

Лабораторная работа №1. Разработка программ на языке C++

Часть I. Подготовка к работе

Задание 1: Создайте консольное приложение для вывода текстового сообщения

1. Запустите интегрированную среду разработки **Microsoft Visual Studio 2010**.
2. Создайте новый проект:
 - 1) выберите пункт главного меню: **Файл – Создать – Проект**;
 - 2) в появившемся диалоговом окне в разделе **Установленные шаблоны** выберите группу **Visual C++**;
 - 3) выберите шаблон **Консольное приложение Win32 (C++)**.
 - 4) введите имя проекта **Example_1**;
 - 5) укажите место расположения файлов проекта в вашей личной папке (**кнопка Обзор**);
 - 6) введите имя решения **Лабораторная работа 1**;
 - 7) нажмите кнопку **ОК**.
3. В открывшемся **Мастере приложений win32** ознакомьтесь с информацией и нажмите кнопку **Далее**;
 - 1) установите флаги:
Тип приложения: консольное приложение.
Дополнительные параметры: пустой проект.
 - 2) нажмите кнопку **Готово**.
4. Создайте файл исходного кода:
 - 1) выберите пункт главного меню: **Проект – Добавить новый элемент...**;
 - 2) выберите шаблон **Файл C++ (.cpp)**;
 - 3) введите имя файла **source** и нажмите кнопку **Добавить**.
5. В редакторе исходного кода введите текст программы:

```
#include <iostream> //подключение библиотеки ввода-вывода
using namespace std; //подключение пространства имен, содержащего функции библиотеки

void main() //главная функция - точка входа в программу
{
    cout << "Hello, world!" << endl; //вывод сообщения на экран
    system("pause"); //отправка сообщения "Пауза" системе
}
```
6. Запустите проект на выполнение: **Отладка – Начать отладку**.
7. В папке сохранения проекта найдите каталог **Debug** и запустите приложение **Example_1.exe**.
8. Скопируйте этот файл в любой другой каталог на локальном диске и запустите. Сделайте вывод.
9. Закройте проект командой **Файл – Заккрыть решение** и откройте заново командой **Файл – Открыть – Решение или проект...** . В каталоге сохранения выберите файл решения **Лабораторная работа 1.sln**.

Часть II. Ввод-вывод данных

Задание 2: Создайте приложение, обеспечивающее ввод и вывод данных с использованием символов кириллицы.

1. Добавьте в решение новый проект:

- 1) выберите пункт главного меню: **Файл – Добавить – Создать проект...**;
- 2) выберите тип проекта **Консольное приложение Win32**;
- 3) введите имя проекта **Example_2**;
- 4) расположение проекта не редактируйте;
- 5) нажмите кнопку **ОК**;
- 6) выполните настройку проекта согласно п.3 из Части I.

2. Сделайте новый проект запускаемым по умолчанию:

- 1) выделите проект **Example_2** в **Обозревателе решений**;
- 2) выберите пункт главного меню: **Проект – Назначить запускаемым**.

3. Создайте файл исходного кода **source.cpp** согласно п. 4. Части I.

4. В файле исходного кода подключите необходимые библиотеки:

```
#include <iostream>    //библиотека ввода-вывода
#include <string>       //библиотека работы со строками
#include <windows.h>    //библиотека системных функций Windows
```

5. Там же подключите пространство имен стандартной библиотеки:

```
using namespace std;
```

6. Создайте главную функцию программы:

```
void main()
{
    //тело главной функции
}
```

7. В теле главной функции настройте использование русских символов:

```
SetConsoleOutputCP(1251); //установка кириллицы для вывода в консоль
SetConsoleCP(1251);       //установка кириллицы для ввода с консоли
```

8. Там же объявите строковую переменную:

```
string name; //имя пользователя
```

9. Добавьте вывод на экран сообщения для пользователя:

```
cout << "Введите ваше имя: " << endl;
```

10. Считайте из потока ввода значение:

```
cin >> name; //считывание значения из консоли в строковую переменную
```

11. Выведите сообщение с приветствием:

```
cout << "Здравствуй, " << name << "!" << endl; //вывод сообщения на экран
```

12. Отправьте сообщение «Пауза» системе, чтобы предотвратить преждевременное закрытие консоли:

```
system("pause"); //отправка сообщения "Пауза" системе
```

13. Запустите проект на выполнение и проверьте правильность его работы. Вводите строки, содержащие английские буквы, русские буквы, цифры и другие символы.

Задание 3: Разработайте консольное приложение для нахождения среднего арифметического целых чисел вводимых пользователем.

1. Создайте новый проект **Example_3** в решении **Лабораторная работа 1** и назначьте его запускаемым по умолчанию.
2. В новом файле исходного кода подключите библиотеки и пространства имен для работы с функциями ввода-вывода и настройки кодировки, выполните настройку использования русских символов.
3. Объявите следующие переменные:

```
int count, n, sum, averange; //количество чисел, число, сумма, среднее
```

4. В переменную **count** считайте количество чисел, которые будет вводить пользователь:

```
cout << "Введите количество чисел: " << endl;  
cin >> count;
```

5. С помощью цикла считайте числа и вычислите их сумму:

```
sum=0; //обнуляем сумму  
//считываем необходимое количество чисел  
for (int i=1; i<=count; i++)  
{  
    cout << "Введите " << i << "-е число:" << endl;  
    cin >> n;  
    sum=sum+n; //прибавляем введенное число к текущему значению суммы  
}
```

6. Выведите на экран получившееся значение:

```
cout << "Сумма чисел равна: " << sum << endl;
```

7. Вычислите и выведите среднее арифметическое:

```
averange = sum / count; //деление целого числа на целое  
cout << "Среднее арифметическое чисел равно: " << averange << endl;
```

8. Выполните отладку и проверьте правильность работы программы.
9. Для получения дробного результата деления, объявите переменную вещественного типа:

```
float averangeFloat;
```

10. Снова вычислите среднее арифметическое, преобразовав сумму к вещественному типу:

```
averangeFloat = (float)sum / count; //деление дробного числа на целое  
cout.precision(2); //устанавливаем количество знаков после запятой для дробных чисел  
cout << "Среднее арифметическое чисел равно: " << averangeFloat << endl;
```

11. Выполните отладку и подберите тестовые данные, обеспечивающие получение дробного результата.
12. Проверьте правильность работы программы.

13. **Самостоятельно** дополните программу подсчетом количества четных чисел, введенных пользователем.

Задание 4: Создайте консольное приложение, демонстрирующее обработку ввода данных с помощью механизма исключений.

1. Создайте новый проект **Example_4** в решении **Лабораторная работа 1**.

2. Создайте новый файл исходного кода.

3. В теле главной функции объявите следующие переменные:

```
int a, b; //целочисленные переменные
```

4. Опишите ввод значений переменных и вывод их суммы:

```
try //блок, в котором предполагается возникновение исключительной ситуации
{
    cout << "Введите значение переменной a: " << endl;
    cin >> a;          //считываем значения в переменную

    if (!cin.good()) //если в потоке возникла ошибка - введено не число
    {
        throw 1;     //генерируем исключение с кодом 1
    }

    cout << "Введите значение переменной b: " << endl;
    cin >> b;

    if (!cin.good()) //если в потоке возникла ошибка - введено не число
    {
        throw 2;     //генерируем исключение с кодом 2
    }
    cout << "Сумма чисел a и b равна: " << a+b << endl;
    cout << "Программа выполнена успешно!" << endl;
}
```

5. Далее добавьте блок обработки исключений:

```
catch (int errorCode)//блок обработки исключений
{
    if (errorCode==1) //если получено исключение с кодом 1
    {
        cout << "Неверное значение переменной a!" << endl;
    }
    else
    if (errorCode==2) //если получено исключение с кодом 2
    {
        cout << "Неверное значение переменной b!" << endl;
    }
    cout << "Программа выполнена с ошибкой!" << endl;
}
```

6. Запустите проект на выполнение и проверьте его работу при вводе числовых и строковых данных.

Задание 5: Разработайте консольное приложение «Калькулятор».

1. В текущее решение добавьте проект **Example_5** и назначьте его запускаемым.

2. Подключите библиотеку математических функций **math.h**.

3. В теле главной функции объявите следующие переменные:

```
float a, b; //аргументы операций калькулятора
int choise; //выбранная пользователем операция
```

4. Далее организуйте вывод текстового меню и выполнение математических операций в соответствии с выбранной пользователем операцией:

```
//начало цикла работы программы
do
{
    system("cls"); //очистка экрана

    //вывод текстового меню
    cout<<"Выберите действие:" << endl;
    cout<<"1 - Нахождение суммы двух чисел" << endl;
    cout<<"2 - Нахождение произведения двух чисел" << endl;
    cout<<"3 - Нахождение значения функции синус" << endl;
    cout<<"4 - Выход" << endl;
    cout<< "Ваш выбор: " << endl;

    //считывание значения, введенного пользователем
    cin >> choise;

    //проверка значения
    switch (choise)
    {
        case 1:
            //нахождение суммы
            cout<<"Введите число a:" << endl;
            cin>>a;
            cout<<"Введите число b:" << endl;
            cin>>b;
            cout<< "a+b= " << a+b << endl;
            break;
        case 2:
            //нахождение произведения
            cout<< "Введите число a:" << endl;
            cin>>a;
            cout<< "Введите число b:" << endl;
            cin >>b;
            cout<< "a*b=" << a*b << endl;
            break;
        case 3:
            //нахождение значения синуса
            cout<< "Введите число a:" << endl;
            cin>>a;
            cout<< "sin(a)=" << sin(a) << endl;
            break;
        case 4:
            //выход из программы
            cout<< "Программа завершена" << endl;
            break;
        default:
            //если введено неверное значение
            cout << "Ошибка ввода!" << endl;
            break;
    }
    system("pause"); //приостановка выполнения программы
}
//цикл выполняется до тех пор, пока пользователь не введет правильное значение
while (choise!=4);
```

5. Самостоятельно дополните программу возможностью вычисления косинуса, квадратного корня, возведения в степень и нахождения остатка от деления.

Часть III. Задания для самостоятельной работы

Задание 6: Разработайте консольное приложение, позволяющее производить вычисление значения функции $y = -2,4x^2 + 5x - 3$ в диапазоне от -2 до 2 , с шагом $0,5$.

Задание 7: Разработайте программу, которая проверяет существование треугольника со сторонами a , b и c . Значения длин сторон задает пользователь. Обеспечьте проверку корректности ввода.

Задание 8: Разработайте консольное приложение с текстовым меню, позволяющее производить следующие действия по конвертации валют: 1 – рубли в доллары, 2 – рубли в евро, 3 – евро в рубли, 4 – доллары в рубли. Курс валют задайте в коде программы. При выводе дробного результата отображайте две цифры после запятой.