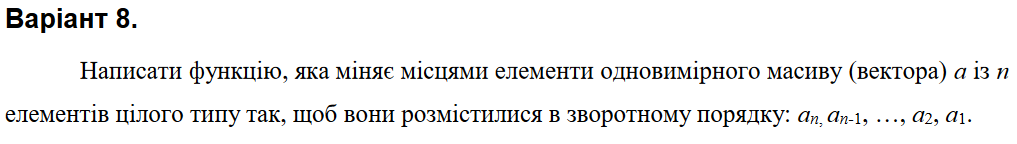
ЗВІТ  
про виконання лабораторної роботи № 6.2.2 (рекурсія)  
«Опрацювання одновимірних  
масивів ітераційними та рекурсивними способами»  
з дисципліни  
«Алгоритмізація та програмування»  
студента групи ІН-105 Б  
Горанов Анастас

**Мета роботи:** навчитися опрацьовувати одновимірні масиви.

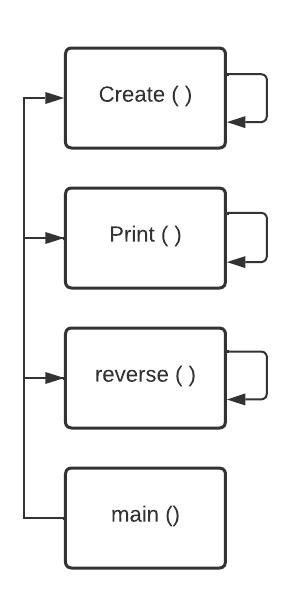
**Умова завдання:** Необхідно написати програму для того, щоб виконати такі дії:

- сформувати масив;  
- вивести його на екран у вигляді рядка, використовуючи форматне виведення;  
- виконати вказані у завдання дії;  
- вивести результат, причому, якщо масив був змінений – то вивести на екран модифікований масив у вигляді наступного рядка, використовуючи виведення з тими самими специфікаціями формату.

Спосіб 1. Всі вказані дії необхідно реалізувати за допомогою окремих функцій, що використовують ітераційні алгоритми.  
Спосіб 2. Всі вказані дії необхідно реалізувати за допомогою окремих рекурсивних функцій.  
Інформація у функції має передаватися лише за допомогою параметрів. Використання глобальних змінних – не допускається. Кожна функція має виконувати лише одну роль, і ця роль має бути відображена у назві функції.  
Рекурсивний та ітераційний способи – це 2 різні проекти, для яких потрібно 2 різних unit-тести і 2 різних звіти.  
«Функція, яка повертає / обчислює / шукає ...» – має не виводити ці значення, а повернути їх у місце виклику як результат функції або як відповідний вихідний параметр.



**Структурна схема програми:**



**Текст програми:**

// PR6/2/2.cpp

// Горанов Анастас

// Лабораторна робота №6.2.2

// Опрацювання одновимірних масивів ітераційними та рекурсивними способами

// Варіант 8

#include <iostream>

#include <cmath>

#include <iomanip>

using namespace std;

void Create(int\* a, const int size, const int Low, const int High, int i = 0)

{

if (i == size) {

return;

}

a[i] = Low + rand() % (High - Low + 1);

i++;

Create(a, size, Low, High, i);

}

void Print(int\* a, const int size, int i = 0)

{

if (i == size) {

cout << endl;

return;

}

cout << setw(4) << a[i];

i++;

Print(a, size, i);

}

void reverse(int\* arr, int i, int size) {

if (i == size)

return;

int elem = arr[i];

reverse(arr, i + 1, size);

arr[size - i - 1] = elem;

}

int main()

{

srand((unsigned)time(NULL));

const int n = 10;

int a[n];

int Low = -10;

int High = 10;

Create(a, n, Low, High);

Print(a, n);

reverse(a, 0, n);

cout << "After revers: " << endl;

Print(a, n);

return 0;

}

****

**Посилання на Git-репозиторій:** *https://github.com/StassNG/PR6.2.2*

**Результати unit-тесту:**

****

**Текст Unit-тесту:**

#include "pch.h"

#include "CppUnitTest.h"

#include "../PR6.2.2.cpp"

using namespace Microsoft::VisualStudio::CppUnitTestFramework;

namespace UnitTest1

{

TEST\_CLASS(UnitTest1)

{

public:

TEST\_METHOD(TestMethod1)

{

int arr[] = { 4, 5, -7,15, 4, -5, 6, 7 };

int res[] = { 7, 6, -5, 4, 15, -7, 5, 4 };

reverse(arr, 0, 8);

for (int i = 0; i < 8; i++) {

Assert::IsTrue(arr[i] == res[i]);

}

}

};

}

**Висновок:** навчився опрацьовувати одновимірні масиви.