Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет комп’ютерних систем та автоматики

Кафедра КСУ

Лабораторна робота №6

з дисципліни: «Компютерні технології та програмування»

Тема: «Рекурсія»

Виконав

студент групи 2АКІТ-20б

Н. Бондар

Перевірив

к.т.н., доц.кафедри КСУ

Севастьянов В. М.

м.Вінниця, ВНТУ 2021

**Лабораторна робота №6**

**Тема: «Рекурсія»**

**Варіант 1.**

**Завдання :**

1. Написати програму для виконання завдання згідно варіанту двома методами: рекурсивним та без використання рекурсії.

2. Відкомпілювати та відлагодити програму.

3. Розробити набір тестів і перевірити роботу програми на них.

4. Відповісти на контрольні запитання.

5. Зробити висновки.

6. Звіт по лабораторній роботі має складатися з титульної сторінки, лістингів програм, висновків по роботі.

**Варіант 1.**

Обчислити середнє значення елементів одновимірного масиву.

**Хід роботи :**

**1. Ретельно вивчаю поставлене передімною завдання.**

**2. Створюю програму на мові С згідно варіанту використавши середовище програмування Dev-C++ 4.0:**

**3. Відкомпільовую та відлагоджую програму.**

**4. Розробляю набір тестів і перевіряю роботу програми на них.**

**5. Відповідаю на контрольні запитання.**

**6. Роблю висновки.**

**1. Розписую функцію, яка обчислює середнє значення елементів одновимірного масиву рекурсивним методом :**

int recursion\_func(int \* array, int current\_position, int finish\_position)

{

if (current\_position == finish\_position)

{

return 0;

}

else

{

std::cout << \* array;

if (current\_position < finish\_position - 1)

{

std::cout << " + ";

}

return \* array + recursion\_func(array + 1, current\_position + 1, finish\_position);

}

}

**2. Розписую функцію, яка обчислює середнє значення елементів одновимірного масиву не рекурсивним методом :**

int unrecursion\_func(int \* array, int current\_position, int finish\_position)

{

int middle\_sum = 0;

for ( ; current\_position < finish\_position; current\_position++)

{

// std::cout << current\_position << ' ' << finish\_position << ' ' << array[current\_position] << '\n';

middle\_sum += array[current\_position];

}

return middle\_sum;

}

**3. Розписую головну функцію :**

int main()

{

int \* array;

int array\_size;

std::cout << "How many elements the array should store ? : ";

std::cin >> array\_size;

array = new int[array\_size];

for (int i = 0; i < array\_size; i++)

{

std::cout << "Enter " << i + 1 << " element : ";

std::cin >> array[i];

}

std::cout << "The example : (";

int temp = (recursion\_func(array, 0, array\_size) / array\_size);

std::cout << ") / " << array\_size << " = " << temp << "\nThe recursion func rusult : " << temp << '\n';

std::cout << "The unrecursion func rusult : " << (unrecursion\_func(array, 0, array\_size) / array\_size)<< '\n';

return 0;

}

**Лістинг коду :**

/\*

Варіант 1.

Обчислити середнє значення елементів одновимірного масиву.

\*/

#include<iostream>

int recursion\_func(int \* array, int current\_position, int finish\_position);

int unrecursion\_func(int \* array, int current\_position, int finish\_position);

int main()

{

int \* array;

int array\_size;

std::cout << "How many elements the array should store ? : ";

std::cin >> array\_size;

array = new int[array\_size];

for (int i = 0; i < array\_size; i++)

{

std::cout << "Enter " << i + 1 << " element : ";

std::cin >> array[i];

}

std::cout << "The example : (";

int temp = (recursion\_func(array, 0, array\_size) / array\_size);

std::cout << ") / " << array\_size << " = " << temp << "\nThe recursion func rusult : " << temp << '\n';

std::cout << "The unrecursion func rusult : " << (unrecursion\_func(array, 0, array\_size) / array\_size)<< '\n';

return 0;

}

int recursion\_func(int \* array, int current\_position, int finish\_position)

{

if (current\_position == finish\_position)

{

return 0;

}

else

{

std::cout << \* array;

if (current\_position < finish\_position - 1)

{

std::cout << " + ";

}

return \* array + recursion\_func(array + 1, current\_position + 1, finish\_position);

}

}

int unrecursion\_func(int \* array, int current\_position, int finish\_position)

{

int middle\_sum = 0;

for ( ; current\_position < finish\_position; current\_position++)

{

// std::cout << current\_position << ' ' << finish\_position << ' ' << array[current\_position] << '\n';

middle\_sum += array[current\_position];

}

return middle\_sum;

}

**Висновок :** виконуючи дану лабораторно роботу, я навчився використовувати практично рекурсію та її цикло-реалізований відповідник.