

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

ЗВІТ

Про виконання лабораторної роботи № 6
«Масиви»

Керівник: викладач
Бульба С. С.

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

Харків 2020

Лабораторна робота №6. Масиви

1. Вимоги

1.1 Розробник

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

1.2 Загальне завдання

На оцінку “відмінно”. Необхідно виконати усі завдання з даної категорії (проте звіт та відповідні зміни до системи контролю версіями виконуються лише за одним обраним студентом варіантом).

1. Центрувати заданий рядок на площині з із заданим заповнювачем.
Наприклад,
 - заповнювач = “_”,
 - довжина строки = 15,
 - рядок = "Ivanov \0" (6 символів слово “Ivanov”, 8 - пробілів, останній символ = ‘\0’)
 - результат = “____Ivanov____” (4 символи заповнювача, слово “Ivanov”, 4 символи заповнювача, останній символ = ‘\0’)
2. Заповнити масив із заданої кількості елементів простими числами, що не повторюються. Розмір вихідного масиву задати наперед відомим значенням, що може будуть більшим ніж результуюча кількість отриманих елементів.
3. Перетворити число (максимальне значення якого - 9999) в рядок.
Наприклад,
 - 123 – “Сто двадцять три”,
 - 4321 – “Чотири тисячі триста двадцять один”.
4. У заданому тексті знайти кількість слів за умови, що між словами може бути будь-яка кількість пропусків.
5. Дано двовимірний масив з $N \times N$ цілих чисел. Виконати циклічне зрушення елементів рядків масиву в напрямку справа наліво (перший елемент рядка повинен переміститися в її кінець).
6. Дано двовимірний масив з $N \times N$ цілих чисел. Помножити матрицю саму на себе (відповідно до правил множення матриць).

1.3 Індивідуальне завдання

Перетворити число (максимальне значення - 9999) в рядок.

2 Виконання роботи

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

```
void numToStr(int num) {
    int temp = 0;
    int temp1 = 0;
    int temp2 = 0;
    int temp3 = 0;
    char res[100] = {0};
    const char arr1[10][10] = {"", "one", "two", "three", "four", "five",
        "six", "seven", "eight", "nine"};
    const char arr2[10][10] = {"ten", "eleven", "twelve", "thirteen", "fourteen",
        "fifteen", "sixteen", "seventeen", "eighteen", "nineteen"};
    const char arr3[10][15] = {"", "", "twenty", "thirty", "fourty", "fifty",
        "sixty", "seventy", "eighty", "ninety"};
    const char arr4[10][15] = {"", "one hundred", "two hundred", "three hundred",
        "four hundred", "five hundred", "six hundred", "seven hundred", "eight hundred", "nine hundred"};
    const char arr5[10][15] = {"", "one thousand", "two thousand", "three thousand",
        "four thousand", "five thousand", "six thousand", "seven thousand", "eight thousand", "nine thousand"};

    for (int i = 0; i <= num / 1000; i++)
    {
        temp = 0;
        for (int j = 0; j < 15; j++)
        {
            res[temp++] = arr5[i][j];
            temp1 = temp;
        }
    }

    for (int i = 0; i <= num / 100 % 10; i++)
    {
        temp = temp1;
        for (int j = 0; j < 15; j++)
        {
            res[temp++] = arr4[i][j];
            temp2 = temp;
        }
    }

    if (num % 100 >= 10 && num % 100 < 20)
    {
        for (int i = 0; i <= num % 10; i++)
        {
            temp = temp2;
            for (int j = 0; j < 10; j++)
            {
                res[temp++] = arr2[i][j];
            }
        }
    }
}
```

```

else if (num % 100 >= 20)
{
    for (int i = 0; i <= num / 10 % 10; i++)
    {
        temp = temp2;
        for (int j = 0; j < 10; j++)
        {
            res[temp++] = arr3[i][j];
            temp3 = temp;
        }
    }
    for (int i = 0; i <= num % 10; i++)
    {
        temp = temp3;
        for (int j = 0; j < 10; j++)
        {
            res[temp++] = arr1[i][j];
        }
    }
}
else
{
    for (int i = 0; i <= num % 10; i++)
    {
        temp = temp2;
        for (int j = 0; j < 10; j++)
        {
            res[temp++] = arr1[i][j];
        }
    }
}

if (num == 0)
{
    res[0] = 'z';
    res[1] = 'e';
    res[2] = 'r';
    res[3] = 'o';
}
}
}

```

Рисунок 1,2 — код програми

2.2 Компіляція проекту за допомогою команди “make clean prep compile”. Зображено на рис.3.

```

anton@anton-X55VD:~/dev/Programming-repo/lab06$ 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

```

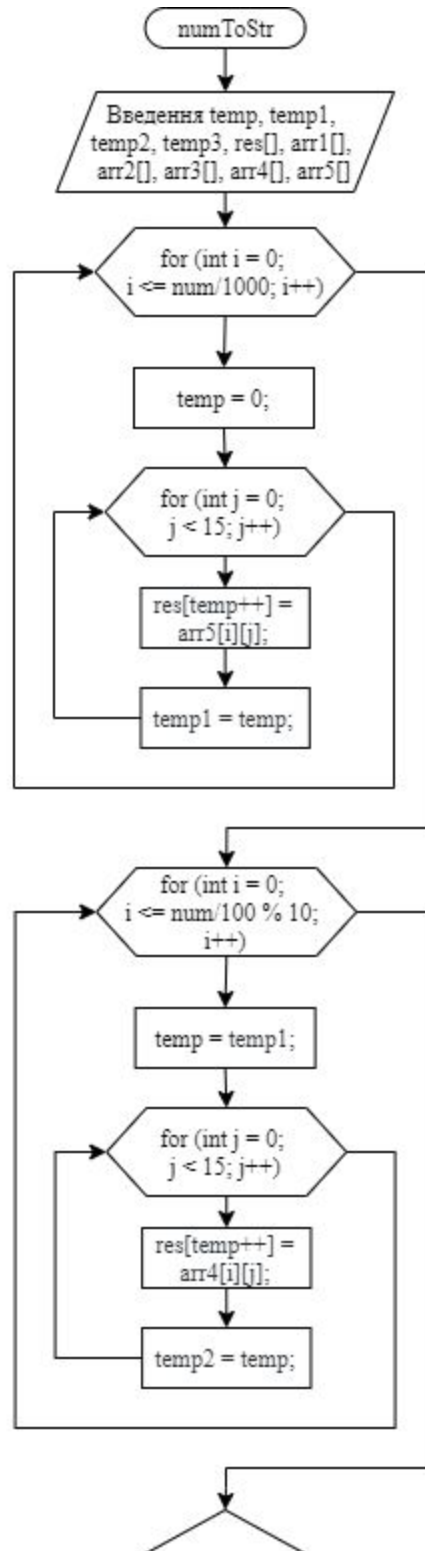
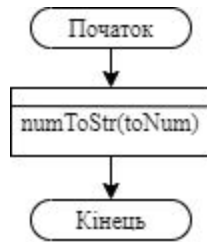
Рисунок 3 — компіляція проекту

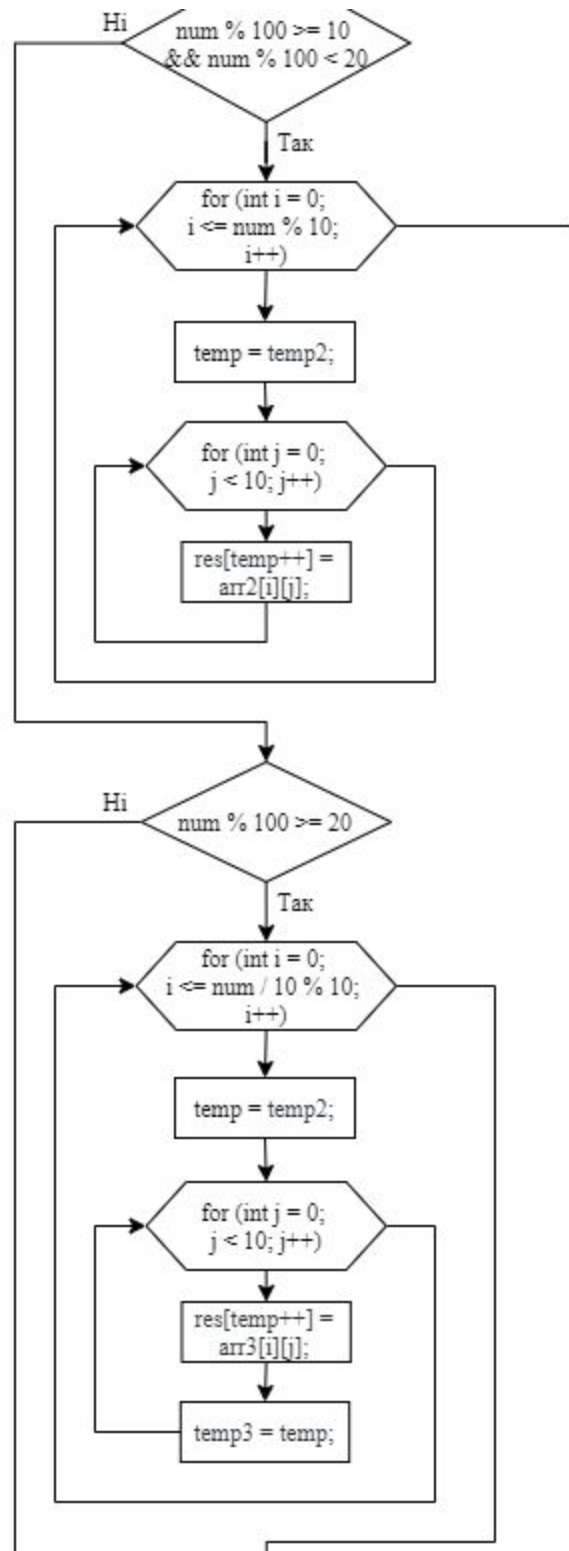
2.3 Відкрито у відладнику nemiver виконуючий файл main.bin. Ставимо точку зупину, проходимо по файлі і бачимо зміну значення в масиві в момент проходження по циклу. Зображено на рис.4.

▼ res	[100]	char [100]
0	111 'o'	char
1	110 'n'	char
2	101 'e'	char
3	32 ''	char
4	116 't'	char
5	104 'h'	char
6	111 'o'	char
7	117 'u'	char
8	115 's'	char
9	97 'a'	char
10	110 'n'	char
11	100 'd'	char

Рисунок 4 — файл у відлагоднику

2.5 Створення блок-схеми програми. Зображення блок-схеми на рис.5.





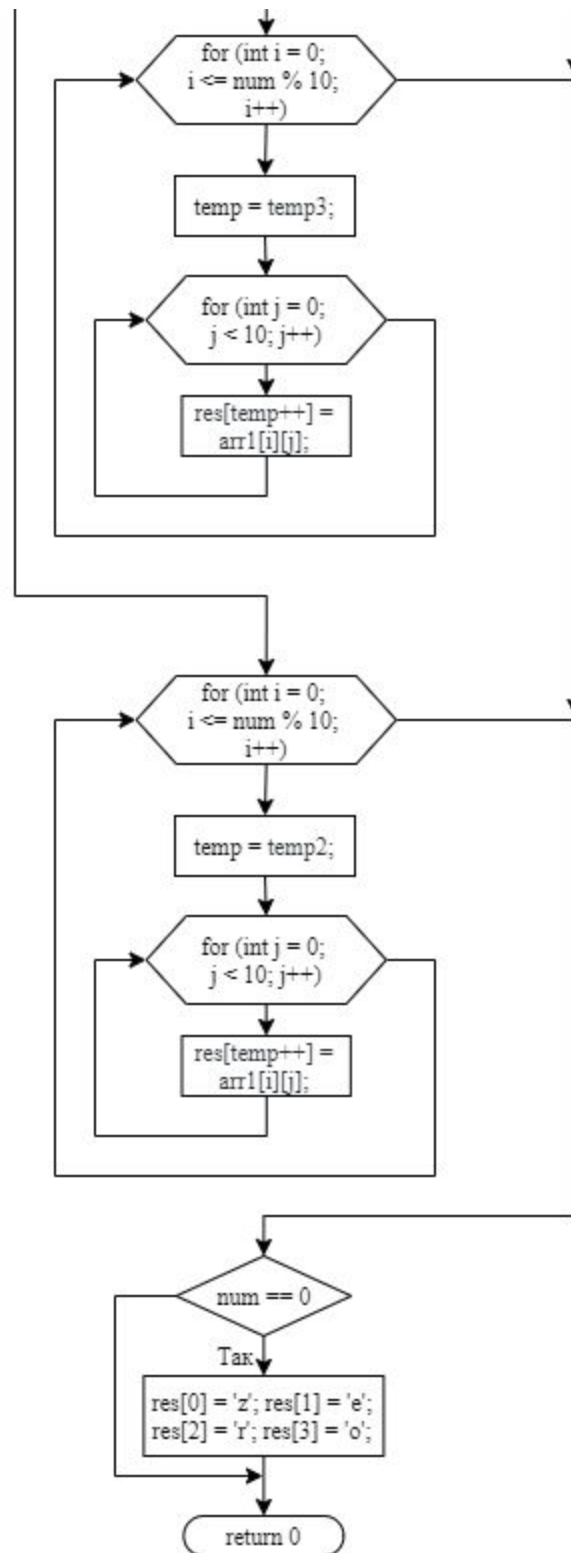


Рисунок 5 — блок-схема

2.10 Виконання опису функцій. Зображення опису зі сторінок документації doxygen на рис. 6, 7.

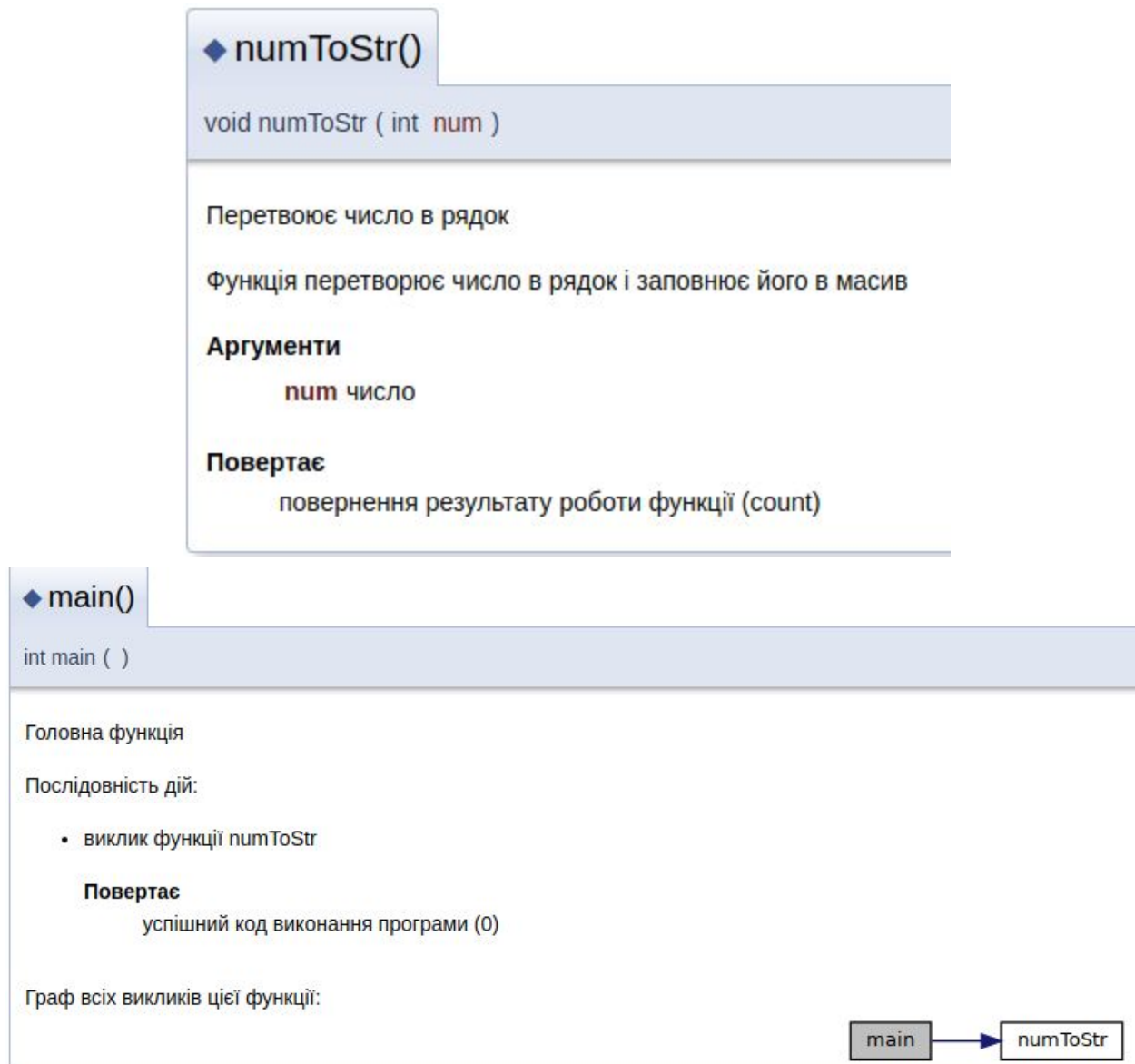


Рисунок 6, 7 — опис функцій

Висновки

При виконанні даної лабораторної роботи було набуто навичок розробки програм з циклічними конструкціями і розроблено 6 програм, а також створено програму, що перетворює число (максимальне значення - 9999) в рядок.

