**ЛАБОРАТОРНА РОБОТА №3**

*АЛГОРИТМИ ПЕРЕТВОРЕННЯ*

*ОДНОВИМІРНИХ МАСИВІВ (ВЕКТОРІВ)*

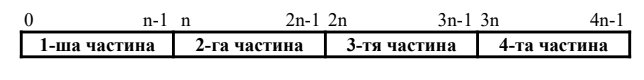
*Виконав: Землянський Едуард*

*Група: КВ-22*

**Постановка задачі**

Задано натуральне число n і одновимірний масив цілих чисел

A[4n], який складається з чотирьох частин по n елементів кожна.



1-ша частина 2-га частина 3-тя частина 4-та частина

Перетворити масив «на тому ж місці» (тобто результат пере-

творення повинен бути у тому ж масиві A), виконавши пере-

становки елементів масиву A у порядку, заданому за варіантом.

Розв’язати задачу двома способами (написати дві програми):

1) для виконання перетворення масиву A в програмі дозволяє-

ться використовувати один додатковий проміжний масив В розмі-

ром не більше 4n елементів. Кращим варіантом рішення буде ви-

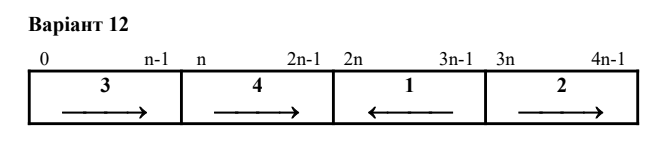
користання проміжного масиву В розміром у n елементів;

2) для виконання перетворення масиву A в програмі дозволяє-

ться використати тільки одну просту додаткову проміжну змінну

цілого типу.

**Варіант:**



**Текст програми 1** *(Додатковий масив)*

#include <time.h>

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main(){

    int \*A, n, B[1];

    srand(time(0));

    printf("Enter list size: ");

    scanf("%d", &n);

    A = malloc(sizeof(int)\*4\*n);

    printf("Unedited list: \n| ");

    for (int i = 0; i < (4\*n); i++){

        A[i] = rand()%10;

        printf("%d ", A[i]);

        if (i != 0 && (i+1) % n == 0){

            printf("| ");

        }

    }

    for (int i = 0; i < n; i++){

        // swap 1 and 3 parts

        B[0] = A[i];

        A[i] = A[2\*n+i];

        A[2\*n+i] = B[0];

        // swap 2 and 4 parts

        B[0] = A[n+i];

        A[n+i] = A[3\*n+i];

        A[3\*n+i] = B[0];

    }

    int counter = 3\*n-1;

    for (int i = 2\*n; i < counter; i++){

        B[0] = A[i];

        A[i] = A[counter];

        A[counter] = B[0];

        counter--;

    }

    printf("\nEdited list: \n| ");

    for (int i = 0; i < (4\*n); i++){

        printf("%d ", A[i]);

        if (i != 0 && (i+1) % n == 0){

            printf("| ");

        }

    }

    free(A);

    return 0;

    }

**Текст програми 2** *(Додаткова змінна)*

#include <time.h>

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main(){

    int \*A, n, temp;

    srand(time(0));

    printf("Enter list size: ");

    scanf("%d", &n);

    A = malloc(sizeof(int)\*4\*n);

    printf("Unedited list: \n| ");

    for (int i = 0; i < (4\*n); i++){

        A[i] = rand()%10;

        printf("%d ", A[i]);

        if (i != 0 && (i+1) % n == 0){

            printf("| ");

        }

    }

    for (int i = 0; i < n; i++){

        // swap 1 and 3 parts

        temp = A[i];

        A[i] = A[2\*n+i];

        A[2\*n+i] = temp;

        // swap 2 and 4 parts

        temp = A[n+i];

        A[n+i] = A[3\*n+i];

        A[3\*n+i] = temp;

    }

    int counter = 3\*n-1;

    for (int i = 2\*n; i < counter; i++){

        temp = A[i];

        A[i] = A[counter];

        A[counter] = temp;

        counter--;

    }

    printf("\nEdited list: \n| ");

    for (int i = 0; i < (4\*n); i++){

        printf("%d ", A[i]);

        if (i != 0 && (i+1) % n == 0){

            printf("| ");

        }

    }

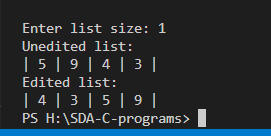
    free(A);

    return 0;

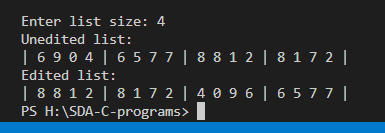
    }

**Тестування програм** *(еквівалентні для обох)*

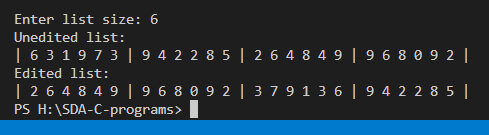
n = 1

****

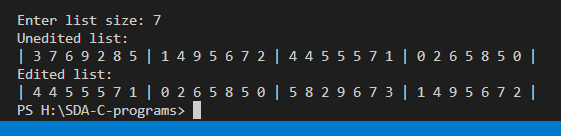
n = 4



n = 6



n = 7



n = 8

