

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Магнитогорский государственный технический университет им. Г.И. Носова»

Многопрофильный колледж



УТВЕРЖДАЮ  
Директор  
С.А. Махновский  
08.02.2023г

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ  
ДЛЯ ЛАБОРАТОРНЫХ ЗАНЯТИЙ  
УЧЕБНОЙ ДИСЦИПЛИНЫ**

**ОП.04 Основы алгоритмизации и программирования**

**для обучающихся специальности**

**09.02.07 Информационные системы и программирование**

Магнитогорск, 2023

## ОДОБРЕНО

Предметно-цикловой комиссией  
«Информатики и вычислительной техники»  
Председатель Т.Б. Ремез  
Протокол № 6 от «25» января 2023

Методической комиссией МпК

Протокол № 4 от «08» февраля 2023

### Разработчик (и):

преподаватель ФГБОУ ВО «МГТУ им. Г.И. Носова» Многопрофильный колледж

Л.А. Фетисова

преподаватель ФГБОУ ВО «МГТУ им. Г.И. Носова» Многопрофильный колледж

С.М. Утралинова

Методические указания по выполнению лабораторных работ разработаны на основе рабочей программы учебной дисциплины / профессионального модуля «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ».

Содержание лабораторных работ ориентировано на подготовку обучающихся к освоению профессионального(ых) модуля(ей) программы подготовки специалистов среднего звена по специальности 09.02.07 Информационные системы и программирование (квалификация Программист) и овладению профессиональными компетенциями.

## СОДЕРЖАНИЕ

|                                                                                                                       |                                 |    |
|-----------------------------------------------------------------------------------------------------------------------|---------------------------------|----|
| 1 Введение .....                                                                                                      | Ошибка! Закладка не определена. | 4  |
| 2 Методические указания.....                                                                                          |                                 | 6  |
| Лабораторное занятие № 1 Знакомство со средой программирования. ....                                                  |                                 | 6  |
| Лабораторное занятие № 2 Составление программ линейной структуры. ....                                                |                                 | 8  |
| Лабораторное занятие № 3 Составление программ разветвляющейся структуры.....                                          |                                 | 10 |
| Лабораторное занятие № 4 Составление программ циклической структуры.....                                              |                                 | 13 |
| Лабораторное занятие № 5 Обработка одномерных массивов.....                                                           |                                 | 15 |
| Лабораторное занятие № 6 Обработка двумерных массивов. ....                                                           |                                 | 17 |
| Лабораторное занятие № 7 Работа со строками.....                                                                      |                                 | 19 |
| Лабораторное занятие № 8 Работа с данными типа множество.....                                                         |                                 | 20 |
| Лабораторное занятие № 9 Файлы последовательного доступа. ....                                                        |                                 | 22 |
| Лабораторное занятие № 10 Типизированные файлы. Нетипизированные файлы.....                                           |                                 | 24 |
| Лабораторное занятие № 11 Организация процедур.....                                                                   |                                 | 25 |
| Лабораторное занятие № 12 Организация функций.....                                                                    |                                 | 27 |
| Лабораторное занятие № 13 Применение рекурсивных функций. ....                                                        |                                 | 31 |
| Лабораторное занятие № 14 Программирование модуля. ....                                                               |                                 | 31 |
| Лабораторное занятие № 15 Создание библиотеки подпрограмм. ....                                                       |                                 | 33 |
| Лабораторное занятие № 16 Использование указателей для организации связанных списков.....                             |                                 | 35 |
| Лабораторное занятие № 17 Изучение интегрированной среды разработчика.....                                            |                                 | 37 |
| Лабораторное занятие № 18 Создание проекта с использованием компонентов для работы с текстом. ....                    |                                 | 42 |
| Лабораторное занятие № 19 Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени..... |                                 | 44 |
| Лабораторное занятие № 20 События компонентов (элементов управления), их сущность и назначение .....                  |                                 | 44 |
| Лабораторное занятие № 21 Создание процедур на основе событий.....                                                    |                                 | 45 |
| Лабораторное занятие № 22 Создание проекта с использованием кнопочных компонентов. ....                               |                                 | 50 |
| Лабораторное занятие № 23 Создание проекта с использованием компонентов стандартных диалогов и системы меню. ....     |                                 | 52 |
| Лабораторное занятие № 24 Разработка функциональной схемы работы приложения ....                                      |                                 | 54 |
| Лабораторное занятие № 25 Создание процедур обработки событий. Компиляция и запуск приложения .....                   |                                 | 56 |
| Лабораторное занятие № 26 Разработка оконного приложения с несколькими формами .....                                  |                                 | 58 |
| Лабораторное занятие № 27 Разработка игрового приложения.....                                                         |                                 | 60 |
| Лабораторное занятие № 28 Разработка интерфейса приложения. ....                                                      |                                 | 62 |
| Лабораторное занятие № 29 Тестирование, отладка приложения. ....                                                      |                                 | 64 |
| Лабораторное занятие № 30 Классы ООП: виды, назначение, свойства, методы, события .....                               |                                 | 67 |
| Лабораторное занятие № 31 Объявления класса. ....                                                                     |                                 | 69 |
| Лабораторное занятие № 32 Создание наследованного класса.....                                                         |                                 | 71 |
| Лабораторное занятие № 33 Программирование приложений. ....                                                           |                                 | 73 |
| Лабораторное занятие № 34 Перегрузка методов.....                                                                     |                                 | 76 |

## 1 ВВЕДЕНИЕ

Важную часть теоретической и профессиональной практической подготовки обучающихся составляют лабораторные занятия.

Состав и содержание лабораторных занятий направлены на реализацию Федерального государственного образовательного стандарта среднего профессионального образования.

Ведущей дидактической целью лабораторных занятий является формирование профессиональных практических умений (умений выполнять определенные действия, операции, необходимые в последующем в профессиональной деятельности), необходимых в последующей учебной деятельности.

Ведущей дидактической целью лабораторных занятий является экспериментальное подтверждение и проверка существенных теоретических положений (законов, зависимостей).

В соответствии с рабочей программой учебной дисциплины «Основы алгоритмизации и программирования» предусмотрено проведение лабораторных занятий.

В результате их выполнения, обучающийся должен:

**уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

Содержание практических и лабораторных занятий ориентировано на подготовку обучающихся к освоению профессионального модуля программы подготовки специалистов среднего звена по специальности и овладению **профессиональными компетенциями:**

ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;

ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием;

ПК.1.3. Выполнять отладку программных модулей с использованием специализированных программных средств;

ПК 1.4. Выполнять тестирование программных модулей;

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода;

ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения;

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

А также формированию **общих компетенций:**

ОК 01 - Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 02 - Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.

ОК 04 - Эффективно взаимодействовать и работать в коллективе и команде.

ОК 05 - Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.

ОК 09 - Пользоваться профессиональной документацией на государственном и иностранном языках.

Выполнение обучающихся практических и/или лабораторных работ по учебной дисциплине «Основы алгоритмизации и программирования» направлено на:

- *обобщение, систематизацию, углубление, закрепление, развитие и детализацию полученных теоретических знаний по конкретным темам учебной дисциплины;*
- *формирование умений применять полученные знания на практике, реализацию единства интеллектуальной и практической деятельности;*
- *развитие интеллектуальных умений у будущих специалистов: аналитических, проектировочных, конструктивных и др.;*
- *выработку при решении поставленных задач профессионально значимых качеств, таких как самостоятельность, ответственность, точность, творческая инициатива.*

Лабораторные занятия проводятся в рамках соответствующей темы, после освоения дидактических единиц, которые обеспечивают наличие знаний, необходимых для ее выполнения.

## 2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ

### Тема 1.2. Типы данных

#### Лабораторное занятие № 1 Знакомство со средой программирования.

##### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

##### Выполнив работу, Вы будете:

###### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

##### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

##### Задание:

1. Какие операции определены над переменными булевого типа?
2. Записать на языке C++ следующие выражения:

а.  $\frac{x^2 + 3x - y}{a \sin x + e^y}$  б.  $\frac{ab^{-2}}{2c}$

3. Организовать ввод и вывод данных заданных типов, снабдив распечатки соответствующими заголовками

x='\*'

y='/'

z=0,75

J=0.0

$\pi$

Дать протокол программы.

4. Какого типа будет результат деления 15 на 4?
5. Какие из приведенных ниже операторов присвоения являются правильными?
  - а. x:=I+J-B
  - б. I:=I+K/J

если известно, что I,J,K: integer; x,B: real?

6. Написать программу вычисления выражения

$$\frac{5.23 + 7.6^2 + \sin \frac{\pi}{7}}{\sin \frac{2\pi}{7} + 3.1}$$

Дать протокол программы.

**Порядок выполнения работы:**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

**Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

**Форма предоставления результата**

- Код программы.

**Критерии оценки:**

«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 2.1. Операторы языка программирования

### Лабораторное занятие № 2 Составление программ линейной структуры.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

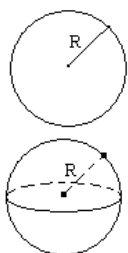
Calculate Linux Desktop

#### Задание:

##### 1. Формулировка задачи

Составить алгоритм вычисления длины окружности, площади круга, площади сферы и объема шара по заданному радиусу окружности.

##### 2. Математическая постановка задачи



Для расчета перечисленных характеристик воспользуемся формулами:

длина окружности –  $L = 2\pi R$ ;

площадь круга –  $S_{кр} = \pi R^2$ ;

площадь сферы –  $S_{сф} = 4\pi R^2$ ;

объем шара –  $V = \frac{4}{3}\pi R^3$ ,

где  $\pi$  – число Пи, математическая константа, которая выражает отношение длины окружности к её диаметру  $\pi \approx 3,141592653589793238462643...$ ,  $R$  – радиус окружности.

##### 3. Выбор переменных программы

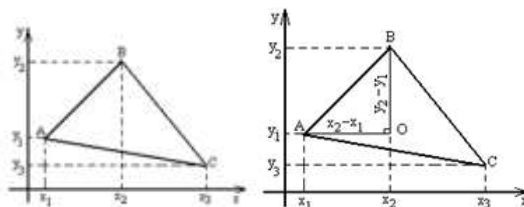
Из приведенного выше решения определяем следующие переменные:

исходные данные – радиус окружности ( $R$ );

справочные данные – число  $\pi$  ( $Pi$ );  
результат – длина окружности ( $L$ ), площадь круга ( $S_{кр}$ ), площадь сферы ( $S_{сф}$ ) и объем шара ( $V$ ).

### 1. Формулировка задачи

Треугольник задается координатами своих вершин на плоскости:  $A(x_1, y_1)$ ,  $B(x_2, y_2)$ ,  $C(x_3, y_3)$ . Требуется составить алгоритм программы, которая вычисляет площадь треугольника  $ABC$ .



### 2. Математическая постановка задачи

Для решения задачи можно использовать формулу Герона:  
 $S = \sqrt{p(p - A)(p - B)(p - C)}$ , где  $p$  – полупериметр. Для вычисления длины сторон по координатам вершин необходимо воспользоваться формулой  $|AB| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ , где  $|AB|$  – длина сторона треугольника;  $(x_1, y_1)$ ,  $(x_2, y_2)$  – координаты вершин  $A$  и  $B$ .

### 3. Выбор переменных программы

Из приведенного выше решения определяем следующие переменные:  
исходные данные – координаты вершин на плоскости  $(x_1, y_1)$ ,  $(x_2, y_2)$ ,  $(x_3, y_3)$ ;  
промежуточные данные – длины сторон ( $A$ ,  $B$ ,  $C$ ),  $p$  – полупериметр.  
результат – площадь треугольника ( $S$ ).

### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата

- Алгоритм программы.
- Код программы.

### Критерии оценки:

«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой

обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 2.1. Операторы языка программирования**

### **Лабораторное занятие № 3 Составление программ разветвляющейся структуры.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### ***уметь:***

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

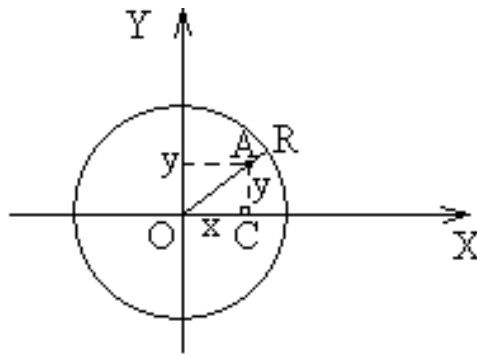
MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

1. Составить алгоритм решения задачи, который определяет по введенным координатам, попадает ли заданная точка в окружность с центром в точке  $O(0;0)$  и заданным радиусом  $R$ .
2. Математическая постановка задачи



Заданная точка имеет координаты  $x, y$ . Рассмотрим треугольник AOC.  $\angle OCA = 90^\circ$ ,  $OC = x$ ,  $AC = y$ , следовательно, по теореме Пифагора:  
 $AO^2$  (гипотенуза)  $= OC^2 + AC^2$  ( $AO^2 = x^2 + y^2$ ).

Поэтому условие принадлежности точки окружности можно записать в виде:  $x^2 + y^2 \leq R^2$ .

### 3. Выбор переменных программы

Из приведенного выше решения определяем следующие переменные:

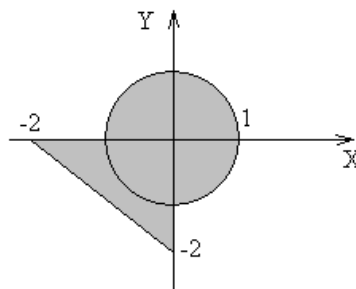
исходные данные – радиус окружности ( $R$ ) и координаты точки ( $x$  и  $y$ );

результат – сообщение «В окружности» или «Вне окружности».

Так как радиус окружности и координаты точки могут принимать любые значения (2; 2,5; 3,75), все переменные для данной задачи определены как действительные числа.

### 1. Формулировка задачи

Составить алгоритм решения задачи, который по введенным координатам точки определяет, попадает ли эта точка в заштрихованную область.



### 2. Математическая постановка задачи

Запишем условия попадания точки в область в виде формул. Область можно описать как круг, пересекающийся с треугольником. Точка может попадать либо в круг, либо в треугольник, либо в их общую часть. Условие попадание точки в круг запишем как  $x^2 + y^2 \leq 1^2$ , т.к. радиус окружности в соответствии с графической интерпретацией равен 1.

Так как треугольник расположен в третьей четверти координатной плоскости, следовательно, для него выполняется условие  $\begin{cases} x \leq 0 \\ y \leq 0 \end{cases}$ . Выведем уравнение прямой линии, ограничивающей треугольник снизу.

Известно, что прямая линия проходит через две точки:  $(-2; 0)$  и  $(0; -2)$ . Требуется составить уравнение прямой линии.

Общий вид уравнения прямой  $y = kx + b$ , где  $k, x, b$  – фиксированные числа.

Искомая прямая  $y = kx + b$  с пока неизвестными коэффициентами  $k, x, b$  проходит через точки  $(-2; 0)$  и  $(0; -2)$ , а, значит, выполняются равенства  
 $0 = k \cdot (-2) + b$  и  $-2 = k \cdot 0 + b$ , что можно записать в виде системы:  
 $\begin{cases} 0 = k \cdot (-2) + b \\ -2 = k \cdot 0 + b \end{cases}$  или  $\begin{cases} 0 = k \cdot (-2) + b \\ -2 = b \end{cases}$ .

Решив систему относительно неизвестных  $k$  и  $b$ , мы найдем уравнение прямой.

Подставим значение  $b = -2$  в уравнение  $0 = k \cdot (-2) + b$ :

$$\begin{aligned}-2k &= -b \\ -2k &= -(-2) \\ -2k &= 2 \\ k &= -1\end{aligned}$$

Таким образом, искомое уравнение прямой имеет вид  $y = (-1) \cdot x - 2$ . Так как заштрихованная область расположена выше прямой линии, ограничивающей треугольник снизу, то попадание точки в треугольник запишем как  $\begin{cases} x \leq 0 \\ y \leq 0 \\ y \geq -x - 2 \end{cases}$ . Следовательно, заштрихованная область

определена следующими условиями  $x^2 + y^2 \leq 1^2$  или  $\begin{cases} x \leq 0 \\ y \leq 0 \\ y \geq -x - 2 \end{cases}$ .

### 3. Выбор переменных программы

Из приведенного выше решения определяем следующие переменные:

исходные данные – координаты точки ( $x$  и  $y$ );

результат – сообщение «Принадлежит» или «Не принадлежит».

Так как координаты точки могут принимать любые значения (2; 2,5; 3,75), все переменные определяем как действительные числа.

### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата

- Алгоритм программы.
- Код программы.

### Критерии оценки:

«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 2.1. Операторы языка программирования

### Лабораторное занятие №4 Составление программ циклической структуры

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1. Формулировка задачи

Составить алгоритм расчета факториала  $N!$  с использованием различных видов цикла.

##### 2. Математическая постановка задачи

Факториал числа  $N$  (обозначается  $N!$ ) – произведение всех натуральных чисел до  $N$  включительно:

$$N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot N$$

По определению полагают  $0! = 1$ . Факториал определён только для целых неотрицательных чисел.

##### 3. Выбор переменных программы

Из приведенного выше решения определяем следующие переменные:

исходные данные – значение  $N$ ;

результат – значение *factorial*.

Выбор переменных: аргумент и значение функции могут принимать целые неотрицательные значения.

### 1. Формулировка задачи

Составить алгоритм вычисления суммы ряда:

$$y = x - \frac{x^2}{2} + \frac{x^3}{3} - \dots + \frac{(-1)^{n-1} x^n}{n} \text{ с точностью } e=0.0001 \text{ с использованием цикла с}$$

постусловием и предусловием.

### 2. Математическая постановка задачи

Ряд представляет собой сумму слагаемых, в котором перед каждым слагаемым чередуется знак (перед первым слагаемым знак «+», перед вторым – «-», перед третьим – «+», перед четвертым – «-», и т.д.). Поэтому для его учета в алгоритм накопления суммы вводим переменную, отвечающую за знак – z.

При накоплении суммы, начальное значение суммы обнуляем, т.е. S=0. Первое слагаемое можно представить в виде  $\frac{x^1}{1}$ , т.е. начальное значение степени x и знаменатель равны 1. Шаг изменения степени и знаменателя совпадает и равен 1. Точность вычисления определяется значением слагаемого.

### 3. Выбор переменных программы

Из приведенного выше описания определяем следующие переменные:

- исходные данные – значение x;
- начальное значение суммы S равно 0 (S=0);
- начальное значение степени x и знаменатель равны 1 (i=1);
- знак перед первым слагаемым – «+» (z=1);
- результат – значение S.

### Порядок выполнения работы:

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата:

- Алгоритм программы.
- Код программы.

### Критерии оценки:

«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным

материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 2.1. Операторы языка программирования**

### **Лабораторное занятие № 5 Обработка одномерных массивов.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

*Заполнение массива с помощью генератора случайных чисел*

Для генерации чисел в диапазоне  $[A, B]$  можно использовать следующее выражение:  
 $random(B-A+1)+A$ .

Например, выражение  $A[i]=random(100)$  генерирует числа от 0 до 99, при этом 100 не входит в диапазон.

Записать выражение для генерации чисел в диапазоне  $[-15; 38]$ :

$A[i]=random(38-(-15)+1)+(-15)$ , т.е.  $A[i]=random(54)-15$ .

#### **1. Формулировка задачи**

Заполнить массив  $n$  целыми случайными числами в диапазоне  $[-10; 15]$ . Составить алгоритм вычисления суммы четных чисел.

## 2. Математическая постановка задачи

При накоплении суммы, начальное значение суммы обнуляем, т.е.  $Sum=0$ .

Выражение для генерации чисел в диапазоне  $[-10; 15]$ :

$A[i]=random(15-(-10)+1)+(-10)$ , т.е. запишем в следующем виде  $A[i]=random(26)-10$ .

Для четного значения элемента выполняется условие  $A[i] \bmod 2=0$ .

## 3. Выбор переменных программы

Из приведенного выше описания определяем следующие переменные:

исходные данные – количество элементов массива  $n$ ;

начальное значение суммы равно 0 ( $Sum=0$ );

результат – значение накопленной суммы  $Sum$ .

## Задача

### 1. Формулировка задачи

Заполнить массив  $n$  целыми случайными числами в диапазоне  $[-5; 10]$ . Составить алгоритм вычисления произведения положительных элементов.

### 2. Математическая постановка задачи

При накоплении произведения, начальное значение произведения равно 1, т.е.  $pr=1$ .

Выражение для генерации чисел в диапазоне  $[-5; 10]$ :

$A[i]=random(10-(-5)+1)+(-5)$ , т.е. запишем в следующем виде  $A[i]=random(16)-5$ .

Для положительного значения элемента выполняется условие  $A[i] > 0$ .

### 3. Выбор переменных программы

Из приведенного выше описания определяем следующие переменные:

исходные данные – количество элементов массива  $n$ ;

начальное значение произведения равно 1 ( $pr=1$ );

результат – значение произведения  $pr$ .

## Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

## Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

## Форма предоставления результата

- Алгоритм программы.
- Код программы.

## Критерии оценки:

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные

программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 2.1. Операторы языка программирования**

### **Лабораторное занятие № 6 Обработка двумерных массивов.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

##### **1. Формулировка задачи:**

Составить алгоритм и написать программу:

Вычислить сумму и число положительных элементов матрицы  $A[N, N]$ , находящихся над главной диагональю.

### 1. Формулировка задачи:

Составить алгоритм и написать программу:

Дана матрица  $B[N, M]$ .. Найти в каждой строке матрицы максимальный и минимальный элементы и поменять их местами с первым и последним элементом строки соответственно.

### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата

- Алгоритм программы.
- Код программы.

### Критерии оценки:

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 2.1. Операторы языка программирования

### Лабораторное занятие № 7 Работа со строками.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

*уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:
  - MS Windows 7 (подписка Imagine Premium)
  - MS Office 2007
  - 7 Zip
  - MS Visual Studio (подписка Imagine Premium)
  - Visual Studio Code
  - Calculate Linux Desktop

#### **Задание:**

##### 1. Формулировка задачи

- Дана строка, заканчивающаяся точкой. Подсчитать, сколько слов в строке.
- *Лишние пробелы.* Дана строка, состоящая из слов, разделенных пробелами. Напишите программу, удаляющую лишние пробелы. Пробел считается лишним, если он: стоит в начале строки.

##### 1. Формулировка задачи:

- Дана строка. Подсчитать, сколько в ней букв R, K, T.
- *Лишние пробелы.* Дана строка, состоящая из слов, разделенных пробелами. Напишите программу, удаляющую лишние пробелы. Пробел считается лишним, если он: следует за пробелом.

##### 2. Выбор переменных программы

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

- «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

- «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

- «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

**Тема 2.1. Операторы языка программирования****Лабораторное занятие № 8 Работа с данными типа множество.****Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

**Выполнив работу, Вы будете:****уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

**Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

**Задание:**

1. Формулировка задачи:

Дана непустая последовательность символов. Требуется построить и напечатать множество, элементами которого являются символы, встречающиеся в последовательности индивидуального варианта:

Вариант Последовательность символов

- 1 Буквы от 'A' до 'F' и от 'X' до 'Z';
- 2 Цифры от '5' до '9' и знаки арифметических операций;
- 4 Буквы от 'T' до 'X' и цифры от '1' до '4';
- 5 Знаки препинания и операций отношения;
- 6 Знаки арифметических операций и буквы от 'E' до 'N';
- 7 Буквы от 'A' до 'Z' и знаки препинания;
- 8 Знаки операций отношения;
- 9 Цифры от '0' до '9';
- 10 Знаки арифметических операций и операций отношения.

2. При выполнении задания необходимо:

- а) ознакомиться с конечным и упорядоченным множеством символов, определенным на используемой для выполнения задания ЭВМ;
- б) составить программу для конкретного варианта, работающую для произвольного набора символов.
- с) входная строка символов может быть длиннее строки экрана терминала, при этом программа работает не с функцией EOLN, а с признаком конца строки, который задается программистом.

**Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

**Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

**Форма предоставления результата**

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой

обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 2.1. Операторы языка программирования**

### **Лабораторное занятие № 9 Файлы последовательного доступа.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### ***уметь:***

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

##### **1. Формулировка задачи:**

Записать в файл прямого доступа ряд вещественных чисел. Найти наибольшее из значений модулей компонентов с нечетными номерами.

##### **1. Формулировка задачи:**

Записать в файл последовательного доступа произвольных натуральных чисел. Переписать в другой файл последовательного доступа те элементы, которые кратны  $K$ .

Вывести полученный файл на печать.

**Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

**Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

**Форма предоставления результата**

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 2.1. Операторы языка программирования

### Лабораторное занятие № 10 Типизированные файлы. Нетипизированные файлы.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1. Формулировка задачи:

Дана последовательность целых чисел, оканчивающаяся нулем. Записать все числа последовательности в типизированный файл.

В конец существующего типизированного файла записать:

- а) число 0;
- б) фразу «До свидания!».

##### 1. Формулировка задачи:

Создать типизированный файл, элементами которого являются двенадцать первых членов последовательности Фибоначчи (последовательности, в которой первые два члена равны 1, а каждый следующий равен сумме двух предыдущих).

#### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;

- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

**Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

**Форма предоставления результата**

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

### **Тема 3.1. Процедуры и функции**

#### **Лабораторное занятие № 11 Организация процедур.**

**Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

**Выполнив работу, Вы будете:**

**уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;

- оформлять код программы в соответствии со стандартом кодирования;

### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:
  - MS Windows 7 (подписка Imagine Premium)
  - MS Office 2007
  - 7 Zip
  - MS Visual Studio (подписка Imagine Premium)
  - Visual Studio Code
  - Calculate Linux Desktop

### **Задание:**

#### **1. Формулировка задачи:**

- Разработать программу, используя модульный подход.
  - Сформировать по одному или несколько заголовочных модулей и файлов реализации, которые будут содержать разработанные функции.
  - Написать главную функцию main, используя правила структурного подхода: она должна содержать в основном только вызов функций, все вычисления должны быть внутри разработанных функций.
  - Выполнить отладку и компиляцию программы, получить исполняемый файл.
  - Выполнить тестирование программы несколькими наборами входных данных.
  - Составить диаграмму модулей, дерево вызова функций и спецификации функций
- Даны целые числа  $a_1, \dots, a_n, b_1, \dots, b_m, k$ .

Если в последовательности  $a_1, \dots, a_n$  нет ни одного члена со значением  $k$ , то первый по порядку член этой последовательности, не меньший всех остальных членов, заменить на значение  $k$ . По такому же правилу преобразовать последовательность  $b_1, \dots, b_m$  применительно к 10.

### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

- «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные

программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

### **Тема 3.1. Процедуры и функции**

#### **Лабораторное занятие № 12 Организация функций.**

##### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

##### **Выполнив работу, Вы будете:**

###### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

##### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

##### **Задание:**

###### **1. Формулировка задачи:**

Написать функцию, выводящую в порядке возрастания элементы одномерного массива. В главной программе вызвать функцию для двух разных массивов.

## 1. Формулировка задачи:

Написать функцию вычисления произведения прямоугольной матрицы  $A$  размера  $k \times m$  на прямоугольную матрицу  $B$  размера  $m \times n$ . В главной программе обратиться к этой функции.

### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата

- Алгоритм программы.
- Код программы.

### Критерии оценки:

– «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 3.1. Процедуры и функции

### Лабораторное занятие № 13 Применение рекурсивных функций.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1. Формулировка задачи:

Ввести с клавиатуры целое число N. Используя рекурсию, распечатать сначала последовательность, состоящую из N букв 'A', а затем из N букв 'B'.

##### 1. Формулировка задачи:

Напечатать в обратном порядке последовательность чисел, признаком конца которой является 0.

#### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### Ход работы:

- Выделить входные и выходные данные

- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

### Тема 3.3. Модульное программирование

#### Лабораторное занятие № 14 Программирование модуля.

##### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

##### Выполнив работу, Вы будете:

###### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

##### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

##### Задание:

###### 1.Формулировка задачи:

Средствами модульного программирования решить задачу обработки одномерного числового массива.

Найти сумму элементов до первого положительного и произведение остальных элементов.

###### 1.Формулировка задачи:

Средствами модульного программирования решить задачу обработки одномерного числового массива.

Найти количество элементов, стоящих на нечётных местах и меньших заданного числа.

##### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

##### Ход работы:

- Выделить входные и выходные данные

- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

### Тема 3.3. Модульное программирование

#### Лабораторное занятие № 15 Создание библиотеки подпрограмм.

##### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

##### Выполнив работу, Вы будете:

###### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

##### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

##### Задание:

###### 1.Формулировка задачи:

Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над обыкновенными дробями вида  $P/Q$  ( $P$  - целое,  $Q$  - натуральное):

- сложение;
- вычитание;
- умножение;
- деление;
- сокращение дроби;
- возведение дроби в степень  $N$  ( $N$  - натуральное);
- функции, реализующие операции отношения (равно, не равно, больше или равно, меньше или равно, больше, меньше).

Используя этот модуль, решить задачу:

Дан массив  $A$  - массив обыкновенных дробей. Найти сумму всех дробей, ответ представить в виде несократимой дроби. Вычислить среднее арифметическое всех дробей, ответ представить в виде несократимой дроби.

### 1. Формулировка задачи:

Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над обыкновенными дробями вида  $P/Q$  ( $P$  - целое,  $Q$  - натуральное):

- сложение;
- вычитание;
- умножение;
- деление;
- сокращение дроби;
- возведение дроби в степень  $N$  ( $N$  - натуральное);
- функции, реализующие операции отношения (равно, не равно, больше или равно, меньше или равно, больше, меньше).

Используя этот модуль, решить задачу:

Дан массив  $A$  - массив обыкновенных дробей. Отсортировать его в порядке возрастания.

### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата

- Алгоритм программы.
- Код программы.

### Критерии оценки:

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 4.1 Указатели.

### Лабораторное занятие № 16 Использование указателей для организации связанных списков.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1.Формулировка задачи:

Сформировать односвязный список из 5 записей, удалить записи, удовлетворяющие некоторому условию, и вывести список на экран.

Запись содержит марку магнитофона, качество, цену. Удалить из списка записи с ценой ниже 800 руб.

##### 1.Формулировка задачи:

Сформировать односвязный список из 5 записей, удалить записи, удовлетворяющие некоторому условию, и вывести список на экран.

Запись содержит фамилию и 4 оценки. Удалить из списка записи с неудовлетворительными оценками

#### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

**Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

**Форма предоставления результата**

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 4.1 Указатели.

### Лабораторное занятие № 17 Изучение интегрированной среды разработчика.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

Формулировка задачи:

Изучить алгоритм решения задачи определения траектории полета снаряда, запущенного под углом к горизонту.

Уравнения, описывающие движение снаряда, т.е. зависимость его координат  $x$  и  $y$  от времени  $t$ , прошедшего с момента выстрела, выглядят следующим образом:

$$\begin{aligned}x(t) &= x_0 + V_{0x}t; \\ y(t) &= y_0 + V_{0y}t - \frac{gt^2}{2}; \\ V_{0x} &= V_0 \cos \alpha; \\ V_{0y} &= V_0 \sin \alpha.\end{aligned}$$

Здесь введены следующие обозначения:

$x_0, y_0$  — координаты начальной точки траектории полета снаряда, м;

$V_0$  — начальная скорость снаряда, м/с;

$\alpha$  — угол наклона начального участка траектории полета снаряда, рад;

$V_{0x}, V_{0y}$  — проекции вектора начальной скорости  $V_0$  на координатные оси  $x$  и  $y$ , соответственно, м/с;

$g$  — ускорение свободного падения,  $g = 9,81 \text{ м/с}^2$ .

Будем считать, что необходимо определить положение (координаты) снаряда в моменты времени  $0 \text{ с}, \Delta t, 2\Delta t, \dots, (n-1)\Delta t$ , причем

$$\Delta t = \frac{T}{n-1},$$

где  $T$  — время полета снаряда, с,

$n$  — число точек, в которых рассчитываются координаты траектории полета снаряда.

Величина  $T$  определяется по формуле

$$T = \frac{2V_{0y}}{g}.$$

Исходя из записанных выше уравнений, алгоритм решения задачи состоит из следующих действий:

1. Ввести начальные значения скорости снаряда  $V_0$ , угла наклона начального участка траектории  $\alpha$  и число точек  $n$ .
2. Вычислить значение  $V_{0x}$
3. Вычислить значение  $V_{0y}$
4. Присвоить  $g = 9,81 \text{ м/с}^2$ .
5. Вычислить  $T$
6. Вычислить  $\Delta t$
7. Присвоить  $i = 1$ .
8. Присвоить  $t = 0$ .
9. Вычислить  $x(t)$
10. Вычислить  $y(t)$
11. Вывести координаты снаряда.
12. Присвоить  $t = t + \Delta t$ .
13. Присвоить  $i = i + 1$ .
14. Если  $i \leq n$ , то перейти к шагу 9, иначе остановить выполнение программы.

Написать программу, вычисляющую координаты траектории полета снаряда

Выполнить компиляцию программы.

Исправить синтаксические ошибки, если такие имеются, и повторить компиляцию программы.

Запустить программу и проанализировать результаты ее работы.

Добавить в окно просмотра *Watch* переменные  $t, T, n, i$ .

### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 4.1 Указатели.

### Лабораторное занятие № 18 Создание проекта с использованием компонентов для работы с текстом.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

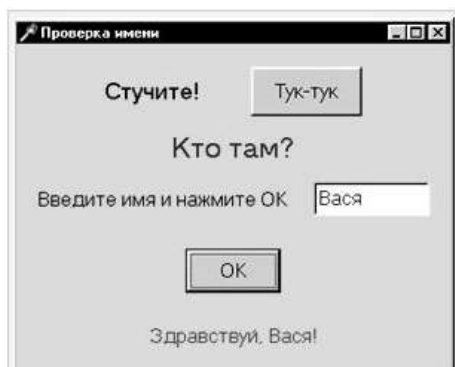
Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1.Формулировка задачи:

Компьютер должен запросить имя пользователя (пароль). Если пользователь дает правильный ответ (вводит знакомое имя, например, Вася), то компьютер приветствует его. В случае ввода любого другого слова, компьютер должен реагировать иначе, например, выводить запрос: "А где Вася?"



**Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

**Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

**Форма предоставления результата**

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)

### Лабораторное занятие № 19 Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1. Формулировка задачи:

Создайте приложение, при выполнении которого на экране появляется окно с тем же заголовком и с теми же компонентами, отображающими дату и время, что и на рисунке. Цвета частей календарей должны отличаться от стандартного цвета. Компоненты должны отображать текущую дату или текущее время. Компонент календаря должен быть установлен с текущим месяцем в середине этого компонента с использованием кнопок в его верхней части. При демонстрации выполнения приложения один из компонентов даты и времени должен быть раскрыт так же, как на рисунке



### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### Форма предоставления результата

- Алгоритм программы.
- Код программы.

### Критерии оценки:

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)**

### **Лабораторное занятие № 20 События компонентов (элементов управления), их сущность и назначение**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Выберите свое индивидуальное задание. Уточните условие задания, количество, наименование, типы исходных данных. Прочтите в Приложении 1 описание свойств и описание элементов управления Form, Label, TextBox, Button. С помощью окна свойств установите первоначальный цвет формы, шрифт выводимых символов.

##### **Задание**

- Разместите на форме ряд кнопок (Button) напротив каждой поле ввода (TextBox) и одну метку (Label). Создайте обработчики события нажатия на данные кнопки, которые будут менять текст в метке. Текст в метке берется из поля ввода напротив нажимаемой кнопки.
- Разместите на форме ряд кнопок (Button), и одно поле ввода (TextBox). Создайте обработчики события нажатия на данные кнопки, которые будут менять текст на нажатой кнопке. Текст на кнопке берется из поля ввода.

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

### **Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)**

#### **Лабораторное занятие № 21 Создание процедур на основе событий**

##### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

##### **Выполнив работу, Вы будете:**

###### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;

- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:  
 MS Windows 7 (подписка Imagine Premium)  
 MS Office 2007  
 7 Zip  
 MS Visual Studio (подписка Imagine Premium)  
 Visual Studio Code  
 Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Составить программу, отображающую графики функций  $\sin(x)$  и  $\cos(x)$  на интервале  $[X_{\min}, X_{\max}]$ . Предусмотреть возможность изменения разметки координатных осей, а так же шага построения таблицы. Прежде всего, следует поместить на форму сам элемент управления Chart. Он располагается в панели элементов в разделе Данные. Список графиков хранится в свойстве Series, который можно изменить, выбрав соответствующий пункт в окне свойств. Поскольку на одном поле требуется вывести два отдельных графика функций, нужно добавить еще один элемент. Оба элемента, и существующий и добавленный, нужно соответствующим образом настроить: изменить тип диаграммы ChartType на Spline. Здесь же можно изменить подписи к графикам с абстрактных Series1 и Series2 на  $\sin(x)$  и  $\cos(x)$  – за это отвечает свойство Legend. Наконец, с помощью свойства BorderWidth можно сделать линию графика по - толще, а затем поменять цвет линии с помощью свойства Color.

##### **Задание**

Постройте график функции для своего варианта. Таблицу данных получить путем изменения параметра X с шагом dx. Добавьте второй график для произвольной функции.

|                                             |                                        |
|---------------------------------------------|----------------------------------------|
| 1) $y = 10^{-2}bc / x + \cos \sqrt{a^3 x},$ | 2) $y = 1.2(a-b)^3 e^{x^2} + x,$       |
| $x_0 = -1.5; x_k = 3.5; dx = 0.5;$          | $x_0 = -0.75; x_k = -1.5; dx = -0.05;$ |
| $a = -1.25; b = -1.5; c = 0.75;$            | $a = 1.5; b = 1.2;$                    |

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.

- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 5.2 Интегрированная среда разработчика.**

### **Лабораторное занятие № 22 Создание проекта с использованием кнопочных компонентов.**

**Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

**Выполнив работу, Вы будете:**

**уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

**Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

## Visual Studio Code

### Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Когда происходит какое – либо событие (например, событие `Click` при нажатии на кнопку), в обработчик этого события передаются до-полнительные сведения об этом событии в параметре `e`. Например, при щелчке кнопки мыши на объекте возникает событие `MouseClick`. Для этого события параметр `e` содержит целый ряд переменных, которые позволяют узнать информацию о нажатии:

- `Button` – какая кнопка была нажата;
- `Clicks` – сколько раз была нажата и отпущена кнопка мыши;
- `Location` – координаты точки, на которую указывал курсор в момент нажатия, в виде объекта класса `Point`;
- `X` и `Y` – те же координаты в виде отдельных переменных.

##### **Задание**

Если в индивидуальном задании используется элемент `Panel`, измените его цвет, чтобы он визуально выделялся на форме. Если используется элемент `Label`, не забудьте присвоить ему какой – либо текст, иначе он не будет виден на форме.

-Разработать программу, динамически порождающую на окне кнопки. Левый верхний угол кнопки определяется местоположением курсора при щелчке. Вывести надпись на кнопке с координатами ее левого верхнего угла.

-Разработать программу, динамически порождающую на окне кнопки и поля ввода. Левый верхний угол элемента управления определяется местоположением курсора при щелчке. Кнопка порождается, если курсор находится в левой половине окна, в ином случае порождается поле ввода.

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

- «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 5.2 Интегрированная среда разработчика.**

### **Лабораторное занятие № 23 Создание проекта с использованием компонентов стандартных диалогов и системы меню.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### ***уметь:***

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Реализовать программно элемент управления «Список с прокруткой» для текстового режима. Работа со списком должна сводиться к вызову одной функции, в которую передаются координаты области на экране, которую занимает список, и массив строк, составляющих список. Функция самостоятельно организует интерфейс для выбора элемента.

Каждая строка представляет один элемент списка. В строках не должно быть символов возврата каретки и перевода строки. Элементы должны выводиться в заданном окне. Количество строк в

списке может как превышать высоту окна, так и быть меньше ее. В первом случае необходимо предусмотреть прокрутку содержимого в окне, во втором – обеспечить корректный выбор только тех позиций в окне, которые содержат строки списка (другими словами, курсор не должен выходить за пределы и окна, и списка).

Текущий элемент подсвечивается каким-либо цветом, заданным разработчиком. Выбор элемента осуществляется нажатием клавиш «Up» (перемещение курсора вверх) и «Down» (перемещение курсора вниз).

Подтверждение выбора (и выход из функции) осуществляется при нажатии клавиш Enter, Esc, Left, Right, Tab, Shift+Tab. Каждой из этих клавиш должен соответствовать определенный код, который также возвращается функцией выбора из списка. Этот код может использоваться вызывающей функцией для того, чтобы определить дальнейшее действие: переход к предыдущему (Shift+Tab, Left) или следующему (Tab, Right) элементу управления, выполнение основного действия программы (Enter), выход из программы (Esc).

Помимо указанного кода функция должна возвращать индекс выбранного элемента списка.

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 5.2 Интегрированная среда разработчика.

### Лабораторное занятие № 24 Разработка функциональной схемы работы приложения.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

Формулировка задачи:

Функциональная схема – это схема взаимодействия компонентов программного обеспечения с описанием информационных потоков, состава данных в потоках и указанием используемых файлов и устройств. Для изображения функциональных схем используют специальные обозначения, установленные стандартом ГОСТ 19.701-90 (ИСО 5807-85).

Задание

Создать приложение Сортировка, в котором с клавиатуры вводятся элементы массива целых чисел, производится сортировка элементами различными методами, результаты сортировки выводятся на экран. Эскиз функциональной схемы приведен на рисунке 1.

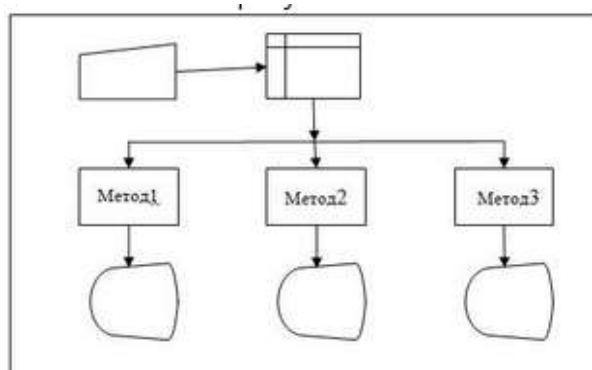


Рис. 1. Функциональная схема приложения

1. Разработать интерфейс и написать программные коды для событий кнопок.
2. Провести тестирование и отладку программы.
3. Нарисовать интерфейс программы со спецификацией и записать программные коды с комментариями в отчете по работе.
4. Записать несколько вариантов тестирования программы.
5. Провести тестирование исполняемого файла.

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 5.3. Визуальное событийно-управляемое программирование

### Лабораторное занятие № 25 Создание процедур обработки событий. Компиляция и запуск приложения

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1. Формулировка задачи:

Написать функцию ввода текстовой строки с редактированием. При вызове функции ей передается длина поля ввода (максимальная длина отображаемой на экране части строки), а также координаты первого символа поля ввода на экране. Поле ввода однострочное, т.е. имеет высоту, равную 1 знакоместу.

Длина вводимой строки может превышать размеры поля ввода. Если в процессе ввода или перемещения курсора по строке текущая позиция ввода выходит за пределы поля, необходимо производить прокрутку строки в окне.

По усмотрению разработчика максимальная длина строки может либо передаваться в функцию, либо определяться объемом доступной динамической памяти. В первом случае буфер под строку резервирует вызывающая функция, во втором – функция ввода.

Функция ввода должна предоставлять следующие возможности по редактированию текста: перемещение курсора, ввод символа, удаление символа над курсором (клавиша Delete), удаление символа перед курсором (клавиша Backspace).

Если часть строки находится за пределами поля, желательно показывать это каким-либо образом (например, ставить выделенные цветом символы «<» и «>» в начале и конце поля ввода, если есть текст за пределами поля с соответствующей стороны).

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

### **Тема 5.4 Разработка оконного приложения**

#### **Лабораторное занятие № 26 Разработка оконного приложения с несколькими формами.**

##### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

##### **Выполнив работу, Вы будете:**

###### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;

- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Разработать многооконное приложение.

-На главную форму приложения поместить компонент TAnimate. Компонент должен реализовать просмотр пользовательской анимации, путь которому прописывается в свойстве FileName.

- На главной форме реализовать прослушивание музыкальной композиции при помощи компонента TMediaPlayer.

-Поместить на форму главное меню, содержащее следующие пункты:

-задание 1

-задание 2

-выход

При выборе пункта меню «Задание 1», необходимо реализовать открытие формы «График» (в модальном режиме).

На форме «График» разместить 2 вкладки:

- «Таблица»- содержит таблицу, содержащее протабулированное значение функции  $y=f(x)$  на отрезке  $[a,b]$ . Количество значений функций равно  $n$ . Шаг вычисляется по формуле  $h=(b-a)/n..$

- «График» - содержит график функции  $y=f(x)$ , начерченный с помощью компонента TChart.

При выборе пункта меню «Задание 2», необходимо открытие формы «Теория» (в немодальном режиме). На форме «Теория» расположить 2 компонента TreeView и WebBrowser. Узлами дерева TreeView являются темы теоретического материала. По щелчку названию темы необходимо осуществить загрузку соответствующей страницы html в компонент WebBrowser. При выборе пункта меню «Выход», необходимо закрыть приложение.

| №   | Функция                     | Интервал          |
|-----|-----------------------------|-------------------|
| 1.  | $Y=2\sin(x/3)$              | $[-\pi/2; \pi/2]$ |
| 2.  | $Y=2\cos(x/4)$              | $[0; 3\pi/2]$     |
| 3.  | $Y= \sin(x)+\cos(x) $       | $[0; \pi]$        |
| 4.  | $Y= \sin(x)-\cos(x) $       | $[0; \pi]$        |
| 5.  | $Y=2\sin(x)+3\cos(x)$       | $[-\pi; \pi]$     |
| 6.  | $Y=\sin(x)+\cos^2(x)$       | $[-\pi; \pi]$     |
| 7.  | $Y=2-\cos(x)$               | $[0; 3\pi/2]$     |
| 8.  | $Y=\sin(\sqrt{2}x)+\cos(x)$ | $[0; 2\pi]$       |
| 9.  | $Y=2\sin(2x)+1$             | $[-\pi/2; \pi/2]$ |
| 10. | $Y=\sin(x)+\cos(x)-1$       | $[-\pi; \pi]$     |

2. Создать внешний вид формы программы
3. Написать программные коды для созданной формы
4. Запустить и отладить программу

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 5.4 Разработка оконного приложения

### Лабораторное занятие № 27 Разработка игрового приложения.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1.Формулировка задачи:

Игра «Угадай карту». Дано пять карт, четыре короля и дама пик, перевернутые «рубашкой вверх». Цель игры: угадать, где находится дама пик за наименьшее число ходов. Щелчок по карте (ход) переворачивает ее. Добавить звуковое сопровождение. (3 балла).

##### 2. Создать внешний вид формы программы

##### 3. Написать программные коды для созданной формы

##### 1.Формулировка задачи:

Игра «Где яйцо». Даны три корзины. В одной из корзин спрятано яйцо. Нужно угадать в какой корзине яйцо. Щелчком мыши корзина переворачивается. Добавить звуковое сопровождение. (3 балла).

##### 2. Создать внешний вид формы программы

##### 3. Написать программные коды для созданной формы

#### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.

- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 5.5 Этапы разработки приложений**

### **Лабораторное занятие № 28 Разработка интерфейса приложения.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

### **Задание:**

#### **1.Формулировка задачи:**

Для программы требуется разработать консольный интерфейс, интерфейс-меню и интерфейс со свободной навигацией. Результатом разработки является текстовое описание запросов, пунктов меню или компонентов интерфейса соответственно.

Разработать описание пользовательского интерфейса для программы «Будильник».

#### **2. Исходными данными для программы являются:**

- время, на которое установлен будильник;
- флажок, показывающий, включен будильник или выключен;
- период повторения будильника: однократное срабатывание, включение каждый день, включение раз в неделю.

#### **3. Создать внешний вид формы программы**

#### **4. Написать программные коды для созданной формы**

### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным

материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 5.5 Этапы разработки приложений**

### **Лабораторное занятие № 29 Тестирование, отладка приложения.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### ***уметь:***

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

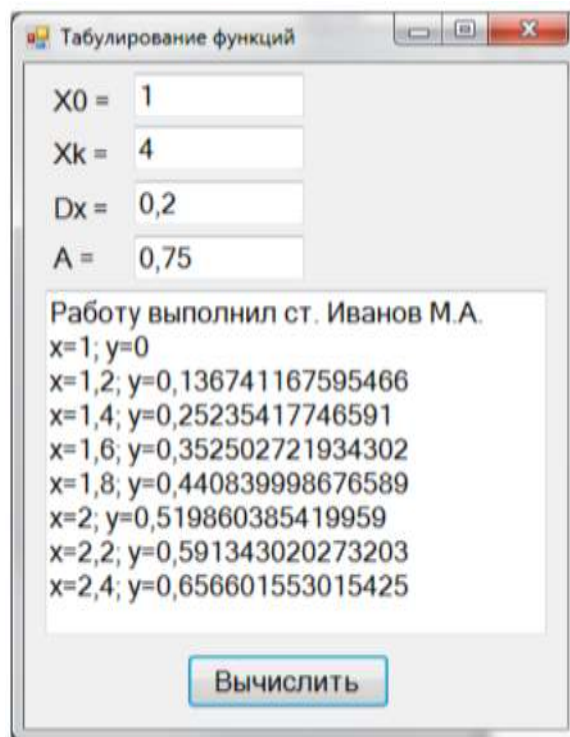
Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Вычислить и вывести на экран таблицу значений функции  $y=a \cdot \ln(x)$  при  $x$ , изменяющемся от  $x_0$  до  $x_k$  с шагом  $dx$ ,  $a$  – константа. Панель диалога представлена на рисунке. После отладки программы следует проверить правильность работы программы с помощью контрольного примера (см. рис.). Установите точку останова на оператор перед циклом и запустите программу. После попадания на точку остановки, выполните пошагово программу и проследите, как меняются все переменные в процессе выполнения.



Составьте программу табулирования функции  $y(x)$ , выведите на экран значения  $x$  и  $y(x)$ . Нужный вариант задания выберите из нижеприведенного списка по указанию преподавателя. Откорректируйте элементы управления в форме в соответствии со своим вариантом задания.

$$1) \quad y = 10^{-2}bc / x + \cos \sqrt{a^3 x},$$

$$x_0 = -1.5; x_k = 3.5; dx = 0.5;$$

$$a = -1.25; b = -1.5; c = 0.75;$$

$$2) \quad y = 1.2(a-b)^3 e^{x^2} + x,$$

$$x_0 = -0.75; x_k = -1.5; dx = -0.05;$$

$$a = 1.5; b = 1.2;$$

2. Создать внешний вид формы программы
3. Написать программные коды для созданной формы

#### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### Ход работы:

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### Форма предоставления результата

- Алгоритм программы.
- Код программы.

#### Критерии оценки:

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные

программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 5.6 Иерархия классов.

### Лабораторное занятие № 30 Классы ООП: виды, назначение, свойства, методы, события.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1.Формулировка задачи:

Построить систему классов для описания плоских геометрических фигур: круга, квадрата, прямоугольника. Предусмотреть методы для создания объектов, перемещения на плоскости, изменения размеров и поворота на заданный угол.

##### 2. Создать внешний вид формы программы

##### 3. Написать программные коды для созданной формы

#### Задача

##### 1.Формулировка задачи:

Построить описание класса, содержащего информацию о почтовом адресе организации. Предусмотреть возможность раздельного изменения составных частей адреса, создания и уничтожения объектов этого класса.

##### 2. Создать внешний вид формы программы

##### 3. Написать программные коды для созданной формы

#### Порядок выполнения работы

- Составить математическую модель задачи.

- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

### **Тема 5.6 Иерархия классов.**

#### **Лабораторное занятие № 31 Объявления класса.**

##### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

##### **Выполнив работу, Вы будете:**

###### **уметь:**

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

##### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;

- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:
  - MS Windows 7 (подписка Imagine Premium)
  - MS Office 2007
  - 7 Zip
  - MS Visual Studio (подписка Imagine Premium)
  - Visual Studio Code
  - Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Определить класс список элементов. В определение класса включить два конструктора для определения списка по его размеру и путем копирования другого списка. Предусмотреть функции инверсии списка (123->321) и поиска подсписка в списке.

2. Создать внешний вид формы программы
3. Написать программные коды для созданной формы

##### **1.Формулировка задачи:**

Определить класс сортированный список элементов. В определение класса включить два конструктора для определения списка по его размеру и путем копирования другого списка. Предусмотреть функции добавления элемента и слияния двух сортированных списков.

2. Создать внешний вид формы программы
3. Написать программные коды для созданной формы

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## **Тема 5.6 Иерархия классов.**

### **Лабораторное занятие № 32 Создание наследованного класса.**

#### **Цель работы:**

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### **Выполнив работу, Вы будете:**

##### ***уметь:***

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### **Материальное обеспечение:**

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### **Задание:**

##### **1.Формулировка задачи:**

Составить описание класса для объектов-векторов, задаваемых координатами концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами.

##### **2. Создать внешний вид формы программы**

##### **3. Написать программные коды для созданной формы**

##### **1.Формулировка задачи:**

Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменения размеров, построения наименьшего прямоугольника, содержащего два заданных прямоугольника, и прямоугольника, являющегося общей частью (пересечением) двух

прямоугольников.

2. Создать внешний вид формы программы

3. Написать программные коды для созданной формы

#### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

#### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

#### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

#### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 5.6 Иерархия классов.

### Лабораторное занятие № 33 Программирование приложений.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1.Формулировка задачи:

Написать Windows-объектов в приложениях на языке C++ приложение, заголовок главного окна которого содержит Ф. И. О., группу и номер варианта. Программа должна динамически отображать информацию об изменении заданной характеристики компьютера в графической форме, используя компонент Chart или средства рисования GDI. График должен отображать значение заданной характеристики с автоматической прокруткой влево по мере появления новых данных (Ctrl+Shift+N приводит к тому же результату), в котором выбираем кроме столбчатых и круговых диаграмм). Промежуток времени между последовательными замерами значения характеристики, а также дополнительные параметры отображения графика в соответствии с заданием должны задаваться в диалоговом окне.

1).. Отображать график загрузки процессора (Ctrl+Shift+N приводит к тому же результату), в котором выбираем в диалоговом окне выбирать номер процессора)

2 Отображать текущий размер заданной папки (Ctrl+Shift+N приводит к тому же результату), в котором выбираем имя папки задается в диалоговом окне)

3 Отображать размер очереди печати заданного в диалоговом окне принтера.

4 Отображать количество запущенных в системе процессов (Ctrl+Shift+N приводит к тому же результату), в котором выбираем в диалоговом окне задавать шаблон имен отслеживаемых процессов).

5 Отображать процент загрузки процессора заданным в диалоговом окне процессом.

6 Отображать объем занимаемой заданным в диалоговом окне процессом виртуальной памяти

7 Отображать количество отправленных пакетов через указанный в диалоговом окне сетевой интерфейс.

8 Отображать частоту обращений на чтение/запись к диску, выбранному в диалоговом окне

9 Отображать количество процессов с различными значениями приоритетов в виде столбчатой диаграммы.

10 Отображать количество работающих или остановленных служб (Ctrl+Shift+N приводит к тому же результату), в котором выбираем переключение типов графиков реализовать в диалоговом окне)

11 Отображать размер файла подкачки и свободной физической памяти (Ctrl+Shift+N приводит к тому же результату), в котором выбираем переключение типов графиков реализовать в диалоговом окне).

12 Отображать количество потоков, запущенных в различных процессах, в виде столбчатой диаграммы (Ctrl+Shift+N приводит к тому же результату), в котором выбираем количество столбцов-объектов в приложениях на языке C++ процессов задавать в диалоговом окне).

13 Отображать объем свободной памяти на логических дисках в виде круговой диаграммы (Ctrl+Shift+N приводит к тому же результату), в котором выбираем в диалоговом окне предоставить возможность включать или отключать в диаграмму информации о сменных носителях).

14 Отображать в виде столбчатой диаграммы объем используемой памяти у 10 процессов с наибольшим значением этого параметра.

15 Отображать динамику использования оперативной памяти в виде круговой диаграммы с секторами, соответствующими: 5 наиболее активно использующих память процессов, объему памяти, используемым стальными процессами, а также объему свободной памяти.

2. Создать внешний вид формы программы

3. Написать программные коды для созданной формы

### **Порядок выполнения работы**

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

### **Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

### **Форма предоставления результата**

- Алгоритм программы.
- Код программы.

### **Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

## Тема 5.6 Иерархия классов.

### Лабораторное занятие № 34 Перегрузка методов.

#### Цель работы:

- Научиться применять базовые понятия теории приближенных методов решения задач на ЭВМ.
- Сформировать представление об алгоритмах решения прикладных задач, научиться написанию и отладки программного кода.
- Выбрать и обосновать наиболее рациональный метод и алгоритм решения задачи;
- Разработать алгоритм для решения прикладных задач;
- Написать программный код и отладить программу.

#### Выполнив работу, Вы будете:

##### *уметь:*

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- выполнять проверку, отладку кода программы;
- оформлять код программы в соответствии со стандартом кодирования;

#### Материальное обеспечение:

- Рабочее место преподавателя: персональный компьютер, проектор, рабочие места обучающихся, доска учебная, учебная мебель;
- Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:

MS Windows 7 (подписка Imagine Premium)

MS Office 2007

7 Zip

MS Visual Studio (подписка Imagine Premium)

Visual Studio Code

Calculate Linux Desktop

#### Задание:

##### 1.Формулировка задачи:

Написать метод  $\min(x, y)$ , находящий минимальное значение из двух чисел. С его помощью найти минимальное значение из четырех чисел a, b, c, d.

##### 2. Создать внешний вид формы программы

##### 3. Написать программные коды для созданной формы

##### 1.Формулировка задачи:

Написать метод  $\max(x, y)$ , находящий максимальное значение из двух чисел. С его помощью найти максимальное значение из четырех чисел a, b, c, d.

##### 2. Создать внешний вид формы программы

##### 3. Написать программные коды для созданной формы

#### Порядок выполнения работы

- Составить математическую модель задачи.
- Выбрать и обосновать наиболее рациональный метод решения задачи;
- Разработать алгоритм для решения задачи.
- Написать и отладить программу.

**Ход работы:**

- Выделить входные и выходные данные
- Построить блок-схему согласно типу алгоритмической конструкции
- Написать программный код на язык программирования и проверить работу кода с входными параметрами.

**Форма предоставления результата**

- Алгоритм программы.
- Код программы.

**Критерии оценки:**

- «Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.