МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФИЛИАЛ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ПРОФЕСИОНАЛЬНОГО ОБРАЗОВАНИЯ

«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НИУ)

В Г. НИЖНЕВАРТОВСКЕ

**А.П. Кротова**

**Программирование на языках ссылочно-объектной модели**

Контрольные задания к выполнению

самостоятельной работы

для студентов направления 230100.62- Информатика и вычислительная техника, профиль: Вычислительные машины, комплексы, системы и сети

Нижневартовск

2014

Содержание

[Самостоятельная работа 3](#_Toc410944953)

[Варианты заданий для самостоятельной работы 4](#_Toc410944954)

[Образец выполнения задания 16](#_Toc410944955)

[Учебно-методическое и информационное обеспечение дисциплины 38](#_Toc410944956)

**Самостоятельная работа**

В рамках самостоятельной работы студентами проектируется и разрабатывается Windows-приложение по варианту на языке C#. После чего студент составляет пояснительную записку о ходе проектирования и результатах разработки, и сдает ее преподавателю в печатном виде.

Пояснительной записки должна состоять из следующих пунктов:

1 ПОСТАНОВКА ЗАДАЧИ

2 РАЗРАБОТКА

2.1 Разработка интерфейса Windows-приложения

2.2 Код программы

3 ТЕСТИРОВАНИЕ ПРОГРАММЫ

ЗАКЛЮЧЕНИЕ

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

**Варианты заданий для самостоятельной работы**

Написать Windows-приложение, заголовок главного окна которого содержит Ф. И. О., группу и номер варианта. В программе должна быть предусмотрена обработка исключений, возникающих из-за ошибочного ввода пользователя.

**Вариант 1**

Создать меню с командами Input, Calc и Exit.

При выборе команды Input открывается диалоговое окно, содержащее:

• три поля типа TextBox для ввода длин трех сторон треугольника;

• группу из двух флажков (Периметр и Площадь) типа CheckBox;

• кнопку типа Button.

Обеспечить возможность:

• ввода длин трех сторон треугольника;

• выбора режима с помощью флажков: подсчет периметра и/или площади треугольника.

При выборе команды Calc открывается диалоговое окно с результатами. При выборе команды Exit приложение завершается.

**Вариант 2**

Создать меню с командами Size, Color, Paint, Quit.

Команда Paint недоступна. При выборе команды Quit приложение завершается.

При выборе команды Size открывается диалоговое окно, содержащее:

• два поля типа TextBox для ввода длин сторон прямоугольника;

• группу из трех флажков (Red, Green, Blue) типа CheckBox;

• кнопку типа Button.

Обеспечить возможность:

• ввода длин сторон прямоугольника в пикселах в поля ввода;

• выбора его цвета с помощью флажков.

После задания параметров команда Paint становится доступной.

При выборе команды Paint в главном окне приложения выводится прямоугольник заданного размера и сочетания цветов или выдается сообщение, если введенные размеры превышают размер окна.

**Вариант 3**

Создать меню с командами Input, Work, Exit.

При выборе команды Exit приложение завершает работу. При выборе команды Input открывается диалоговое окно, содержащее:

• три поля ввода типа TextBox с метками Radius, Height, Density;

• группу из двух флажков (Volume, Mass) типа CheckBox;

• кнопку типа Button.

Обеспечить возможность:

• ввода радиуса, высоты и плотности конуса;

• выбора режима с помощью флажков: подсчет объема и/или массы конуса.

При выборе команды Work открывается окно сообщений с результатами.

**Вариант 4**

Создать меню с командами Input, Calc, Draw, Exit.

При выборе команды Exit приложение завершает работу. При выборе команды Input открывается диалоговое окно, содержащее:

• поле ввода типа TextBox с меткой Radius;

• группу из двух флажков (Square, Length) типа CheckBox;

• кнопку типа Button.

Обеспечить возможность:

• ввода радиуса окружности;

• выбора режима с помощью флажков: подсчет площади круга (Square) и/или длины окружности (Length).

При выборе команды Calc открывается окно сообщений с результатами. При выборе команды Draw в центре главного окна выводится круг введенного радиуса или выдается сообщение, что рисование невозможно (если диаметр превышает размеры рабочей области).

**Вариант 5**

Создать меню с командами input, Calc, About.

При выборе команды About открывается окно с информацией о разработчике.

При выборе команды Input открывается диалоговое окно, содержащее:

• три поля ввода типа TextBox с метками Number 1, Number 2, Number 3;

• группу из двух флажков (Summ, Least multiple) типа CheckBox;

• кнопку типа Button.

Обеспечить возможность ввода трех чисел и выбора режима вычислений с помощью флажков: подсчет суммы трех чисел (Summ) и/или наименьшего общего кратного двух первых чисел (Least multiple). При выборе команды Calc открывается диалоговое окно с результатами.

**Вариант 6**

Создать меню с командами Input, Calc, Quit.

Команда Calc недоступна. При выборе команды Quit приложение завершается.

При выборе команды Input открывается диалоговое окно, содержащее:

• два поля ввода типа TextBox с метками Number 1, Number 2;

• группу из трех флажков (Summa, Max divisor, Multiply) типа CheckBox;

• кнопку типа Button.

Обеспечить возможность:

• ввода двух чисел;

• выбора режима вычислений с помощью флажков (можно вычислять в любой комбинации такие величины, как сумма, наибольший общий делитель и произведение двух чисел).

При выборе команды Calc открывается окно сообщений с результатами.

**Вариант 7**

Создать меню с командами Begin, Help, About.

При выборе команды About открывается окно с информацией о разработчике.

При выборе команды Begin открывается диалоговое окно, содержащее:

• поле ввода типа TextBox с меткой input;

• метку типа Label для вывода результата;

• группу из трех переключателей (2, 8, 16) типа RadioButton;

• две кнопки типа Button — Do и ОК.

Обеспечить возможность:

• ввода числа в десятичной системе в поле input;

• выбора режима преобразования с помощью переключателей: перевод в двоичную, восьмеричную или шестнадцатеричную систему счисления.

При щелчке на кнопке Do должен появляться результат перевода.

**Вариант 8**

Создать меню с командами Input color, Change, Exit, Help.

При выборе команды Exit приложение завершает работу. При выборе команды Input color открывается диалоговое окно, содержащее:

• три поля ввода типа TextBox с метками Red, Green, Blue;

• группу из двух флажков (Left, Right) типа CheckBox;

• кнопку типа Button.

Обеспечить возможность ввода RGB-составляющих цвета. При выборе команды Change цвет главного окна изменяется на заданный (левая, правая или обе половины окна в зависимости от установки флажков).

**Вариант 9**

Создать меню с командами Input size, Choose, Change, Exit.

При выборе команды Exit приложение завершает работу. Команда Change недоступна.

При выборе команды Input size открывается диалоговое окно, содержащее:

• два поля ввода типа TextBox с метками Size х, Size у;

• кнопку типа Button.

При выборе команды Choose открывается диалоговое окно, содержащее:

• группу из двух переключателей (Increase, Decrease) типа RadloButton;

• кнопку типа Button.

Обеспечить возможность ввода значений в поля Size х и Size у. Значения интерпретируются как количество пикселов, на которое надо изменить размеры главного окна (увеличить или уменьшить в зависимости от положения переключателей).

После ввода значений команда Change становится доступной. При выборе этой команды размеры главного окна увеличиваются или уменьшаются на введенное количество пикселов.

**Вариант 10**

Создать меню с командами Begin, Work, About.

При выборе команды About открывается окно с информацией о разработчике.

При выборе команды Begin открывается диалоговое окно, содержащее:

• поле ввода типа TextBox с меткой Input word;

• группу из двух переключателей (Upper case, Lower case) типа RadioButton;

• кнопку типа Button.

Обеспечить возможность ввода слова и выбора режима перевода в верхний или нижний регистр в зависимости от положения переключателей. При выборе команды Work открывается диалоговое окно с результатом перевода.

**Вариант 11**

Создать меню с командами Input color, Change, Clear.

При выборе команды Input color открывается диалоговое окно, содержащее:

• группу из двух флажков (Up, Down) типа CheckBox;

• группу из трех переключателей (Red, Green, Blue) типа RadioButton;

• кнопку типа Button.

Обеспечить возможность:

• выбора цвета с помощью переключателей;

• ввода режима, определяющего, какая область закрашивается: все окно, его верхняя или нижняя половина.

При выборе команды Change цвет главного окна изменяется на заданный (верхняя, нижняя или обе половины в зависимости от введенного режима). При выборе команды Clear восстанавливается первоначальный цвет окна.

**Вариант 12**

Создать меню с командами Translate, Help, About, Exit.

При выборе команды Exit приложение завершает работу. При выборе команды Translate открывается диалоговое окно, содержащее:

• поле ввода типа TextBox с меткой Binary number;

• поле ввода типа TextBox для вывода результата (read-only);

• группу из трех переключателей (8, 10, 16) типа RadioButton;

• кнопку Do типа Button.

Обеспечить возможность:

• ввода числа в двоичной системе в поле Binary number;

• выбора режима преобразования с помощью переключателей: перевод в восьмеричную, десятичную или шестнадцатеричную систему счисления.

При щелчке на кнопке Do должен появляться результат перевода.

**Вариант 13**

Создать меню с командами Reverse, About, Exit.

При выборе команды About открывается окно с информацией о разработчике.

При выборе команды Reverse открывается диалоговое окно, содержащее:

• поле ввода типа TextBox с меткой Input;

• группу из двух переключателей (Upper case, Reverse) типа CheckBox;

• кнопку ОК типа Button.

Обеспечить возможность ввода фразы и выбора режима: перевод в верхний регистр и/или изменение порядка следования символов на обратный в зависимости от состояния переключателей. Результат преобразования выводится в исходное поле ввода.

**Вариант 14**

Создать меню с командами Input, Show и Exit.

При выборе команды Exit приложение завершает работу. При выборе команды Input открывается диалоговое окно вида:



Обеспечивается возможность ввода координат двух точек и выбора режима с помощью флажков length и koef: подсчет длины отрезка, соединяющего эти точки, и/или углового коэффициента.

При выборе команды Show открывается окно сообщений с результатами подсчета.

**Вариант 15**

Создать меню с командами Input, About и Exit.

При выборе команды Exit приложение завершает работу. При выборе команды About открывается окно с информацией о разработчике. При выборе команды Input открывается диалоговое окно вида:



Обеспечивается возможность ввода суммы в рублях и перевода ее в евро и доллары по обычному или льготному курсу. Поля Euro и $ доступны только для чтения.

**Вариант 16**

Создать меню с командами Begin, Work, About.

При выборе команды About открывается окно с информацией о разработчике.

При выборе команды Begin открывается диалоговое окно, содержащее:

• два поля ввода типа TextBox;

• группу из двух переключателей (First letter, All letters) типа RadioButton;

• кнопку типа Button.

Обеспечить возможность ввода предложения и выбора режима его преобразования: либо начинать с прописной буквы каждое слово (First letter), либо перевести все буквы в верхний регистр (All letters). При выборе команды Work открывается диалоговое окно с результатом преобразования.

**Вариант 17**

Написать анализатор текстовых файлов, выводящий информацию о количестве слов в тексте, а также статистическую информацию о введенной пользователем букве.

Создать следующую систему меню:

• Файл

- Загрузить текст

- Выход

• Анализ

- Количество слов

- Повторяемость буквы

При выборе файла для загрузки использовать объект типа OpenFileDialog. При выборе команды Количество слов программа должна вывести в окно сообщений количество слов в тексте.

При выборе команды Повторяемость буквы программа предлагает пользователю ввести букву, а затем выводит количество ее повторений без учета регистра в окно сообщений.

**Вариант 18**

Создать редактор текстовых файлов с возможностью сохранения текста в формате HTML. Создать следующую систему меню:

• Файл

- Загрузить текст

- Сохранить как текст

- Сохранить как HTML

• Выход

При выборе файла для загрузки использовать объект OpenFileDialog. При выборе файла для сохранения использовать объект SaveFileDlalog. Для редактирования текста использовать объект Memo.

При сохранении текста в формате HTML текст записывать в файл с заменой:

• всех пробелов на символы &nbsp;;

• всех символов перевода строки на символы <BR>;

• всех символов < на символы &lt;;

• всех символов > На символы &gt:;

• всех символов & на символы &атр;;

• всех символов " (двойные кавычки) на символы Squot;.

**Вариант 19**

Создать меню с командами Input, Draw, Clear.

При выборе команды Input открывается диалоговое окно, содержащее:

• четыре поля для ввода координат двух точек;

• группу из трех переключателей (Red, Green, Blue) типа RadioButton;

• кнопку типа Button.

При выборе команды Draw в главное окно выводится отрезок прямой выбранного цвета с координатами концов отрезка, заданными в диалоговом окне. При выборе команды Clear отрезок стирается.

**Вариант 20**

Создать меню с командами Input, Change, Exit.

При выборе команды Exit приложение завершает работу. Команда Change недоступна.

В центре главного окна выведен квадрат размером 100 х 100 пикселов.

При выборе команды Input открывается диалоговое окно, содержащее:

• два поля ввода типа TextBox с метками Size х, Size у;

• группу из двух переключателей (Increase, Decrease) типа RadioButton;

• кнопку типа Button.

Обеспечить возможность ввода значений в поля Size х и Size у. Значения интерпретируются как количество пикселов, на которое надо изменить размеры квадрата, выведенного в главное окно (увеличить или уменьшить в зависимости от положения переключателей).

После ввода значений команда Change становится доступной. При выборе этой команды размеры квадрата увеличиваются или уменьшаются на введенное количество пикселов. Если квадрат выходит за пределы рабочей области окна, выдается сообщение.

**Вариант 21**

Написать Windows-приложение, которое по заданным в файле исходным данным выводит информацию о компьютерах.

Создать меню с командами Choose, Show, Quit.

Команда Show недоступна. Команда Quit завершает работу приложения.

При запуске приложения из файла читаются исходные данные. Файл необходимо сформировать самостоятельно. Каждая строка файла содержит тип компьютера, цену (price) и емкость жесткого диска (hard drive).

При выборе команды Choose открывается диалоговое окно, содержащее:

• поле типа TextBox для ввода минимальной емкости диска;

• поле типа TextBox для ввода максимальной приемлемой цены;

• группу из двух переключателей (Hard drive, Price) типа RadioButton;

• OK, Cancel — кнопки типа Button.

После ввода всех данных команда меню Show становится доступной. Команда Show открывает диалоговое окно, содержащее список компьютеров, удовлетворяющий введенным ограничениям и упорядоченный по отмеченной характеристике.

**Образец выполнения задания**

**1 ПОСТАНОВКА ЗАДАЧИ**

Написать Windows-приложение, заголовок главного окна которого содержит Ф. И. О., группу и номер варианта. В программе должна быть предусмотрена обработка исключений, возникающих из-за ошибочного ввода пользователя.

Написать Windows-приложение, которое по заданным в файле исходным данным выводит информацию о компьютерах.

Создать меню с командами Choose, Show, Quit.

Команда Show недоступна. Команда Quit завершает работу приложения.

При запуске приложения из файла читаются исходные данные. Файл необходимо сформировать самостоятельно. Каждая строка файла содержит тип компьютера, цену (price) и емкость жесткого диска (hard drive).

При выборе команды Choose открывается диалоговое окно, содержащее:

• поле типа TextBox для ввода минимальной емкости диска;

• поле типа TextBox для ввода максимальной приемлемой цены;

• группу из двух переключателей (Hard drive, Price) типа RadioButton;

• OK, Cancel — кнопки типа Button.

После ввода всех данных команда меню Show становится доступной. Команда Show открывает диалоговое окно, содержащее список компьютеров, удовлетворяющий введенным ограничениям и упорядоченный по отмеченной характеристике.

**2 РАЗРАБОТКА**

**2.1 Разработка интерфейса Windows-приложения**

Для основной формы проекта задаем следующие свойства:

|  |  |
| --- | --- |
| **Свойства** | **Значения** |
| (Name) | Form1 |
| Size | 700; 300 |
| StartPosition | CenterScreen |
| Text | Иванов Иван Иванович, НвФл-323, вариант 21 |

Добавляем на форму элемент управления «MenuStrip» (меню). В это меню добавляем 3 пункта: Choose, Show, Quit.

Добавляем на форму элемент интерфейса «Panel» (панель). Для данного элемента задаем следующие свойства:

|  |  |
| --- | --- |
| **Свойства** | **Значения** |
| (Name) | panel1 |
| Anchor | Top, Left, Right |
| BackColor | ActiveCaption |
| Location | 0; 30 |
| Size | 680; 25 |

На панели размещаем 3 элемента интерфейса «Label» (метка) со следующими свойствами:

|  |  |  |  |
| --- | --- | --- | --- |
| **Свойства** | **Значения для 1-й метки** | **Значения для 2-й метки** | **Значения для 3-й метки** |
| (Name) | Label1 | Label2 | Label3 |
| Location | 5; 5 | 160; 5 | 280; 5 |
| Text | Min hard drive: | Max price: | Order by: |

Затем на этой же панели размещаем еле 3 метки со следующими свойствами:

|  |  |  |  |
| --- | --- | --- | --- |
| **Свойства** | **Значения для 4-й метки** | **Значения для 5-й метки** | **Значения для 6-й метки** |
| (Name) | LMinHardDrive | LMaxPrice | LOrderBy |
| Location | 80; 5 | 215; 5 | 330; 5 |
| Text | None | None | None |

На форме (вне панели) располагаем элемент управления «TextBox» (поле ввода) со следующими свойствами:

|  |  |
| --- | --- |
| **Свойства** | **Значения** |
| (Name) | textBox1 |
| Anchor | Top, Bottom, Left, Right |
| Enabled | True |
| Location | 0; 58 |
| Multiline | True |
| ReadOnly | True |
| ScrollBars | Both |
| Size | 680; 204 |
| Text |  |

В результате получится форма с интерфейсом, изображенным на рисунке:



Теперь добавим в проект еще одну форму, которая будет появляться после щелчка по пункту меню «Choose» в главной форме:



Для новой формы задаем следующие свойства:

|  |  |
| --- | --- |
| **Свойства** | **Значения** |
| (Name) | Form2 |
| Size | 220; 230 |
| StartPosition | CenterParent |
| Text | Choose |

 Затем на этой форме разместим две метки (Label) со следующими свойствами:

|  |  |  |
| --- | --- | --- |
| **Свойства** | **Значения для первой метки** | **Значения для второй метки** |
| (Name) | label1 | label2 |
| Location | 3; 23 | 3; 48 |
| Text | Min hard drive: | Max price: |

На той же форме расположим два поля для ввода (TextBox) со следующими свойствами:

|  |  |  |
| --- | --- | --- |
| **Свойства** | **Значения для первого поля** | **Значения для второго поля** |
| (Name) | TBMinHardDrive | TBMaxPrice |
| Anchor | Top, Left | Top, Left |
| Enabled | True | True |
| Location | 80; 20 | 80; 45 |
| Multiline | False | False |
| ReadOnly | False | False |
| ScrollBars | None | None |
| Size | 120; 20 | 120; 20 |
| Text |  |  |

Разместим элемент управления «GroupBox» (панель группировки) со следующими свойствами:

|  |  |
| --- | --- |
| **Свойства** | **Значения** |
| (Name) | GBOrder |
| Anchor | Top, Left |
| Location | 6; 70 |
| Size | 194; 67 |

В группирующей панели разместим два переключателя (RadioButton):

|  |  |  |
| --- | --- | --- |
| **Свойства** | **Значения для первого переключателя** | **Значения для второго переключателя** |
| (Name) | RBHardDrive | RBPrice |
| Checked | False | False |
| Location | 6; 18 | 6; 41 |

Расположим на форме две кнопки (Button) со следующими свойствами:

|  |  |  |
| --- | --- | --- |
| **Свойства** | **Значения для первой кнопки** | **Значения для второй кнопки** |
| (Name) | ButtonOK | ButtonCancel |
| DialogResult | OK | Cancel |
| Location | 12; 156 | 117; 156 |
| Size | 75; 23 | 75; 23 |
| Text | OK | Cancel |

В результате получится форма с интерфейсом, изображенным на рисунке:



Добавим в проект еще одну форму, которая будет появляться после щелчка по пункту меню «Show» в главной форме:

|  |  |
| --- | --- |
| **Свойства** | **Значения** |
| (Name) | Form3 |
| Size | 300; 230 |
| StartPosition | CenterParent |
| Text | Result |



На этой форме расположим поле для ввода (TextBox) со следующими свойствами:

|  |  |
| --- | --- |
| **Свойства** | **Значения для первого поля** |
| (Name) | textBox1 |
| Anchor | Top, Bottom, Left, Right |
| Enabled | True |
| Location | 1; 2 |
| Multiline | True |
| ReadOnly | True |
| ScrollBars | Both |
| Size | 280; 190 |
| Text |  |

Далее добавим в проект форму, которая будет выводить пользователю сообщения при исключениях:

|  |  |
| --- | --- |
| **Свойства** | **Значения** |
| (Name) | Form4 |
| Size | 432; 166 |
| StartPosition | CenterParent |
| Text | Exeption |



В этой форме разместим метку со свойствами:

|  |  |
| --- | --- |
| **Свойства** | **Значения для метки** |
| (Name) | label1 |
| Anchor | Top, Left, Right |
| AutoSize | False |
| Location | 12; 9 |
| Size | 392; 86 |
| Text | label1 |

и кнопку:

|  |  |
| --- | --- |
| **Свойства** | **Значения для кнопки** |
| (Name) | button1 |
| DialogResult | OK |
| Location | 169; 98 |
| Size | 75; 23 |
| Text | OK |

**2.2 Код программы**

 Код программы в файле «Form1.Designer.cs»:

|  |
| --- |
| namespace WindowsFormsApplication1{ partial class Form1 { /// <summary> /// Требуется переменная конструктора. /// </summary> private System.ComponentModel.IContainer components = null; /// <summary> /// Освободить все используемые ресурсы. /// </summary> /// <param name="disposing">истинно, если управляемый ресурс должен быть удален; иначе ложно.</param> protected override void Dispose(bool disposing) { if (disposing && (components != null)) { components.Dispose(); } base.Dispose(disposing); } #region Код, автоматически созданный конструктором форм Windows /// <summary> /// Обязательный метод для поддержки конструктора - не изменяйте /// содержимое данного метода при помощи редактора кода. /// </summary> private void InitializeComponent() { this.Label1 = new System.Windows.Forms.Label(); this.menuStrip1 = new System.Windows.Forms.MenuStrip(); this.chooseToolStripMenuItem = new System.Windows.Forms.ToolStripMenuItem(); this.showToolStripMenuItem = new System.Windows.Forms.ToolStripMenuItem(); this.quitToolStripMenuItem = new System.Windows.Forms.ToolStripMenuItem(); this.panel1 = new System.Windows.Forms.Panel(); this.LOrderBy = new System.Windows.Forms.Label(); this.Label3 = new System.Windows.Forms.Label(); this.LMaxPrice = new System.Windows.Forms.Label(); this.Label2 = new System.Windows.Forms.Label(); this.LMinHardDrive = new System.Windows.Forms.Label(); this.textBox1 = new System.Windows.Forms.TextBox(); this.menuStrip1.SuspendLayout(); this.panel1.SuspendLayout(); this.SuspendLayout(); //  // Label1 //  this.Label1.AutoSize = true; this.Label1.Location = new System.Drawing.Point(5, 5); this.Label1.Name = "Label1"; this.Label1.Size = new System.Drawing.Size(77, 13); this.Label1.TabIndex = 6; this.Label1.Text = "Min hard drive:"; //  // menuStrip1 //  this.menuStrip1.Items.AddRange(new System.Windows.Forms.ToolStripItem[] { this.chooseToolStripMenuItem, this.showToolStripMenuItem, this.quitToolStripMenuItem}); this.menuStrip1.Location = new System.Drawing.Point(0, 0); this.menuStrip1.Name = "menuStrip1"; this.menuStrip1.Size = new System.Drawing.Size(684, 24); this.menuStrip1.TabIndex = 8; this.menuStrip1.Text = "menuStrip1"; //  // chooseToolStripMenuItem //  this.chooseToolStripMenuItem.Name = "chooseToolStripMenuItem"; this.chooseToolStripMenuItem.Size = new System.Drawing.Size(59, 20); this.chooseToolStripMenuItem.Text = "Choose"; this.chooseToolStripMenuItem.Click += new System.EventHandler(this.chooseToolStripMenuItem\_Click); //  // showToolStripMenuItem //  this.showToolStripMenuItem.Enabled = false; this.showToolStripMenuItem.Name = "showToolStripMenuItem"; this.showToolStripMenuItem.Size = new System.Drawing.Size(48, 20); this.showToolStripMenuItem.Text = "Show"; this.showToolStripMenuItem.Click += new System.EventHandler(this.showToolStripMenuItem\_Click); //  // quitToolStripMenuItem //  this.quitToolStripMenuItem.Name = "quitToolStripMenuItem"; this.quitToolStripMenuItem.Size = new System.Drawing.Size(42, 20); this.quitToolStripMenuItem.Text = "Quit"; this.quitToolStripMenuItem.Click += new System.EventHandler(this.quitToolStripMenuItem\_Click); //  // panel1 //  this.panel1.Anchor = ((System.Windows.Forms.AnchorStyles)(((System.Windows.Forms.AnchorStyles.Top | System.Windows.Forms.AnchorStyles.Left) | System.Windows.Forms.AnchorStyles.Right))); this.panel1.BackColor = System.Drawing.SystemColors.ActiveCaption; this.panel1.Controls.Add(this.LOrderBy); this.panel1.Controls.Add(this.Label3); this.panel1.Controls.Add(this.LMaxPrice); this.panel1.Controls.Add(this.Label2); this.panel1.Controls.Add(this.LMinHardDrive); this.panel1.Controls.Add(this.Label1); this.panel1.Location = new System.Drawing.Point(0, 30); this.panel1.Name = "panel1"; this.panel1.Size = new System.Drawing.Size(680, 25); this.panel1.TabIndex = 9; //  // LOrderBy //  this.LOrderBy.AutoSize = true; this.LOrderBy.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204))); this.LOrderBy.Location = new System.Drawing.Point(330, 5); this.LOrderBy.Name = "LOrderBy"; this.LOrderBy.Size = new System.Drawing.Size(37, 13); this.LOrderBy.TabIndex = 11; this.LOrderBy.Text = "None"; //  // Label3 //  this.Label3.AutoSize = true; this.Label3.Location = new System.Drawing.Point(280, 5); this.Label3.Name = "Label3"; this.Label3.Size = new System.Drawing.Size(53, 13); this.Label3.TabIndex = 10; this.Label3.Text = "Order by: "; //  // LMaxPrice //  this.LMaxPrice.AutoSize = true; this.LMaxPrice.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204))); this.LMaxPrice.Location = new System.Drawing.Point(215, 5); this.LMaxPrice.Name = "LMaxPrice"; this.LMaxPrice.Size = new System.Drawing.Size(37, 13); this.LMaxPrice.TabIndex = 9; this.LMaxPrice.Text = "None"; //  // Label2 //  this.Label2.AutoSize = true; this.Label2.Location = new System.Drawing.Point(160, 5); this.Label2.Name = "Label2"; this.Label2.Size = new System.Drawing.Size(59, 13); this.Label2.TabIndex = 8; this.Label2.Text = "Max price: "; //  // LMinHardDrive //  this.LMinHardDrive.AutoSize = true; this.LMinHardDrive.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204))); this.LMinHardDrive.Location = new System.Drawing.Point(80, 5); this.LMinHardDrive.Name = "LMinHardDrive"; this.LMinHardDrive.Size = new System.Drawing.Size(37, 13); this.LMinHardDrive.TabIndex = 7; this.LMinHardDrive.Text = "None"; //  // textBox1 //  this.textBox1.Anchor = ((System.Windows.Forms.AnchorStyles)((((System.Windows.Forms.AnchorStyles.Top | System.Windows.Forms.AnchorStyles.Bottom) | System.Windows.Forms.AnchorStyles.Left) | System.Windows.Forms.AnchorStyles.Right))); this.textBox1.Location = new System.Drawing.Point(0, 58); this.textBox1.Multiline = true; this.textBox1.Name = "textBox1"; this.textBox1.ReadOnly = true; this.textBox1.ScrollBars = System.Windows.Forms.ScrollBars.Both; this.textBox1.Size = new System.Drawing.Size(680, 204); this.textBox1.TabIndex = 10; //  // Form1 //  this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F); this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font; this.ClientSize = new System.Drawing.Size(684, 262); this.Controls.Add(this.textBox1); this.Controls.Add(this.panel1); this.Controls.Add(this.menuStrip1); this.MainMenuStrip = this.menuStrip1; this.Name = "Form1"; this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen; this.Text = "Иванов Иван Иванович, НвФл-323, вариант 21"; this.menuStrip1.ResumeLayout(false); this.menuStrip1.PerformLayout(); this.panel1.ResumeLayout(false); this.panel1.PerformLayout(); this.ResumeLayout(false); this.PerformLayout(); } #endregion private System.Windows.Forms.Label Label1; private System.Windows.Forms.MenuStrip menuStrip1; private System.Windows.Forms.ToolStripMenuItem chooseToolStripMenuItem; private System.Windows.Forms.ToolStripMenuItem showToolStripMenuItem; private System.Windows.Forms.ToolStripMenuItem quitToolStripMenuItem; private System.Windows.Forms.Panel panel1; private System.Windows.Forms.TextBox textBox1; private System.Windows.Forms.Label LOrderBy; private System.Windows.Forms.Label Label3; private System.Windows.Forms.Label LMaxPrice; private System.Windows.Forms.Label Label2; private System.Windows.Forms.Label LMinHardDrive; private string HeadStr; private struct Computer { public string CompType; public double Price; public double HardDrive; }; private Computer[] mas; }} |

Код в файле «Form1.cs»:

|  |
| --- |
| using System;using System.Collections.Generic;using System.ComponentModel;using System.Data;using System.Drawing;using System.Linq;using System.Text;using System.Windows.Forms;using System.IO;namespace WindowsFormsApplication1{ public partial class Form1 : Form { public Form1() { InitializeComponent(); LMinHardDrive.Text= "None"; LMaxPrice.Text = "None"; LOrderBy.Text = "None"; try { StreamReader f = new StreamReader("DB\_PC.txt");  string s = f.ReadToEnd(); string[] StrLines = s.Split(new char[] { '\n' }); HeadStr = StrLines[0]; int n = StrLines.Count()-1; mas = new Computer[n]; for (int i = 0; i < n; i++) { string[] Words = StrLines[i+1].Split(new char[] { '\t' }); mas[i].CompType = Words[0]; mas[i].Price = Convert.ToDouble(Words[1]); mas[i].HardDrive = Convert.ToDouble(Words[2]);  } f.Close(); textBox1.Text = s; } catch (Exception ex) { Form4 FormExeption = new Form4(); FormExeption.label1.Text = ex.Message; FormExeption.ShowDialog(); } } private void quitToolStripMenuItem\_Click(object sender, EventArgs e) { Form1.ActiveForm.Close(); } private void chooseToolStripMenuItem\_Click(object sender, EventArgs e) { Form2 FormChoose = new Form2(); // создание экземпляра класса окна FormChoose.SMinHardDrive = LMinHardDrive.Text; FormChoose.SMaxPrice = LMaxPrice.Text; FormChoose.SOrderBy = LOrderBy.Text; if (FormChoose.ShowDialog() == DialogResult.OK) // отображение окна { LMinHardDrive.Text = FormChoose.SMinHardDrive; LMaxPrice.Text = FormChoose.SMaxPrice; LOrderBy.Text = FormChoose.SOrderBy; } showToolStripMenuItem.Enabled = (!((LMinHardDrive.Text == "None") && (LMaxPrice.Text == "None") && (LOrderBy.Text == "None"))); } private void showToolStripMenuItem\_Click(object sender, EventArgs e) { try { Computer[] newmas = new Computer[mas.Count()]; int i=0, j; bool Flag = false; for (j = 0; j < mas.Count(); j++) { if (LMinHardDrive.Text != "None" && LMaxPrice.Text != "None") Flag = (mas[j].Price <= Convert.ToDouble(LMaxPrice.Text) && mas[j].HardDrive >= Convert.ToDouble(LMinHardDrive.Text));  else if (LMinHardDrive.Text != "None" && LMaxPrice.Text == "None") Flag = (mas[j].HardDrive >= Convert.ToDouble(LMinHardDrive.Text)); else if (LMinHardDrive.Text == "None" && LMaxPrice.Text != "None") Flag = (mas[j].Price <= Convert.ToDouble(LMaxPrice.Text)); else Flag = false; if (Flag) { newmas[i] = mas[j]; i++; } } Computer temp = new Computer(); for (int k = 0; k < i; k++) for (j = 0; j < i-1; j++) if (LOrderBy.Text == "Price" && newmas[j].Price > newmas[j + 1].Price || LOrderBy.Text == "Hard drive" && newmas[j].HardDrive > newmas[j + 1].HardDrive) { temp = newmas[j]; newmas[j] = newmas[j + 1]; newmas[j + 1]=temp; } Form3 FormResult = new Form3(); // создание экземпляра класса окна FormResult.textBox1.Text = HeadStr; for (j = 0; j < i; j++) FormResult.textBox1.Text = FormResult.textBox1.Text + "\r\n" + newmas[j].CompType + "\t" + newmas[j].Price + "\t" + newmas[j].HardDrive; FormResult.ShowDialog(); } catch (Exception ex) { Form4 FormExeption = new Form4(); FormExeption.label1.Text = ex.Message; FormExeption.ShowDialog(); } } }} |

 Код программы в файле «Form2.Designer.cs»:

|  |
| --- |
| namespace WindowsFormsApplication1{ partial class Form2 { /// <summary> /// Требуется переменная конструктора. /// </summary> private System.ComponentModel.IContainer components = null; /// <summary> /// Освободить все используемые ресурсы. /// </summary> /// <param name="disposing">истинно, если управляемый ресурс должен быть удален; иначе ложно.</param> protected override void Dispose(bool disposing) { if (disposing && (components != null)) { components.Dispose(); } base.Dispose(disposing); } #region Код, автоматически созданный конструктором форм Windows /// <summary> /// Обязательный метод для поддержки конструктора - не изменяйте /// содержимое данного метода при помощи редактора кода. /// </summary> private void InitializeComponent() { this.panel1 = new System.Windows.Forms.Panel(); this.GBOrder = new System.Windows.Forms.GroupBox(); this.RBHardDrive = new System.Windows.Forms.RadioButton(); this.RBPrice = new System.Windows.Forms.RadioButton(); this.ButtonCancel = new System.Windows.Forms.Button(); this.label2 = new System.Windows.Forms.Label(); this.ButtonOK = new System.Windows.Forms.Button(); this.label1 = new System.Windows.Forms.Label(); this.TBMaxPrice = new System.Windows.Forms.TextBox(); this.TBMinHardDrive = new System.Windows.Forms.TextBox(); this.panel1.SuspendLayout(); this.GBOrder.SuspendLayout(); this.SuspendLayout(); //  // panel1 //  this.panel1.Anchor = ((System.Windows.Forms.AnchorStyles)((((System.Windows.Forms.AnchorStyles.Top | System.Windows.Forms.AnchorStyles.Bottom) | System.Windows.Forms.AnchorStyles.Left) | System.Windows.Forms.AnchorStyles.Right))); this.panel1.BackColor = System.Drawing.SystemColors.ActiveCaption; this.panel1.Controls.Add(this.GBOrder); this.panel1.Controls.Add(this.ButtonCancel); this.panel1.Controls.Add(this.label2); this.panel1.Controls.Add(this.ButtonOK); this.panel1.Controls.Add(this.label1); this.panel1.Controls.Add(this.TBMaxPrice); this.panel1.Controls.Add(this.TBMinHardDrive); this.panel1.Location = new System.Drawing.Point(0, 1); this.panel1.Name = "panel1"; this.panel1.Size = new System.Drawing.Size(204, 190); this.panel1.TabIndex = 10; //  // GBOrder //  this.GBOrder.Controls.Add(this.RBHardDrive); this.GBOrder.Controls.Add(this.RBPrice); this.GBOrder.Location = new System.Drawing.Point(6, 70); this.GBOrder.Name = "GBOrder"; this.GBOrder.Size = new System.Drawing.Size(194, 67); this.GBOrder.TabIndex = 3; this.GBOrder.TabStop = false; this.GBOrder.Text = "Order by"; //  // RBHardDrive //  this.RBHardDrive.AutoSize = true; this.RBHardDrive.Location = new System.Drawing.Point(6, 18); this.RBHardDrive.Name = "RBHardDrive"; this.RBHardDrive.Size = new System.Drawing.Size(74, 17); this.RBHardDrive.TabIndex = 3; this.RBHardDrive.TabStop = true; this.RBHardDrive.Text = "Hard drive"; this.RBHardDrive.UseVisualStyleBackColor = true; //  // RBPrice //  this.RBPrice.AutoSize = true; this.RBPrice.Location = new System.Drawing.Point(6, 41); this.RBPrice.Name = "RBPrice"; this.RBPrice.Size = new System.Drawing.Size(49, 17); this.RBPrice.TabIndex = 0; this.RBPrice.TabStop = true; this.RBPrice.Text = "Price"; this.RBPrice.UseVisualStyleBackColor = true; //  // ButtonCancel //  this.ButtonCancel.DialogResult = System.Windows.Forms.DialogResult.Cancel; this.ButtonCancel.Location = new System.Drawing.Point(117, 156); this.ButtonCancel.Name = "ButtonCancel"; this.ButtonCancel.Size = new System.Drawing.Size(75, 23); this.ButtonCancel.TabIndex = 5; this.ButtonCancel.Text = "Cancel"; this.ButtonCancel.UseVisualStyleBackColor = true; //  // label2 //  this.label2.AutoSize = true; this.label2.Location = new System.Drawing.Point(3, 48); this.label2.Name = "label2"; this.label2.Size = new System.Drawing.Size(56, 13); this.label2.TabIndex = 7; this.label2.Text = "Max price:"; //  // ButtonOK //  this.ButtonOK.DialogResult = System.Windows.Forms.DialogResult.OK; this.ButtonOK.Location = new System.Drawing.Point(12, 156); this.ButtonOK.Name = "ButtonOK"; this.ButtonOK.Size = new System.Drawing.Size(75, 23); this.ButtonOK.TabIndex = 4; this.ButtonOK.Text = "OK"; this.ButtonOK.UseVisualStyleBackColor = true; this.ButtonOK.Click += new System.EventHandler(this.ButtonOK\_Click); //  // label1 //  this.label1.AutoSize = true; this.label1.Location = new System.Drawing.Point(3, 23); this.label1.Name = "label1"; this.label1.Size = new System.Drawing.Size(77, 13); this.label1.TabIndex = 6; this.label1.Text = "Min hard drive:"; //  // TBMaxPrice //  this.TBMaxPrice.Location = new System.Drawing.Point(80, 45); this.TBMaxPrice.Name = "TBMaxPrice"; this.TBMaxPrice.Size = new System.Drawing.Size(120, 20); this.TBMaxPrice.TabIndex = 1; //  // TBMinHardDrive //  this.TBMinHardDrive.Location = new System.Drawing.Point(80, 20); this.TBMinHardDrive.Name = "TBMinHardDrive"; this.TBMinHardDrive.Size = new System.Drawing.Size(120, 20); this.TBMinHardDrive.TabIndex = 0; //  // Form2 //  this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F); this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font; this.ClientSize = new System.Drawing.Size(204, 192); this.Controls.Add(this.panel1); this.Name = "Form2"; this.StartPosition = System.Windows.Forms.FormStartPosition.CenterParent; this.Text = "Choose"; this.Shown += new System.EventHandler(this.Form2\_Shown); this.panel1.ResumeLayout(false); this.panel1.PerformLayout(); this.GBOrder.ResumeLayout(false); this.GBOrder.PerformLayout(); this.ResumeLayout(false); } #endregion private System.Windows.Forms.Panel panel1; private System.Windows.Forms.GroupBox GBOrder; private System.Windows.Forms.RadioButton RBHardDrive; private System.Windows.Forms.RadioButton RBPrice; private System.Windows.Forms.Button ButtonCancel; private System.Windows.Forms.Label label2; private System.Windows.Forms.Button ButtonOK; private System.Windows.Forms.Label label1; private System.Windows.Forms.TextBox TBMaxPrice; private System.Windows.Forms.TextBox TBMinHardDrive; public string SMinHardDrive, SMaxPrice, SOrderBy; }} |

Код в файле «Form2.cs»:

|  |
| --- |
| using System;using System.Collections.Generic;using System.ComponentModel;using System.Data;using System.Drawing;using System.Linq;using System.Text;using System.Windows.Forms;namespace WindowsFormsApplication1{ public partial class Form2 : Form { public Form2() { InitializeComponent(); }  private void ButtonOK\_Click(object sender, EventArgs e) { SMinHardDrive=TBMinHardDrive.Text; if (SMinHardDrive == "") SMinHardDrive = "None"; SMaxPrice = TBMaxPrice.Text; if (SMaxPrice == "") SMaxPrice = "None"; SOrderBy = "None"; if (RBHardDrive.Checked) SOrderBy = "Hard drive"; else  if (RBPrice.Checked) SOrderBy = "Price"; }  private void Form2\_Shown(object sender, EventArgs e) {  if (SMinHardDrive == "None") TBMinHardDrive.Text = ""; else TBMinHardDrive.Text = SMinHardDrive; if (SMaxPrice == "None") TBMaxPrice.Text = ""; else TBMaxPrice.Text = SMaxPrice; if (SOrderBy == "Hard drive") RBHardDrive.Checked = true; else if (SOrderBy == "Price") RBPrice.Checked = true; } }} |

Код программы в файле «Form3.Designer.cs»:

|  |
| --- |
| namespace WindowsFormsApplication1{ partial class Form3 { /// <summary> /// Требуется переменная конструктора. /// </summary> private System.ComponentModel.IContainer components = null; /// <summary> /// Освободить все используемые ресурсы. /// </summary> /// <param name="disposing">истинно, если управляемый ресурс должен быть удален; иначе ложно.</param> protected override void Dispose(bool disposing) { if (disposing && (components != null)) { components.Dispose(); } base.Dispose(disposing); } #region Код, автоматически созданный конструктором форм Windows /// <summary> /// Обязательный метод для поддержки конструктора - не изменяйте /// содержимое данного метода при помощи редактора кода. /// </summary> private void InitializeComponent() { this.textBox1 = new System.Windows.Forms.TextBox(); this.SuspendLayout(); //  // textBox1 //  this.textBox1.Anchor = ((System.Windows.Forms.AnchorStyles)((((System.Windows.Forms.AnchorStyles.Top | System.Windows.Forms.AnchorStyles.Bottom) | System.Windows.Forms.AnchorStyles.Left) | System.Windows.Forms.AnchorStyles.Right))); this.textBox1.Location = new System.Drawing.Point(1, 2); this.textBox1.Multiline = true; this.textBox1.Name = "textBox1"; this.textBox1.ReadOnly = true; this.textBox1.ScrollBars = System.Windows.Forms.ScrollBars.Both; this.textBox1.Size = new System.Drawing.Size(280, 190); this.textBox1.TabIndex = 0; //  // Form3 //  this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F); this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font; this.ClientSize = new System.Drawing.Size(284, 192); this.Controls.Add(this.textBox1); this.Name = "Form3"; this.StartPosition = System.Windows.Forms.FormStartPosition.CenterParent; this.Text = "Result"; this.ResumeLayout(false); this.PerformLayout(); } #endregion public System.Windows.Forms.TextBox textBox1; }} |

Код в файле «Form3.cs»:

|  |
| --- |
| using System;using System.Collections.Generic;using System.ComponentModel;using System.Data;using System.Drawing;using System.Linq;using System.Text;using System.Windows.Forms;namespace WindowsFormsApplication1{ public partial class Form3 : Form { public Form3() { InitializeComponent(); } }} |

 Код программы в файле «Form4.Designer.cs»:

|  |
| --- |
| namespace WindowsFormsApplication1{ partial class Form4 { /// <summary> /// Требуется переменная конструктора. /// </summary> private System.ComponentModel.IContainer components = null; /// <summary> /// Освободить все используемые ресурсы. /// </summary> /// <param name="disposing">истинно, если управляемый ресурс должен быть удален; иначе ложно.</param> protected override void Dispose(bool disposing) { if (disposing && (components != null)) { components.Dispose(); } base.Dispose(disposing); } #region Код, автоматически созданный конструктором форм Windows /// <summary> /// Обязательный метод для поддержки конструктора - не изменяйте /// содержимое данного метода при помощи редактора кода. /// </summary> private void InitializeComponent() { this.label1 = new System.Windows.Forms.Label(); this.button1 = new System.Windows.Forms.Button(); this.SuspendLayout(); //  // label1 //  this.label1.Anchor = ((System.Windows.Forms.AnchorStyles)(((System.Windows.Forms.AnchorStyles.Top | System.Windows.Forms.AnchorStyles.Left) | System.Windows.Forms.AnchorStyles.Right))); this.label1.Location = new System.Drawing.Point(12, 9); this.label1.Name = "label1"; this.label1.Size = new System.Drawing.Size(392, 86); this.label1.TabIndex = 0; this.label1.Text = "label1"; //  // button1 //  this.button1.DialogResult = System.Windows.Forms.DialogResult.OK; this.button1.Location = new System.Drawing.Point(169, 98); this.button1.Name = "button1"; this.button1.Size = new System.Drawing.Size(75, 23); this.button1.TabIndex = 1; this.button1.Text = "OK"; this.button1.UseVisualStyleBackColor = true; //  // Form4 //  this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F); this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font; this.ClientSize = new System.Drawing.Size(416, 128); this.Controls.Add(this.button1); this.Controls.Add(this.label1); this.Name = "Form4"; this.StartPosition = System.Windows.Forms.FormStartPosition.CenterParent; this.Text = "Exeption"; this.ResumeLayout(false); } #endregion private System.Windows.Forms.Button button1; public System.Windows.Forms.Label label1; }} |

Код в файле «Form4.cs»:

|  |
| --- |
| using System;using System.Collections.Generic;using System.ComponentModel;using System.Data;using System.Drawing;using System.Linq;using System.Text;using System.Windows.Forms;namespace WindowsFormsApplication1{ public partial class Form4 : Form { public Form4() { InitializeComponent(); } }} |

**3 ТЕСТИРОВАНИЕ ПРОГРАММЫ**

 Файл «DB\_PC.txt» с исходными данными:



После запуска программы открывается главная форма, в которой будет отображаться исходное содержимое файла:



По вызову команды «Choose» из главного меню открывается форма ввода параметров выборки и сортировки. Если заполнить данную форму, например, так:



и нажать кнопку «ОК», то в главной форме изменится строка состояния параметров. Если же будет нажата кнопка «Cancel», то в главной форме все останется без изменений. Если все-таки хотя бы один из параметров будет значащим, то пункт меню «Show» станет доступен:



По вызову команды «Show» из главного меню открывается форма c результатом выборки и сортировки по заданным параметрам:



По вызову команды «Quit» из главного меню выход из программы.

Если переименовать файл с исходными данными, то при запуске приложения появится диалоговое окно предусмотренное на случай возникновения исключений:



**Учебно-методическое и информационное обеспечение дисциплины**

Основная литература:

1. Павловская, Т.А. С#. Программирование на языке высокого уровня. Учебник для вузов. / Т.А. Павловская — СПб.: Питер, 2014. —
432 с.: ил.
2. Орлов, С.А. Теория и практика языков программирования: Учебник для вузов. Стандарт 3-го поколения. / С.А. Орлов – СПб.: Питер, 2014. – 688 с.: ил.

Дополнительная литература:

1. Иванова, Г.С. Объектно- ориентированное программирование: учебник для вузов / Г.С.Иванова, Т.Н.Ничушкина, Е.К.Пугачев. – М.: Изд-во МГТУ им.Н.Э.Баумана, 2004.-320с.: ил.- (Серия Информатика в техническом университете).
2. Павловская, Т.А. С/С ++. Программирование на языке высокого уровня: учебник для вузов /Т.А.Павловская. – СПб.: Питер, 2009.- 461с.: ил.

 Программное обеспечение

1. Embarcadero RAD Studio 2006
2. Microsoft Visual Studio Express

Базы данных, информационно-справочные и поисковые системы:

1. Электронный ресурс Национального открытого университета ИНТУИТ www.intuit.ru