Разработать программу, проверяющую, является ли введенное с клавиатуры слово палиндромом (читается одинаково в обе стороны).

Вариант №6	
№	Описание задачи
1	<p>Вычислить значение функции:</p> $F(x) = \begin{cases} \cos^2 x + \sin^3 x, & \text{если } x > 0 \\ \sin 2x, & \text{иначе} \end{cases}$
2	Даны длины трех отрезков. Определить, являются ли они сторонами треугольника; если да, то вычислить площадь данного треугольника.
3	По заданному символьному массиву S сформировать массив двоичных элементов В. Элементу массива В <sub>і</sub> присвоить значение 1, если S <sub>і</sub> является цифрой, и 0 - в противном случае.
4	В символьной матрице А размером MxN элементов выбрать те элементы, сумма индексов которых кратна 5
5	Во введенном с клавиатуры тексте подсчитать сумму встречающихся в нем целых чисел.



Вариант №7	
№	Описание задачи
1	Вычислить значение функции: $F(x) = \begin{cases} \sqrt{x}, & \text{если } x \geq 0 \\ \sqrt{ x }, & \text{иначе} \end{cases}$
2	Дано натуральное число $n$ , определяющее возраст человека в годах. Для этого числа дать наименование «год», «года» или «лет». Вывести на экран сообщение вида: «Возраст человека $n$ лет (год, года)».
3	Заданы массивы случайных действительных чисел $A[N]$ и $B[N]$ . Сформируйте массив $S[N]$ , где $S_i = A_i + B_i$ .
4	Дана символьная матрица $B$ размером $M \times N$ . Составить программу, осуществляющую перестановку в каждой строке матрицы так, чтобы первый элемент строки поменялся с последним
5	Дан текст. Найти максимальную длину слова

Вариант №8	
№	Описание задачи
1	Вычислить значение функции, при условии, что $x$ – целое: $F(x) = \begin{cases} x^2, & \text{если } x - \text{четно} \\  x ^3, & \text{иначе} \end{cases}$
2	Составить программу пересчёта денег в рублях в валюту по выбору (например: доллар, марка, франк). Исходные данные: курсы валют (не меньше трёх, их можно задать в виде констант), сумма в рублях (вводить в клавиатуры).
3	Задан массив, состоящий из целых положительных и отрицательных чисел. Разбить его на два массива, в одном из которых только положительные числа, в другом - отрицательные.
4	Дана целочисленная матрица размером $N \times N$ , все элементы которой ненулевые. Найти скалярное произведение строки с заданным номером $k$ , на столбец с заданным номером $d$ .
5	Заданы фамилия, имя и отчество, разделенные пробелом. Напечатайте фамилию и инициалы.

Вариант №9	
№	Описание задачи
1	Вычислить значение функции: $F(x) = \begin{cases} tg\ 2x + ctg\ 2x, & \text{если } x > 0 \\ tg^2 x, & \text{иначе} \end{cases}$
2	Вычислить значение функции: $F(x, y) = \begin{cases} \ln(x + y^2), & \text{если } x \text{ и } y - \text{положительные} \\ \ln( x  +  y ), & \text{если } x \text{ и } y - \text{отрицательные} \\ 0, & \text{иначе} \end{cases}$
3	Даны действительные числа $a_1, \dots, a_n$ , $p$ , натуральное число $k$ ( $a_1 \leq a_2 \leq \dots \leq a_k \leq \dots \leq a_n$ , $k \leq n$ ). Удалить из $a_1, \dots, a_n$ элемент с номером $k$ (т.е. $a_k$ ) и вставить элемент, равный $p$ , так, чтобы не нарушилась упорядоченность
4	В данной действительной матрице размером $p \times m$ найти наибольший по модулю элемент. Получить матрицу размером $N \times (M-1)$ путем выбрасывания из исходной матрицы столбца, содержащего элемент с найденным значением.
5	Дан текст. Каждое встречающееся слово child заменить на слово children.

Вариант №10	
№	Описание задачи
1	<p>Вычислить значение функции, при условии, что <math>x</math> - целое:</p> $F(x) = \begin{cases} \sqrt{x}, & \text{если } x - \text{четно} \\ \sqrt[3]{x}, & \text{иначе} \end{cases}$
2	Вывести на экран сообщение в зависимости от введенного значения оценки (по десятибалльной системе), например: от 1 до 2 плохо, от 3 до 5 удовлетворительно, и т.д., иначе – неправильный ввод данных.
3	Описать символьный массив D, размерностью 30 элементов, заполнять массив до тех пор, пока не встретится символ ?. Вывести все элементы в три строки.
4	Найти максимальные элементы каждого столбца действительной матрицы K размерности NxN и поместить их на главную диагональ, а диагональные элементы – на место соответствующего элемента столбца.
5	Напишите программу, которая вычисляет целочисленные арифметические выражения любой длины, не содержащие скобок. Учесть все арифметические операции: сложение, вычитание, умножение и целочисленное деление.

Вариант №11	
№	Описание задачи
1	<p>Вычислить значение функции: <math>F(x) = \begin{cases} e^{\sin x}, &amp; \text{если } x \geq 0 \\ e^{\cos x}, &amp; \text{иначе} \end{cases}</math></p>
2	Составить программу, которая определит план вашей жизни на неделю. Например, в субботу – еду в баню, воскресенье – еду в гости и т.д. Программа запрашивает ввод дня недели и выводит на экран ваши действия в зависимости от составленного расписания.
3	Задан массив A[N], содержащий большое количество нулевых элементов. Группы элементов, состоящие из нечетного количества нулей, замените на один нулевой элемент, а из четного - на два.
4	Даны целые числа $a_1, \dots, a_n$ , целая матрица порядка NxM ( $m \geq 6$ ). Получить матрицу размера $N \times (M+1)$ , вставив в исходную матрицу столбец из чисел $a_1, \dots, a_n$ в середину.
5	Одно слово является анаграммой другого, если оно может быть получено перестановкой всех букв данного (КОЛОС = СОКОЛ). Составьте программу, которая находит все анаграммы в данном наборе слов и подсчитывает их число.

Вариант №12	
№	Описание задачи
1	<p>Вычислить значение функции: <math>F(x, y) = \begin{cases} \sqrt{ x + y }, &amp; \text{если } \cos x &gt; \sin y \\ \sqrt[3]{ x + y }, &amp; \text{иначе} \end{cases}</math></p>
2	Составить программу, которая в зависимости от введенного номера месяца выводит название времени года, к которому он относится.
3	В символьном массиве A[N] осуществите циклический сдвиг на $K < N$ элементов вправо. Последние K элементов при этом перемещаются на первые K позиций.
4	Дана квадратная матрица действительных чисел D размерности NxN. Создать новую матрицу C размерности NxN, на главной диагонали которой разместить элементы заданного столбца матрицы D, а остальные элементы сделать равными 0
5	Дан текст. Напечатать все слова в алфавитном порядке

Вариант №13	
№	Описание задачи
1	Определить, является ли число $M$ делителем числа $N$ .
2	Заданы три положительных числа $a$ , $b$ и $c$ . Определить, являются ли они последовательно стоящими членами арифметической или геометрической прогрессии. Если являются, то вычислить разность или знаменатель прогрессии.
3	Дан массив случайных целых чисел $a$ размером $n$ , $4 < a_i < 15$ . Посчитать, сколько различных чисел в этом массиве.
4	Найти минимальный элемент в матрице случайных целых чисел $A$ размерности $N \times M$ и вычесть его значение из элементов строки, где он находится.
5	Определить и вывести на экран номера позиций и количество повторений запрашиваемого символа в строке, введенной с клавиатуры.

Вариант №14	
№	Описание задачи
1	Даны три вещественных числа $x$ , $y$ , $z$ . Вычислить значения выражений $x+y+z$ и $xuz$ ; определить максимальное из них.
2	Заданы три стороны треугольника $a$ , $b$ и $c$ . определить является ли этот треугольник прямоугольным и какая сторона является гипотенузой.
3	Присвоить переменной $A$ значение 0, если хотя бы один элемент символьного массива $B$ из $N$ элементов равен некоторому элементу символьного массива $C$ из $M$ элементов, в противном случае присвоить $A$ единицу.
4	Подсчитать количество нулевых строк в целочисленной матрице $A$ размерности $M \times N$ , в случае отсутствия таких строк выдать диагностику соответствующего содержания.
5	Даны два слова. Составьте программу, определяющую можно или нет из букв слова $A$ составить слово $B$

Вариант №15	
№	Описание задачи
1	Определить, является ли введенное число числом Амстронга. Числом Амстронга называется число, равное сумме своих цифр в степени их количества. Например: $153 = 1^3 + 5^3 + 3^3$
2	Даны три числа $a$ , $b$ , $c$ . Определить, существует ли треугольник со сторонами $a$ , $b$ , $c$
3	Выяснить, какие из элементов символьного массива $B$ из $n$ элементов входят в него только один раз.
4	В символьной матрице $B$ размером $M \times N$ поменять местами столбцы с заданными номерами.
5	Дан текст. Найти максимальную длину слова.

Вариант №16	
№	Описание задачи
1	Даны две точки: $A(x_1, y_1)$ и $B(x_2, y_2)$ . Составить алгоритм, определяющий, которая из точек находится ближе к началу координат.
2	В каждый подарочный набор входят 1 ручка, 2 линейки, 4 тетрадки. Имеется $a$ линеек, $b$ тетрадей, $c$ ручек. Сколько получится наборов?
3	Дано некоторое число $A$ . Найти в целочисленном массиве $B$ из $N$ элементов ближайший по значению меньший числа $A$ элемент.
4	Дана символьная матрица $A$ размерности $M \times N$ . Построить массив $C$ , элементы которого соответственно равны элементам матрицы $A$ , сумма индексов которых кратна 3.
5	Определить верно ли, что в предложении следующее слово начинается на ту же букву, которой оканчивается предыдущее слово. Если нет вывести номер слова, для которого это не так

Вариант №17	
№	Описание задачи
1	Заданы два натуральных числа. Определить является ли среднее арифметическое этих чисел целым числом.
2	Даны три числа. Возвести в квадрат те из них, значения которых неотрицательны, и в четвертую степень – отрицательные.
3	Дан массив действительных случайных чисел $a$ , размером $n=100$ . Вывести на экран те элементы массива, индексы которых являются полными квадратами.
4	Дана целочисленная квадратная матрица. Выдать на экран разность сумм элементов на главной и побочной диагоналях.
5	Подсчитать общее количество цифр и знаков '+', '-', и '*', входящих во вводимую с клавиатуры строку.

Вариант №18	
№	Описание задачи
1	Два отрезка заданы координатами своих концов. Определить, во сколько раз наибольший из них больше наименьшего.
2	Даны целые числа $m, n, k$ . Если числа не равны, то заменить каждое из них одним и тем же числом, равным большему из исходных, а если равны, то заменить число нулями.
3	Найти 2 наименьших положительных элемента в массиве случайных действительных чисел $A$ из $n$ элементов.
4	Определить есть ли в целочисленной матрице $B$ размерности $N \times M$ строки, максимальный элемент в которых стоит первым в строке. Если есть, то вывести эти строки на экран, нет – вывести сообщение.
5	Дана произвольная строка. Вывести список тех строчных латинских букв, которые в ней встречаются.

Вариант №19	
№	Описание задачи
1	Даны три вещественных числа $x, y, z$ . Вычислить значения выражений $x + y + z$ и $xuz$ ; определить максимальное из них.
2	<p>Вычислить значение функции:</p> $F(x) = \begin{cases} x^2 - 3 - \sqrt[3]{\pi - x}, & \text{если } x < 0; \\ (x^2 + 3)^2 - \sqrt{0,5\pi + x}, & \text{если } 0 \leq x < 1; \\ x(x^2 + 3) + \ln(\pi + x), & \text{если } x \geq 1 \end{cases}$
3	Имеются два одномерных массива целых чисел размером $n$ . Создать из них один одномерный массив, в котором сначала идут отрицательные элементы, затем нулевые и затем положительные.
4	Дан двумерный массив. Удалить строку и столбец, на пересечении которых расположен наименьший по модулю элемент массива.
5	<p>Дано целое положительное число в диапазоне от 0 до 9999. Требуется написать программу для преобразования числа в словесное выражение. Например, число 61912 должно быть преобразовано в строку "шестьдесят одна тысяча девятьсот двенадцать".</p> <p>Число вводится с клавиатуры в виде строки</p>

Вариант №20	
№	Описание задачи
1	Услуги телефонной сети оплачиваются по следующему правилу: за разговоры до $A$ минут – $B$ рублей, а разговоры сверхустановленной нормы оплачиваются из расчета $C$ рублей за минуту. Составить алгоритм программы, вычисляющей плату за пользование телефоном для введенного времени разговоров за месяц.
2	<p>Вычислить значение функции:</p> $F(x, y) = \begin{cases} \sqrt{ x ^3 +  y ^3}, & \text{если } x = y \\ \sqrt[3]{ x + y }, & \text{если } x < y \\ 0, & \text{если } x > y \end{cases}$
3	<p>Описать символьный массив <math>G</math>, размерностью 15 элементов, заполнять массив с клавиатуры. Вывести все элементы в виде:</p> <p style="text-align: center;">* ** *** **** *****</p>
4	Дан двумерный массив из чётного числа строк. Поменять местами первую строку со второй, третью с четвёртой и т.д.
5	Определить и вывести на экран номера позиций и количество повторений запрашиваемого символа в строке, введенной с клавиатуры.

## Теория и примеры реализаций

### Создание консольного приложения

Для создания консольного приложения, необходимо загрузить среду разработки Visual Studio. Перейти в пункт меню «Файл», дальше «Создать» -> «Проект». Выбираем «Консольное приложение», задаем имя проекта -> «ОК» (см. рисунок 1.1).

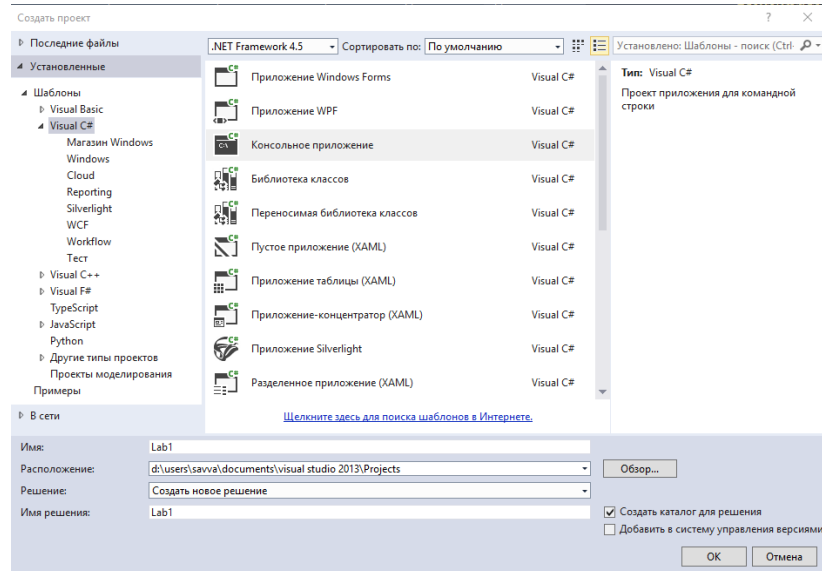


Рисунок 1.1 – Форма создания консольного приложения

Автоматический будет создан класс Program, у которого будет метод Main главная точка входа в приложение.

Для работы с консолью нам понадобятся два основных метода – это управление входящим и исходящим потоком консоли:

- Console.ReadLine() – считывание входного потока в виде строки, возвращает тип String;
- Console.WriteLine(String s) – вывод информации на консоль, в данный момент входным параметром является строка, но так же это может быть и int, и double и др. типы данных.

Пример программы на рисунке 1.2.

```
static void Main(string[] args)
{
    //бесконечный цикл
    while(true)
    {
        //вывод информации на консоль
        Console.WriteLine("Если хотите закрыть приложение введите \"exit\"");
        //ожидание ввода информации в консоли и присвоение полученных данных переменной
        String in_m = Console.ReadLine();
        //проверка на введенные значения, если было введено слово "exit" будет выполнен выход
        if (in_m.Equals("exit"))
        {
            //выход из цикла
            break;
        }
    }
}
```

Рисунок 1.2 – Пример кода «первое консольное приложение»

Пример создания отдельного класса

Пример создания отдельного класса решающий задачу: вычисления периметра окружности по заданному радиусу (см. рисунок 1.3).

```
ссылка 4
class Circle
{
    private double R;
    //конструктор с входным параметром типа double
    ссылка 0
    public Circle(double R)
    {
        this.R = R;
    }
    //конструктор с входным параметром типа string
    ссылка 1
    public Circle(String R)
    {
        /*обработчик ошибки, если входной параметр
        не возможно будет конвертировать в тип данных
        double возникнет исключение в котором параметру
        R будет присвоено значение 0*/
        try
        {
            /*конвертирование из типа данных String в int*/
            this.R = Convert.ToDouble(R);
        }
        catch (Exception e)
        {
            this.R = 0;
        }
    }
}

ссылка 1
public String getResult()
{
    //формула вычисления периметра окружности по радиусу
    double result = 2 * Math.PI * this.R;
    return result.ToString();
}
}
```

Рисунок 1.3 – Листинг класса вычисления периметра окружности

Вызывается данный класс следующим образом (см. рисунок 1.4).

```
ссылка 0
class Program
{
    ссылка 0
    static void Main(string[] args)
    {
        Console.WriteLine("Введите радиус окружности: ");
        String R = Console.ReadLine();

        Circle c = new Circle(R);
        Console.WriteLine("Результат: {0}", c.getResult());

        Console.ReadKey();
    }
}
```

Рисунок 1.4 – Листинг вызова класса Circle

## Работа в Debug-режиме

Debug-режим служит для программиста при отладке кода, он позволяет ставить точки прерывания (точка останова) и наблюдать за полным выполнением программного кода и за изменениями параметров.

Точку прерывания (breakpoint) можно поставить на любую строку кода, которая в действительности исполняется. Один из способов поставить breakpoint – это выбрать строку с которой необходимо начать отслеживание кода и нажать на поле слева. После этого на полях должен загореться красный кругляшек (см. рисунок 1.5).

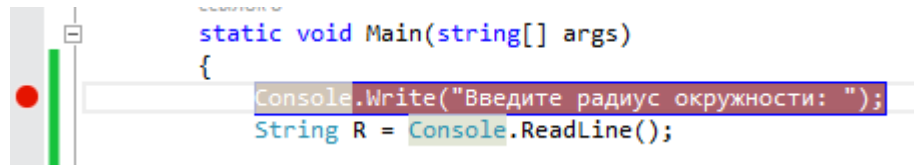


Рисунок 1.5 – Установка breakpoint

Для снятия breakpoint просто следует нажать на то же самое место, и breakpoint пропадет.

Ту же самую операцию добавления/удаления breakpoint можно делать с помощью кнопки <F9>.

### Слежение

После того как breakpoint поставлен в нужном месте, нужно запустить приложение, и когда выполнение кода задействует выделенную строчку, программа на ней остановиться.

После срабатывания breakpoint у разработчика появляется возможность посмотреть значение всех созданных переменных. Увидеть их можно внизу окна среды разработки, вкладки: видимые, локальные и контрольные значения (см. рисунок 1.6).

Вкладка локальные позволяет просматривать значения переменных, к которым получается доступ в методе, выполняемом в текущий момент.

Вкладка видимые позволяет просматривать значения нескольких последних переменных, к которым осуществлялся доступ в процессе выполнения программы.

Вкладка контрольные значения позволяет просматривать значения любых интересующих переменных за счет явного указания их имен непосредственно в окне.

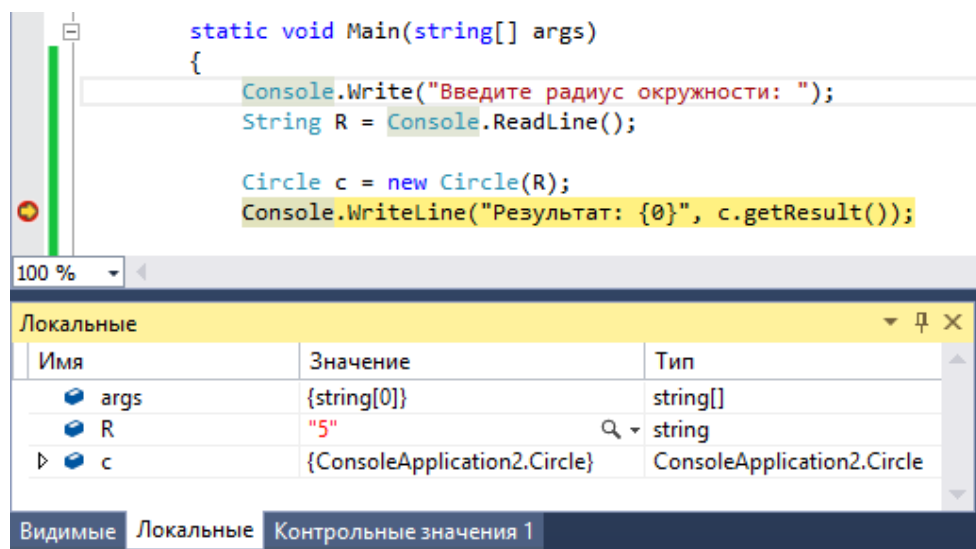




Рисунок 1.6 – Просмотр параметров в debug-режиме




#### Переход по коду

Для перехода по коду существует отдельный функционал:

Для перехода на следующую строчку кода (не заходя во внутренние механизмы), нужно нажать клавишу <F10>, либо в верхней панели инструментов  «шаг с обходом».

Для перехода по коду заходя во внутренние конструкции, нужно нажать клавишу <F11>, либо в верхней панели инструментов  «шаг с заходом».

Для того что бы просто продолжить выполнение программы, нужно нажать <F5>, либо в верхней панели инструментов  Продолжить ▾ «продолжить».

#### Окно команд

При отладке кода может понадобиться проверить как работают некоторые функции, что возвращают. В этом может помочь «окно интерпретации». Во время остановки программы в точке прерывания в окне интерпретации можно просто писать код, в частности обращаться к параметрам и методам видимым из данного места прерывания (см. рисунок 1.7).

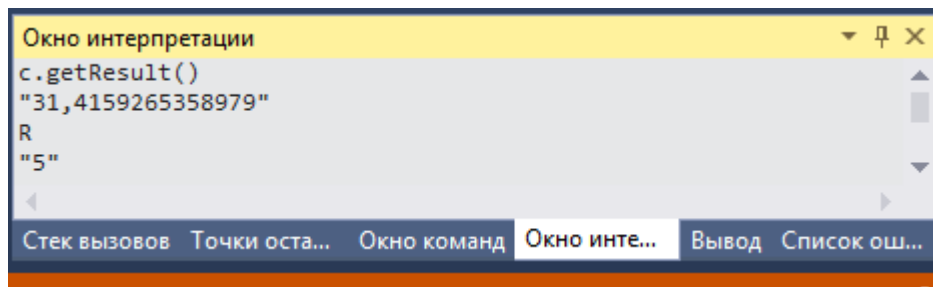


Рисунок 1.7 – Окно интерпретации

## Лабораторная работа №2

### Цель и задачи

- изучить методы работы со строками, списками и словарем;
- изучить методы выборки;
- разработать приложение в соответствии с заданием.

### Описание задания

Все задание выполняется как и в прошлой лабораторной работе с помощью «меню», и приложение работает до того момента пока пользователь не завершит работу, набрав соответствующую команду. То есть для выполнения первого задания вводят в консоль цифру 1 и т.д. для выхода из приложения “exit”.

### Выделение части строки по двум разделителям используя класс String

Пользователь вводит некоторое сообщение. Далее пользователь вводит часть слова с которого он хочет начинать вырезку предложения и часть слова определяющее конец предложения.

Пример, текста:

```
<html>
<head>
  <title>Hello world!</title>
</head>
<body>

</body>
</html>
```

Далее пользователь вводит часть предложения с которого он хочет начинать вырезку. Например: “<title>”. Далее вводит часть предложения которой завершается вырезка. Например: “</title>”.

Результат работы: “Hello world!”.

Так же необходимо предусмотреть механизм обработки ошибок (try – catch) на существование начальных и конечных частей предложения и то что они идут друг за другом.

### Нахождение суммы и среднего значения, мин. и макс. целочисленного списка

Данное задание должно быть описано в отдельный класс с функциями нахождения суммы, среднего значения, минимума и максимума. Должен быть реализован один конструктор с входным параметром строки. Пример строки: «11,54,34,12,43,543,6,16». Попадая в конструктор необходимо разбить данную строку по запятой, в результате получится список (массив), его необходимо преобразовать в целочисленное множество и поместить в глобальную переменную (поле).

Далее необходимо описать 4 метода для нахождения суммы и среднего значения, минимума и максимума списка.

### Работа со списками

Данный блок заданий нужно выполнять в 3 разных вариантах:

- Итеративных методов, то есть методом перебора проверяем условия отбора;
- Лямбда выражений;
- LINQ запросов.

Необходимо создать класс (сущности), который будет характеризовать студента. То есть создать класс в котором будет описано не менее 3 полей: имя, фамилия, возраст.

Необходимо создать класс управления списками студентов (manage\_student). В нем должен быть создан список с набором уже сформированного списка студентов.

#### *1.1. Фильтрация студентов по имени*

Реализовать 2 метода:

- формировать список студентов имена которых начинаются ранее заданного пользователем символа (буквы);
- искать пользователя имя которое содержит часть слова которое напишет пользователь. Пример: пользователь написал часть слова «ал» метод должен найти все имена в которых присутствует часть слова «ал» (В качестве лямбда выражения за основу взять string.Compare).

#### *1.2. Нахождение студентов по возрасту*

Нахождение студентов возраст которых соответствует введенным данным пользователя. То есть, пользователь задает начальное значение, например 18, и конечное значения поиска, например 19. Результат будет список студентов возраст которых соответствует заданным.

#### *1.3. Сортировка*

Создать двойную сортировку. Например: изначально делать сортировка по имени, далее сортировку по возрасту.

#### *1.4. Сложная фильтрация*

Реализовать фильтрацию студентов по следующему критерию: возраст студентов должен без остатка делиться на 7 и 3.

## 2. Словарь

## Лабораторная работа №3

### Цель и задачи

- изучение создания системных служб Windows;
- создание службы для загрузки почты.

### Описание задания

Создать windows-службу, которая через определенный момент времени, производит опрос сервера на наличие почты на нем (с помощью внешней библиотеки OpenPop).

При получении почты, информация каждого письма должна сохраняться в отдельный каталог в следующем виде:

- весь текст, который есть в сообщении необходимо записать в текстовый файл;
- прикрепленные к письму файлы также должны быть сохранены в каталоге письма, при этом тип сохраненного файла должен соответствовать действительности. (Например: музыкальный файл никак не должен быть с расширением docx).

Служба должна иметь 2 основных метода: `onStart()` и `onStop()` и они оба должны выполнять свою функцию в соответствии с названием.

Служба должна запускать опрос сервера с определенной периодичностью, периодичность должна быть не чаще 3 минут, задается с помощью конфигурационного файла (данную функцию реализовать с помощью класса `Timer`).

Для службы должен быть создан конфигурационный файл, позволяющий задать следующие параметры:

- адрес pop-сервера (imap-сервера);
- логин;
- пароль;
- периодичность опроса;
- указание базового каталога хранения сообщений.

При последующем опросе почтового сервера производить проверку на существование данного сообщения в директории:

- если сообщение уже существует, то переходить к следующему сообщению;
- если сообщения не существует, тогда необходимо его подгрузить;
- если существует сообщение, которого не было в списке сообщений с сервера, такое письмо следует удалить.

### Технические требования

При выполнении задания должны быть использованы классы `Pop3Client` (библиотеки `OpenPop`) для установления соединения с pop-сервером. Класс `Timer` для запуска основного метода с определенным интервалом.

В итоге должна получиться скомпилированная служба, которую в итоге необходимо инициализировать и запустить, данные для подключения к pop-серверу должны браться из конфигурационного файла. После демонстрации выполненной работы службу необходимо деинсталлировать.

## Теория и примеры реализации

### Создания проекта и описание класса службы

Создание проекта службы Windows (см. рисунок 3.1).

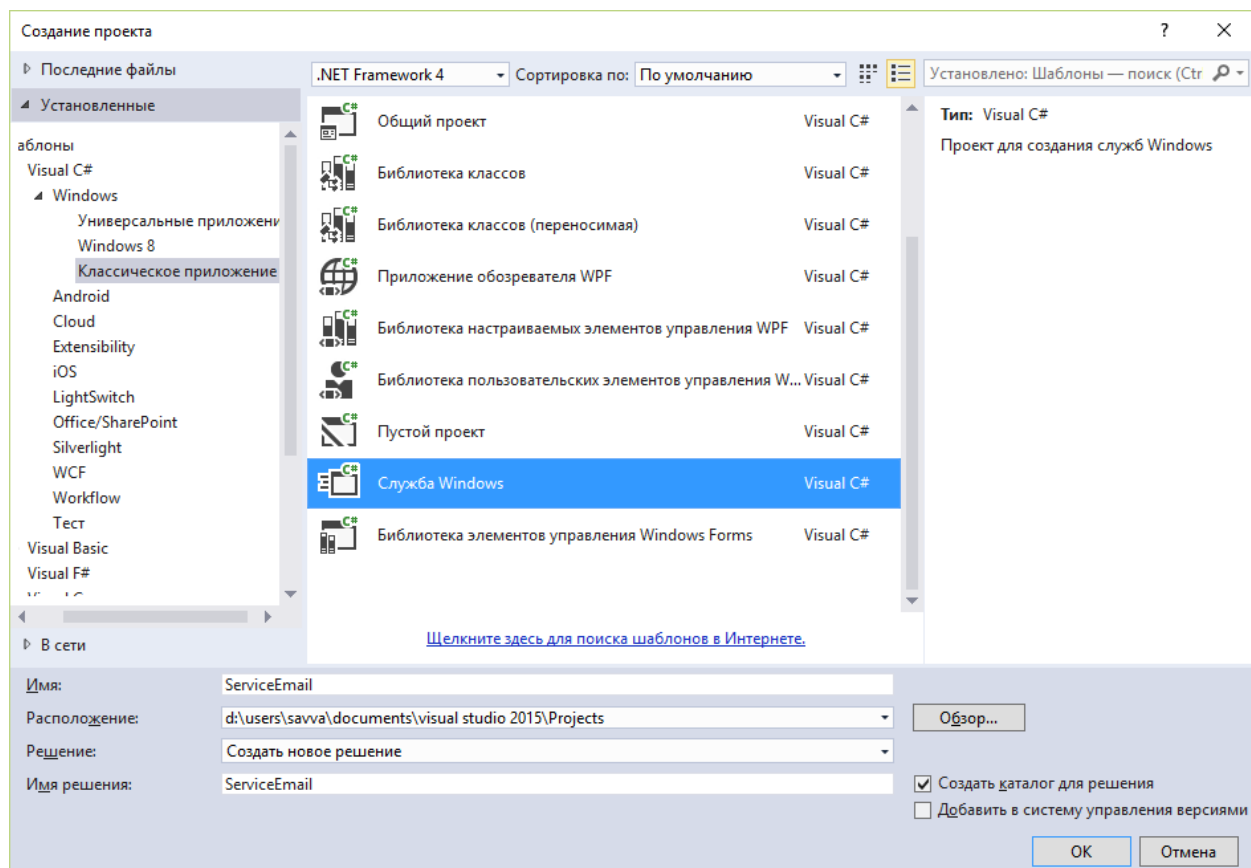


Рисунок 3.1 – Форма создания «Службы Windows»

В обозревателе решений открываем файл Service1.cs. В открывшейся вкладке необходимо нажать на ссылку «щелкните здесь для перехода в режим отображения кодов».

Инициализируем в методе OnStart() – таймер, для того чтобы событие работало не постоянно а с некоторым интервалом ожидания (см. рисунок 3.2).

```
//объявление глобальной переменной типа Timer
Timer tr = null;
//Событие запуска сервиса
ссылка 0
protected override void OnStart(string[] args)
{
    File.AppendAllText("MyFirstService.log",
        Environment.NewLine + "Start service");

    //Создание экземпляра класса Timer
    tr = new System.Timers.Timer();
    //Инициализация параметра "интерва"
    tr.Interval = 10000 * 10; //10000 - 1с.
    //работает без перерывов
    tr.AutoReset = true;
    //инициализация нашего события OnTimedEvent
    tr.Elapsed += new ElapsedEventHandler(OnTimedEvent);
}
//событие которое будет производить запись в файл
ссылка 1
private void OnTimedEvent(object source, ElapsedEventArgs e)
{
    //запись в конец файла
    File.AppendAllText("MyFirstService.log",
        Environment.NewLine + " " + DateTime.Now.ToString("dd/MM/yyyy hh:mm:ss"));
}
//Событие остановки сервиса
ссылка 0
protected override void OnStop()
{
    File.AppendAllText("MyFirstService.log",
        Environment.NewLine + "Stop service");
}
```

Рисунок 3.2 – листинг службы

При запуске службы будет создавать и стартовать *непрерывный* таймер с инициализацией параметров. Таймер в свою очередь, будет запускать метод OnTimedEvent по истечению заданного интервала (то есть 10 секунд).

При остановке службы, таймер будет останавливать свою работу.

Класс Timer:

Свойство:

- AutoReset – дает указание работы класса Timer, false – timer прорабатывает один раз, true – timer будет вызывать событие неоднократно.
- Interval – задает значение частоты вызова события Elapsed. В миллисекундах.

Методы:

- Start – начало вызова события Elapsed.
- Stop – прекращает вызов события Elapsed.

События:

- Elapsed – событие которое будет вызываться при запуске таймера.

### Добавление установщиков в службу

Переходим в конструктор класса службы. В *обозревателе решений* выделить файл в котором описана работа службы, нажимаем правой кнопки мышки (ПКМ) и выбираем пункт меню «Открыть в конструкторе».

В открывшейся вкладке, в центре поля нажать правую кнопку мышки (ПКМ) и выбрать пункт меню «Добавить установщик».

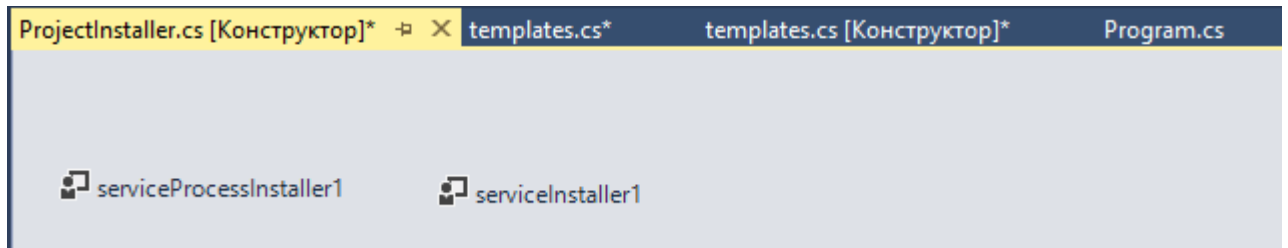


Рисунок 3.3 – вкладка установщика сервиса

Дальше устанавливаем свойства.

1. Выделяем объект serviceInstaller1 ПКМ -> Свойства.

Нужно изменить несколько свойств (см. рисунок 3.4):

- ServiceName – имя службы;
- Description – описание службы, информация также отображается в окне управления службами, и помогает пользователям идентифицировать службы;
- DisplayName – текст, который будет отображаться в окне «Службы» в столбце *Имя*.
- StartType – тип запуска службы: автоматический, ручной, отключен. Обычно устанавливают автоматический.

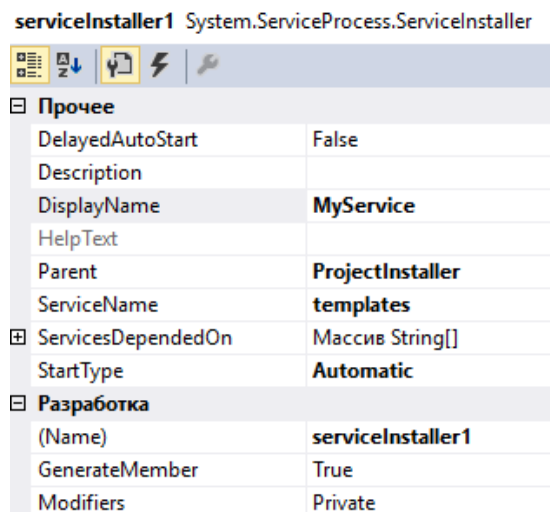


Рисунок 3.4 – пример заполнения свойств инсталлера

2. Выделяем объекта serviceProcessInstaller1 ПКМ -> Свойства.

Нужно изменить только одно свойство: Account присвоить ему значение **LocalSystem**. Получает или задает тип учетной записи, под управлением которой должно запускаться данное служебное приложение.

## Лабораторная работа №4

Цель: создание оконного windows-приложения позволяющего отправлять сообщения по протоколу SMTP.

### Описание задания

Создать windows-приложение отвечающее следующим характеристикам:

Должно быть создано главное окно отправления сообщений:

- поле для ввода информации от кого сообщение (в данном случае имеется ввиду именно никнейм – псевдоним, а не почту отправителя);
- поле для ввода кому будет отправлено сообщение;
- поле для ввода темы сообщения;
- поле для ввода тела сообщения;
- кнопка «отправить»
- кнопка перехода в окно настройки параметров подключения.

Должно быть предусмотрено окно настройки параметров smtp-сервера:

- поле ввода информации о SMTP-сервере;
- поле ввода информации логина;
- поле ввода информации пароля;
- кнопка сохранить (настройки должны сохраняться в отдельном файле конфигурации находящимся в одном каталоге с проектом).

### Технические требования

Приложение описанное ранее должно позволять производить настройки подключения к SMTP-серверу, и отправлять почту с помощью класса `SmtpClient`, `MailMessage`.



## Лабораторная работа №5

Цель: разработка windows-приложения позволяющего просматривать информацию о скаченных с почты файлах.

### Описание задания

Данное задание является доработкой лабораторной работы №3 – 4. В итоге должно получиться одно приложение с возможностью отправить сообщение другому пользователю, прочитать входящие сообщения, а так же просмотреть вложенные файлы.

Форма из лабораторной работы №4 будет доработана следующим образом:

1. Главная страница приложения должна иметь 2 вкладки.
  - а. Отправить сообщение. В этой вкладке должны быть описаны все объекты которые были описаны в прошлой лабораторной работе.
  - б. Входящие сообщения.
    - список (дерево) входящих сообщений с прикрепленными файлами;
    - объект отображение сообщения;
    - объект отображения прикрепленного файла;
    - кнопка «обновить»;
    - кнопка перехода в окно настройки конфигурации (окно конфигурации).
2. Должно быть реализовано 2 конфигурационных окна.
  - а. Окно конфигурации отправляемых сообщения. Это окно оформляется в соответствии с предыдущей лабораторной работой.
  - б. Окно конфигурации входящих сообщений:
    - поле ввода информации о пор-сервере;
    - поле ввода информации логина;
    - поле ввода информации пароля;
    - кнопка сохранить.

На вкладке входящие сообщения при нажатии на элемент списка входящих сообщений в полях отображения сообщения должен отобразиться текст сообщения.

При выборе подгружаемого файла его информация должна отобразиться в поле отображения прикрепленного файла. То есть

- файл текстового документа (и reach text), должен полностью отобразиться в заданном поле;
- файл изображения, должен отобразиться отобразится в заданном поле;
- если файл это pdf-документ, он так же должен отобразиться в заданном поле.

Все конфигурационные данные должны сохраняться в отдельном файле и так же оттуда подгружаться.

### Технические требования

Все конфигурационные файлы должны храниться в отдельном файле. Приложение должно позволять отправку сообщений, чтение сообщения а так же просмотр прикрепленных файлов к сообщению.

## ПРИЛОЖЕНИЕ А

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего профессионального образования

**«Уральский федеральный университет имени первого Президента России Б.Н.  
Ельцина»**

Институт радиоэлектроники и информационных технологий – РтФ  
Кафедра «Информационных технологий»

## **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № N**

Студен группы:  
[номер группы]  
ФИО

Преподаватель:  
Ростунцев С. Д.

Екатеринбург  
2016

## ПРИЛОЖЕНИЕ Б