Лабораторное занятие 21 (6 часов – 3 пары)

**Тема:** Создание и отладка программ работы с функциями

**Цель:** научиться создавать программы работы с функциями.

**Задание 1.** Написать программу вычисления суммы трех чисел, используя подпрограмму вычисления суммы двух чисел.

Текст программы:

#include <iostream>

#include <windows.h>

#include <iomanip>

#include <string>

using namespace std;

double summ (double x1, double x2)

{

double s;

s = x1 + x2;

return s;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

double a, b, c;

cout<<"Введите 3 числа - ";

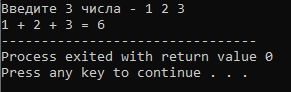
cin>>a>>b>>c;

cout<<a<<" + "<<b<<" + "<<c<<" = "<<summ (summ (a, b), c);

return 0;

}

Скрины результатов работы:



**Задание 2.** Даны два трехзначных числа. Вывести на экран наибольшую из средних цифр этих чисел, используя подпрограмму поиска средней цифры.

Текст программы:

#include <iostream>

#include <iomanip>

#include <windows.h>

using namespace std;

int sr (int x)

{

return (x / 10) % 10;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int a, b;

int sr\_1, sr\_2;

cout<<"Введите 2 трёхзначных числа - ";

cin>>a>>b;

if ((a > 999) || (a < 100))

{

cout<<"Первое число не являеться трйхзначным.";

return 11;

}

if ((b > 999) || (b < 100))

{

cout<<"Второе число не являеться трйхзначным.";

return 12;

}

sr\_1 = sr (a);

sr\_2 = sr (b);

if (sr\_1 > sr\_2) cout<<"Наибольшая средняя цифра: "<<sr\_1;

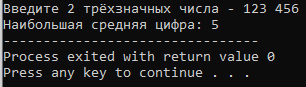
if (sr\_1 < sr\_2) cout<<"Наибольшая средняя цифра: "<<sr\_2;

if (sr\_1 == sr\_2) cout<<"Средние цифры обоих чисел равны";

return 0;

}

Скрины результатов работы:



**Задание 3.** Составьте программу с использованием подпрограммы.

Два круга заданы радиусами. Найти их площади и сравнить.

Текст программы:

#include <iostream>

#include <windows.h>

#include <math.h>

using namespace std;

double S (double x)

{

return M\_PI \* x \* x;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

double a, b;

cout<<"Введите радиус первого круга - ";

cin>>a;

if (a < 0)

{

cout<<"Радиус не может быть меньше 0.";

return 11;

}

cout<<"Введите радиус второго круга - ";

cin>>b;

if (b < 0)

{

cout<<"Радиус не может быть меньше 0.";

return 12;

}

cout<<"Площадь первого круга = "<<S (a)<<endl;

cout<<"Площадь второго круга = "<<S (b)<<endl;

if (S (a) > S (b)) cout<<"Площадь первой круга больше.";

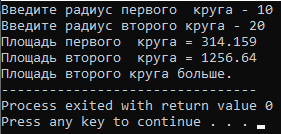
if (S (a) < S (b)) cout<<"Площадь второго круга больше.";

if (S (a) == S (b)) cout<<"Площади обоих кругов равны.";

return 0;

}

Скрины результатов работы:



**Задание 4.** Даны два натуральных числа. Определить, используя функции, в каком из них:

больше количество цифр 5.

Текст программы:

#include <iostream>

#include <windows.h>

#include <math.h>

using namespace std;

int quantity (int x)

{

int i = 0;

while (x > 0)

{

if ((x % 10) == 5) i++;

x /= 10;

}

return i;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int a, b;

int sr\_1, sr\_2;

cout<<"Введите первое натуральное число - ";

cin>>a;

if (a < 1)

{

cout<<"Первое число не являеться натуральным.";

return 11;

}

cout<<"Введите второе натуральное число - ";

cin>>b;

if (b < 1)

{

cout<<"Второе число не являеться натуральным.";

return 11;

}

sr\_1 = quantity (a);

sr\_2 = quantity (b);

cout<<"Количество цифр '5' в первом числе - "<<sr\_1<<endl;

cout<<"Количество цифр '5' во втором числе - "<<sr\_2<<endl;

if (sr\_1 > sr\_2) cout<<"Первое число содержит больше цифр '5'.";

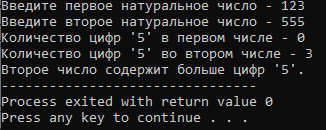
if (sr\_1 < sr\_2) cout<<"Второе число содержит больше цифр '5'.";

if (sr\_1 == sr\_2) cout<<"Оба числа содержат равное количество цифр '5'.";

return 0;

}

Скрины результатов работы:



**Задание 5.** Даны 4 отрезка a, b, c, d. Найти площади треугольников (возможных), построенных из этих отрезков.

Текст программы:

#include <iostream>

#include <windows.h>

#include <math.h>

using namespace std;

boolean check (double x\_1, double x\_2, double x\_3)

{

if ((x\_1 + x\_2 > x\_3) && (x\_1 + x\_2 > x\_3) && (x\_2 + x\_3 > x\_1) && (x\_1 > 0) && (x\_2 > 0) && (x\_3 > 0)) return true;

else return false;

};

void S (double x1, double x2, double x3)

{

double s = 0, p = 0;

p = (x1 + x2 + x3) / 2;

s = sqrt (p \* (p - x1) \* (p - x2) \* (p - x3));

cout<<"Площадь треугольника со сторонами "<<x1<<", "<<x2<<", "<<x3<<" = "<<s<<endl;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int i = 0;

double a, b, c, d;

double s1 = 0, s2 = 0, s3 = 0, s4 = 0;

cout<<"Введите длины 4 отрезков - ";

cin>>a>>b>>c>>d;

if (check (a, b, c))

{

S (a, b, c);

i++;

}

if (check (a, b, d))

{

S (a, b, d);

i++;

}

if (check (a, c, d))

{

S (a, c, d);

i++;

}

if (check (b, c, d))

{

S (b, c, d);

i++;

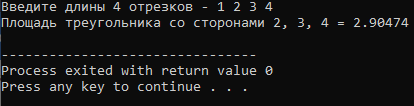
}

if (i == 0) cout<<"Невозможно составить ниодного треугольника.";

return 0;

}

Скрины результатов работы:



**Задание 6.** Даны два предложения. Найти, используя подпрограмму:

общее количество сочетаний «ас» в них.

Текст программы:

#include <iostream>

#include <windows.h>

#include <string.h>

using namespace std;

int quantity (string s)

{

int q = 0;

for (int i = 0 ; i < s.length() ; i ++)

{

if ((s[i] == 'а') && (s[i + 1] == 'с')) q++;

}

return q;

}

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

string s1, s2;

cout<<"Введите первое предложение - ";

getline(cin, s1);

cout<<"Введите второе предложение - ";

getline(cin, s2);

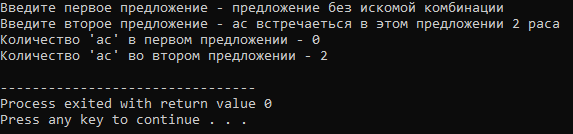
cout<<"Количество 'ас' в первом предложении - "<<quantity (s1)<<endl;

cout<<"Количество 'ас' во втором предложении - "<<quantity (s2)<<endl;

return 0;

}

Скрины результатов работы:



**Задание 7**. Даны два массива a и b целых чисел из диапазона [-15;15].

Определить: наименьший из максимальных элементов массивов.

Текст программы:

#include <iostream>

#include <windows.h>

#include <cstdlib>

#include <time.h>

using namespace std;

int max\_f (int mass[], int range)

{

int i;

if (range == 0) return 0;

int max = mass[0];

for (i = 0 ; i < range ; i ++)

{

if (mass[i] > max) max = mass[i];

}

return max;

};

void input\_mass (int mass[], int range\_in)

{

int i\_in = 0;

for (i\_in ; i\_in < range\_in ; i\_in ++)

mass[i\_in] = rand() % 31 - 15;

};

void output\_mass (int mass[], int range\_out)

{

int i\_out = 0;

for (i\_out ; i\_out < range\_out ; i\_out ++)

cout<<mass[i\_out]<<" ";

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int id, last\_id\_1, last\_id\_2, max\_1, max\_2;

cout<<"Введите количество элементов в первом массиве - ";

cin>>last\_id\_1;

if (last\_id\_1 < 0)

{

cout<<"Количество элементов в первом массиве не может быть меньше 0";

return 11;

}

last\_id\_1 ++;

int a[last\_id\_1];

input\_mass(a, last\_id\_1);

cout<<"Введите количество элементов во втором массиве - ";

cin>>last\_id\_2;

if (last\_id\_2 < 0)

{

cout<<"Количество элементов во втором массиве не может быть меньше 0";

return 12;

}

last\_id\_2 ++;

int b[last\_id\_2];

input\_mass(b, last\_id\_2);

cout<<"Первый массив из "<<last\_id\_1<<" целых чисел в диапазоне от 15 до 15:"<<endl;

output\_mass(a, last\_id\_1);

cout<<endl<<"Второй массив из "<<last\_id\_2<<" целых чисел в диапазоне от 15 до 15:"<<endl;

output\_mass(b, last\_id\_2);

max\_1 = max\_f(a, last\_id\_1);

max\_2 = max\_f(b, last\_id\_2);

cout<<endl<<"Максимальный элемент первого массива - "<<max\_1<<endl;

cout<<"Максимальный элемент второго массива - "<<max\_2<<endl;

if (max\_1 < max\_2) cout<<"Максимальный элемент первоого массива меньше, чем максимальный элемент второго массива.";

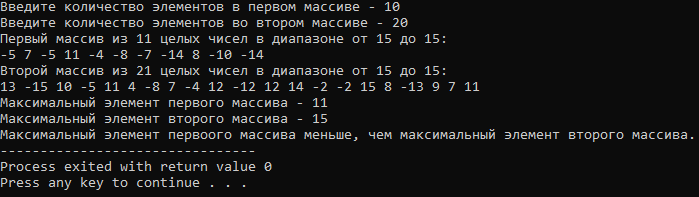
if (max\_1 > max\_2) cout<<"Максимальный элемент второго массива меньше, чем максимальный элемент первого массива.";

if (max\_1 == max\_2) cout<<"Максимальные элементы первоого и второго массивов равны.";

return 0;

}

Скрины результатов работы:



**Задание 8.** Даны два массива целых чисел одинаковой размерности.

Составить из элементов данных массивов новый (третий) массив по правилу: каждый элемент нового массива есть наибольший из соответствующих элементов данных массивов.

Текст программы:

#include <iostream>

#include <windows.h>

#include <cstdlib>

#include <time.h>

using namespace std;

void input\_mass (int mass[], int range\_in, int range\_a, int range\_b)

{

int i\_in = 0;

for (i\_in ; i\_in < range\_in ; i\_in ++)

mass[i\_in] = rand() % (range\_b - range\_a + 1) + range\_a;

};

void output\_mass (int mass[], int range\_out)

{

int i\_out = 0;

for (i\_out ; i\_out < range\_out ; i\_out ++)

cout<<mass[i\_out]<<" ";

};

void input\_max (int mass\_a[], int mass\_b[], int mass\_c[], int range\_max)

{

int i\_max = 0;

for (i\_max ; i\_max < range\_max ; i\_max ++)

{

if (mass\_a[i\_max] > mass\_b[i\_max]) mass\_c[i\_max] = mass\_a[i\_max];

if (mass\_a[i\_max] < mass\_b[i\_max]) mass\_c[i\_max] = mass\_b[i\_max];

if (mass\_a[i\_max] == mass\_b[i\_max]) mass\_c[i\_max] = mass\_a[i\_max];

}

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int id, last\_id, range\_min, range\_max;

cout<<"Введите количество элементов в массивах - ";

cin>>last\_id;

if (last\_id < 0)

{

cout<<"Количество элементов в массивах не может быть меньше 0";

return 11;

}

cout<<"Введите диапазон значений массива - ";

cin>>range\_min>>range\_max;

if (range\_min > range\_max)

{

cout<<"Неверно задан диапазон значений массива.";

return 12;

}

last\_id ++;

int a[last\_id], b[last\_id], c[last\_id];

last\_id --;

input\_mass(a, last\_id, range\_min, range\_max);

input\_mass(b, last\_id, range\_min, range\_max);

cout<<"Первый массив из "<<last\_id<<" целых чисел в диапазоне от "<<range\_min<<" до "<<range\_max<<":"<<endl;

output\_mass(a, last\_id);

cout<<endl<<"Второй массив из "<<last\_id<<" целых чисел в диапазоне от "<<range\_min<<" до "<<range\_max<<":"<<endl;

output\_mass(b, last\_id);

cout<<endl<<"Третий массив, составленный по правилу: каждый элемент нового массива есть наибольший из соответствующих элементов данных массивов:"<<endl;

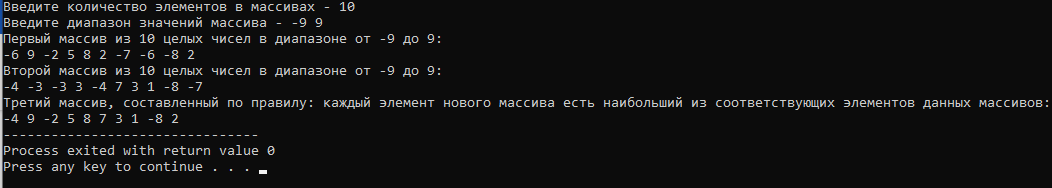
input\_max (a, b, c, last\_id);

output\_mass(c, last\_id);

return 0;

}

Скрины результатов работы:



**Задание 9.**

Вычислить значение *z=f(a-b,2b)+f(a2+b, b3-1),* где



Текст программы:

#include <iostream>

#include <windows.h>

#include <math.h>

using namespace std;

double F (double u, double t)

{

if ((u > 0) && (t > 0)) return 3 \* u + t;

if ((u > 0) && (t <= 0)) return 5 \* u - t;

if (u <= 0) return 2 \* u - sin (t);

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

double a, b, z, f\_1, f\_2;

cout<<"Введите значения A и B - ";

cin>>a>>b;

f\_1 = F (a - b, 2 \* b);

f\_2 = F (a \* a + b, b \* b \* b - 1);

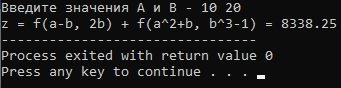
z = f\_1 + f\_2;

cout<<"z = f(a-b, 2b) + f(a^2+b, b^3-1) = "<<z;

return 0;

}

Скрины результатов работы:



**Задание 10.**

Вывести все натуральные числа, не превосходящие заданное N (N>10) и делящиеся на свою вторую цифру (цифры числа нумеруются справа налево).

Текст программы:

#include <iostream>

#include <windows.h>

#include <math.h>

using namespace std;

int F (int X)

{

if ((X / 10) % 10 != 0)

{

if (X % ((X / 10) % 10) == 0) return X;

}

else return 0;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int N, i = 0;

cout<<"Введите значение N - ";

cin>>N;

if (N < 10)

{

cout<<"Значение N должно быть больше 10.";

return 11;

}

cout<<"натуральные числа, не превосходящие "<<N<<" делящиеся на свою 2 справа цифру: ";

for (int n = 10 ; n <= N ; n ++)

{

if (F(n) != 0) cout<<F(n)<<" ";

else i++;

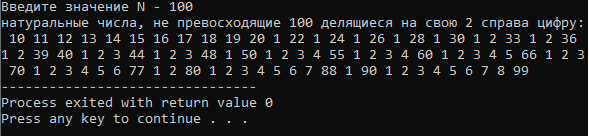
}

if (i == N) cout<<"Отсутствуют.";

return 0;

}

Скрины результатов работы:



**Задание 11.** Используя функцию (а – обычную, б – рекурсивную) вычисления факториала натурального числа n, найти.

1! + 3! + 5! + … + k! (k – нечетное)

Текст программы а:

#include <iostream>

#include <windows.h>

#include <math.h>

using namespace std;

void Fact (int b)

{

int K\_1 = 0, K\_2 = 1, K\_3 = 1, K\_4;

while (K\_2 <= b)

{

K\_4 = K\_2;

K\_3 = 1;

while (K\_4 > 1)

{

K\_3 \*= K\_4;

K\_4 --;

}

K\_1 += K\_3;

K\_2 += 2;

}

cout<<K\_1;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int k;

cout<<"Введите значение k - ";

cin>>k;

if (k < 1)

{

cout<<"Значение k должно быть больше 0.";

return 21;

}

if (k % 2 == 0)

{

cout<<"Значение k должно быть нечётным.";

return 22;

}

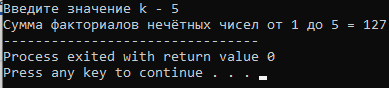
cout<<"Сумма факториалов нечётных чисел от 1 до "<<k<<" = ";

Fact (k);

return 0;

}

Скрины результатов работы а:



Текст программы б:

#include <iostream>

#include <windows.h>

#include <math.h>

using namespace std;

int fact (int a)

{

int N\_1 = a;

if (a > 1)

{

a \*= fact (N\_1 - 1);

}

return a;

};

int sum (int b)

{

int K\_1 = b;

if (b > 1)

{

b = fact (b);

b += sum (K\_1 - 2);

}

return b;

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int k;

cout<<"Введите значение k - ";

cin>>k;

if (k < 1)

{

cout<<"Значение k должно быть больше 0.";

return 21;

}

if (k % 2 == 0)

{

cout<<"Значение k должно быть нечётным.";

return 22;

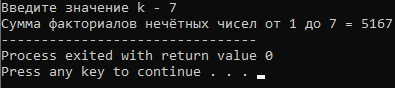
}

cout<<"Сумма факториалов нечётных чисел от 1 до "<<k<<" = "<<sum(k);

return 0;

}

Скрины результатов работы б:



**Задание 12.** Написать рекурсивную функцию вычисления чисел Фибоначчи: 1,1,2,3,5,8,13,21,… Используя ее, найти…

первые семь нечетных членов последовательности.

Текст программы:

#include <iostream>

#include <windows.h>

using namespace std;

int fib (int i)

{

if(i < 2) return 1;

else return fib(i - 1) + fib(i - 2);

};

int main(int argc, char\*\* argv)

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int quantity, n\_1 = 0, n\_2 = 0, i = 0;

cout<<"Первые 7 нечётных чисел последовательности Фибоначи:"<<endl;

while (i != 7)

{

if (fib(n\_1) % 2 == 1)

{

i ++;

}

n\_2 ++;

n\_1 ++;

}

for (n\_1 = 0 ; n\_1 < n\_2 ; n\_1 ++)

{

if (fib(n\_1) % 2 == 1)

{

cout<<fib(n\_1)<<" ";

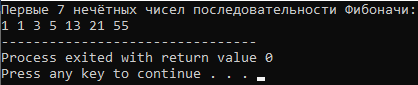
}

}

return 0;

}

Скрины результатов работы:



**Контрольные вопросы**

1. Выделение набора действий в подпрограмму и вызов её по мере необходимости позволяет логически выделить целостную подзадачу, имеющую типовое решение.

2. формальный параметр — аргумент, указываемый при объявлении или определении функции. фактический параметр — аргумент, передаваемый в функцию при её вызове;

3. Локальные параметры действуют только внутри одной функции, глобальные параметры могут работать в нескольких функциях

4. Сокращение и упрощение кода

5. Вызов функции внутри этой-же функци.