**Рекурсия**

**Цель лабораторной работы**: освоить рекурсию на языке программирования C.

**Задача 1**

**Постановка задачи**: Написать программу для вычисления факториала числа, введенного с клавиатуры.

**Программный код**:

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int getFact (int);

int main()

{

int number = 0;

printf("Enter the factorial: ");

scanf("%i", &number);

printf("Factorial of %i! = %i\n", number, getFact(number));

system("pause");

return 0;

}

int getFact(int number)

{

if ( number == 0 )

return 1;

return getFact(number - 1) \* number;

}

**Вывод консоли:**



**Задача 2**

**Постановка задачи**: Написать программу для вычисления чисел Фибоначчи

**Программный код**:

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int getFib(int);

int main()

{

int nmb = 0;

printf("Enter how long raw of fib: ");

scanf("%i", &nmb);

for (int i = 1; i <= nmb; i++)

printf("%i ", getFib(i));

printf("\n");

system("pause");

return 0;

}

int getFib(int number)

{

if ((number == 1) || (number == 2))

return 1;

return getFib(number - 1) + getFib(number - 2);

}

**Вывод консоли:**



**Задача 3**

**Постановка задачи**:

Определите функцию K(n), которая возвращает количество цифр в заданном натуральном числе n

**Программный код**:

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int getAmount(int);

int main()

{

int nmb = 0;

printf("Enter the number: ");

scanf("%i", &nmb);

printf("The number of digits in the number %i: %i \n", nmb, getAmount(nmb));

system("pause");

return 0;

}

int getAmount(int number)

{

if (number < 10)

return 1;

else

return getAmount(number / 10) + 1;

}

**Вывод консоли:**



**Задача 4**

**Постановка задачи**: Функция C(m, n), где 0 <= m <= n, для вычисления биномиального коэффициента.

**Программный код**:

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int getC(int, int);

int main()

{

int n, m;

printf("Enter the numbers n and m: ");

scanf("%i %i", &n, &m);

printf("Bin Coeff: %i\n", getC(n, m));

system("pause");

return 0;

}

int getC(int n, int m)

{

if ((m == 0) || (n == m))

return 1;

else

return getC(n -1 , m) + getC(n - 1, m - 1);

}

**Вывод консоли:**



**Задача 5**

**Постановка задачи**: Вычислить сумму элементов одномерного массива. Математическая модель: При решении задачи используйте следующее соображение: сумма равна нулю, если количество элементов равно нулю, и сумме всех предыдущих элементов плюс последний, если количество элементов не равно нулю.

**Программный код**:

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int sumOfElements(int\*, int);

int main()

{

int array[5] = {1, 2, 3, 4, 5};

printf("Array: ");

for (int i = 0; i < 5; i++)

printf("%i ", array[i]);

printf("\nSumm of elements = %i\n", sumOfElements(array, 5));

system("pause");

return 0;

}

int sumOfElements(int \*arr, int size)

{

if (size == 0)

return 0;

return ( sumsumOfEl(arr, size - 1) + arr[size - 1] ) ;

};

**Вывод консоли:**

