Національний технічний університет України

«Київський політехнічний інститут»

Кафедра автоматизованих систем обробки інформації та управління

Звіт про виконання лабораторної роботи №5 з дисципліни

«Алгоритмизація та програмування»

Перевірено: Ковалюк Т.В. Виконав ст. 1 курсу ФОІТ

Проскура С.Л. гр. ІС-52

Дорошенко А.В.

Київ 2015

## Лабораторна робота 5

# **Рекурсивні функції**

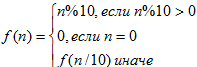
Варіант № 9

*Мета:*

1. Вивчити особливості рукурсивних процесів
2. Опанувати технологію рекурсивних обчислень
3. Навчитися розробляти алгоритми та програми із застосуванням рекурсивних функцій

### Завдання

Визначена рекурсивна функція f(n) через рекурентне співвідношення:



Визначити глибину рекурсії та обчислити функції S(n, q)за заданими значеннями p, q:

C:\Users\Anton\AppData\Local\Temp\Rar$EXa0.325\MethodLabWorkC++Semestr1Html\labs\lab_05\pictures\var9_2.gif

|  |  |
| --- | --- |
|  |  |

### Блок-схема алгоритму



Рис.1 (Головний алгоритм)



Рис.2 (Алгоритм для обчислення суми)



Рис.3 (Алгоритм для обчислення рекурсивної функції)

### Код програми

//Laboratory work

//written by student of the first curse

//of the group IC-52

//Anton Doroshenko

//2015.11.08

//==========================================================================

#include <iostream>

using namespace std;

int depth = 0; //Поточна глибина рекурсії

int dout = 0; //Найбільша глибина рекурсії

float sum; //Сума

int p, q; //Початкове і кінцеве значення параметра суми

float F(int n) //Оголошення функції

{

depth += 1; //Збільшити поточну глибину на 1

if (n % 10 > 0)

return (float)(n % 10);

else if (n == 0)

return 0;

else

return F(n / 10); //Рекурсивний виклик функції

}

float S(int p, int q) //Функція обчислення суми

{

for (int i = p; i <= q; i++)

{

sum += F(i);

if (dout < depth) //Якщо поточна глибина рекурсії більше найбільшої

dout = depth; //присвоїти значення поточної глибини найбільшій

depth = 0; //Обнулити поточну глибину рекурсії

}

return sum;

}

int main()

{

int cont;

do{

cout << "laboratory work number 5 made by Anton Doroshenko, IS-52" << endl;

cout << "Please, enter p, q" << endl;

cin >> p >> q;

S(p, q); //Виклик функції для обчислення суми

cout << "sum = " << sum << endl;

cout << "depth = " << dout << endl;

cout << "If you want to start again enter 1" << endl;

cin >> cont;

} while (cont == 1);

}

1. **Screen Shot результатів**

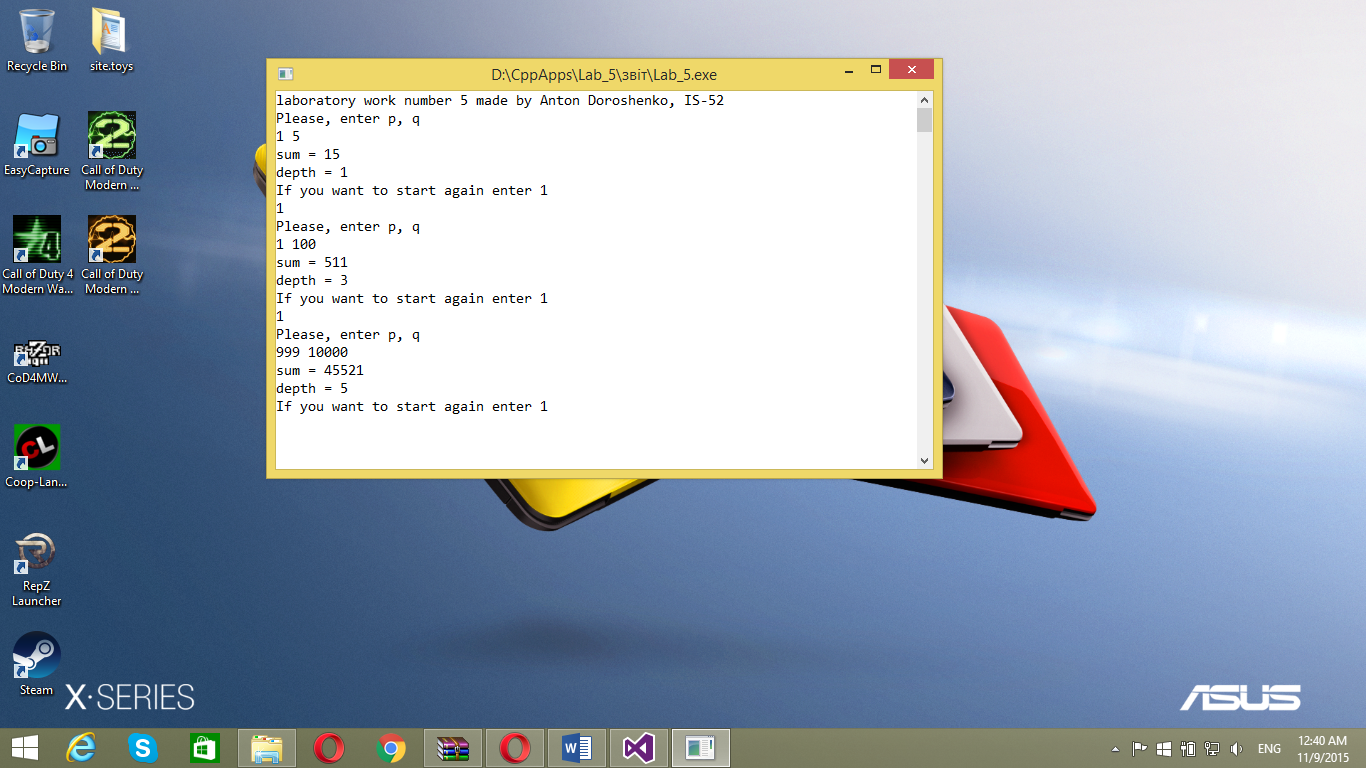


Рис.4 (Screen Shot результатів)

1. **Аналіз результатів**

Як ми бачимо зі ScreenShot’ів, ми ввели 3 варіанти даних.

1. Якщо початкове значення параметра суми дорівнює 1, а кінцеве 5, то сума буде дорівнювати 15, а глибина рекурсії дорівнюватиме 1.
2. Якщо початкове значення параметра суми дорівнює 1, а кінцеве 100, то сума буде дорівнювати 511, а глибина рекурсії дорівнюватиме 3.
3. Якщо початкове значення параметра суми дорівнює 999, а кінцеве 10 000, то сума буде дорівнювати 45 521, а глибина рекурсії дорівнюватиме 5.
4. **Висновок**

Програма працює правильно, про що свідчить аналіз результатів та ScreenShot’и. Програма коректно обчислює суму та глибину рекурсії.