

Міністерство освіти і науки України
Національний технічний університет
«Харківський політехнічний інститут»
Кафедра «Обчислювальна техніка та програмування»

ЗВІТ

Про виконання лабораторної роботи № 7
«Функції»

Керівник: викладач
Бульба С. С.

Виконавець: студент гр. КІТ-120в
Бабенко А. П.

Харків 2020

Лабораторна робота №7. Функції

1. Вимоги

1.1 Розробник

- Бабенко Антон Павлович;
- Студент групи КІТ-120в;
- 29 листопада 2020;

1.2 Загальне завдання

реалізувати функцію, що визначає, скільки серед заданої послідовності чисел таких пар, у котрих перше число менше наступного, використовуючи функцію з варіативною кількістю аргументів.

2 Виконання роботи

2.1 Створення файлу з вихідним кодом, написання коду і коментарів до нього. Зображено на рис.1.

```
1 | #include <stdarg.h>
2 |
3 | int func(int nums, ...)
4 | {
5 |     int count = 0; // Переменная для подсчёта количества пар
6 |     int arr[nums]; // Массив для записи чисел, переданных в функцию
7 |     va_list args; // Объявление списка аргументов
8 |     va_start(args, nums); // Начало работы со списком аргументов
9 |     for (int i = 0; i < nums; i++) {
10 |         arr[i] = va_arg(args, int); // Запись чисел переданных в функцию в массив
11 |     }
12 |     for (int i = 0; i < nums; i++) {
13 |         if (arr[i] < arr[i+1]) { // Сравнение первого и следующего числа из массива
14 |             count++;           // Увеличение количества пар на 1
15 |         }
16 |     }
17 |     va_end(args); // Конец работы со списком аргументов
18 |     return count; // Возвращение результата
19 | }
20 |
21 | int main()
22 | {
23 |     func(5, 1, 2, 3, 5, 6); // Вызов функции
24 |     return 0;
25 | }
```

Рисунок 1 — код програми

2.2 Компіляція проекту за допомогою команди “make clean prep compile”. Зображено на рис.2.

```
anton@anton-X55VD:~/dev/Programming-repo/lab07$ make clean prep compile
rm -rf dist
mkdir dist
gcc -std=gnu11 -g -Wall -Wextra -Wformat-security -Wfloat-equal -Wshadow -Wconversion -Wlogical-not-parentheses -Wnull-dereference -I./src src/task1.c -o ./dist/task1.bin
```

Рисунок 2 — компіляція проекту

2.3 Відкрито у відладнику netiver виконуючий файл main.bin. Ставимо точку зупину, проходимо по файлу і бачимо зміну значення count в момент проходження по циклу. Зображено на рис.3.

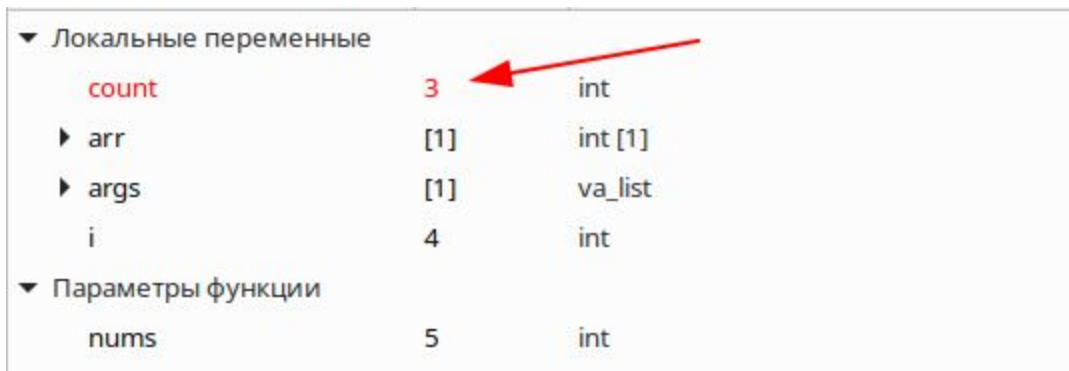


Рисунок 3 — файл у відладнику

2.4 Створення блок-схеми програми. Зображення блок-схеми на рис.4.

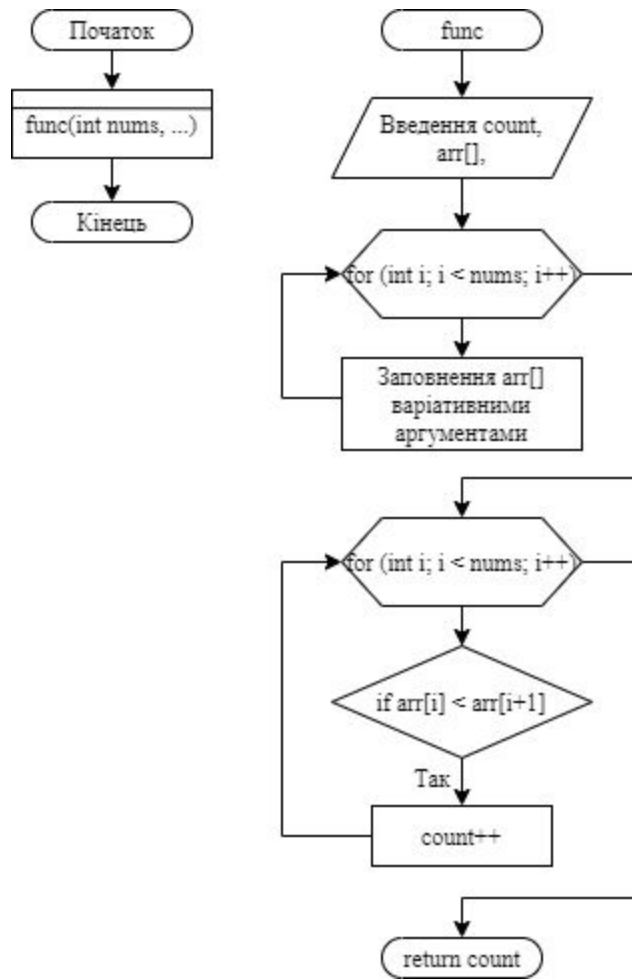


Рисунок 4 — блок-схема

2.10 Виконання опису функцій. Зображення опису зі сторінок документації doxygen на рис. 5, 6.

Опис функцій

◆ func()

```
int func ( int  nums,  
          ...  
          )
```

Розрахунок кількості пар $n-1 < n$.

Функція знаходить та повертає кількість пар чисел у котрих попереднє число менше за наступне. Передається варіативна кількість чисел

Аргументи

nums Кількість чисел

count Результат роботи функції

Повертає

повернення результату роботи функції (count)

◆ main()

```
int main ( )
```

Головна функція

Послідовність дій:

- виклик функції func

Повертає

успішний код виконання програми (0)

Граф всіх викликів цієї функції:



Рисунок 5, 6 — опис функцій

Висновки

При виконанні даної лабораторної роботи було набуто навичок розробки програм з використанням функцій і розроблено функцію з варіативною кількістю аргументів, що рахує кількість пар чисел у котрих перше число менше наступного.