**MINISTERUL EDUCAŢIEI, CULTURII ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**UNIVERSITATEA TEHNICĂ A MOLDOVEI**

**Facultatea Calculatoare, Informatică și Microelectronică**

**Departamentul Ingineria Software și Automatică**

**RAPORT**

**la lucrarea de laborator nr.1**

**Tema: „Algoritmi de sortare in lucrul cu clase si obiecte in C++”**

**Disciplina: „Programarea orientata pe obiecte ”**

**A elaborat *st. gr. SI-211, Vozian Vladimir***

**A verificat *lect.univ S. Scrob***

**Chișinău 2022**

**Sarcina**

De creat o clasa in care ca metode sunt algoritmii de sortare(minim 3)

**Codul programului:**

Main.cpp

#include<iostream>

#include <string>

#include "algoritms.cpp"

using namespace std;

int main()

{

    string metoda="";

    int a\_size;

    cout << "Da marimea: ";

    cin >> a\_size;

    int a[a\_size];

    for(int i=0; i<a\_size; i++)

    {

        a[i] = rand()%100;

    }

    for(int i=0; i<a\_size; i++)

    {

        cout << a[i] << ' ';

    }

    cout << "\n";

    sortare obj;

    cout << "Pentru bubble sort: Bubble"<< "\n";

    cout << "Pentru CockTail sort: Cocktail"<< "\n";

    cout << "Pentru Insertion sort: Insertion"<< "\n";

    cin >> metoda;

    printf("Tabelul sortat:\n");

    obj.metoda\_select(metoda, a, a\_size);

    return 0;

}

Algoritms.cpp

#include<iostream>

#include <string>

using namespace std;

class sortare

{

public:

    void metoda\_select(string metoda, int \*arr, int size)

    {

        if(metoda=="Bubble"){

            bubbleSort(arr,size);

        }

        else if(metoda=="Cocktail"){

            CocktailSort(arr,size);

        }

        else if(metoda=="Insertion"){

            insertionSort(arr,size);

        }

        else printf("Nu este asa metoda");

    }

    void bubbleSort(int a[], int n)

    {

        int i, j;

        for (i = 0; i < n - 1; i++)

            for (j = 0; j < n - i - 1; j++)

                if (a[j] > a[j + 1])

                    swap(a[j], a[j + 1]);

        printArray(a,n);

    }

    void CocktailSort(int a[], int n)

    {

        bool swapped = true;

        int start = 0;

        int end = n - 1;

        while (swapped) {

            swapped = false;

            for (int i = start; i < end; ++i) {

                if (a[i] > a[i + 1]) {

                    swap(a[i], a[i + 1]);

                    swapped = true;

                }

            }

            if (!swapped)

                break;

            swapped = false;

            --end;

            for (int i = end - 1; i >= start; --i) {

                if (a[i] > a[i + 1]) {

                    swap(a[i], a[i + 1]);

                    swapped = true;

                }

            }

            ++start;

        }

        printArray(a,n);

    }

    void insertionSort(int a[], int n)

    {

        int i, key, j;

        for (i = 1; i < n; i++)

        {

            key = a[i];

            j = i - 1;

            while (j >= 0 && a[j] > key)

            {

                a[j + 1] = a[j];

                j = j - 1;

            }

            a[j + 1] = key;

        }

        printArray(a,n);

    }

    void printArray(int a[], int n)

    {

        for (int i = 0; i < n; i++)

            printf("%d ", a[i]);

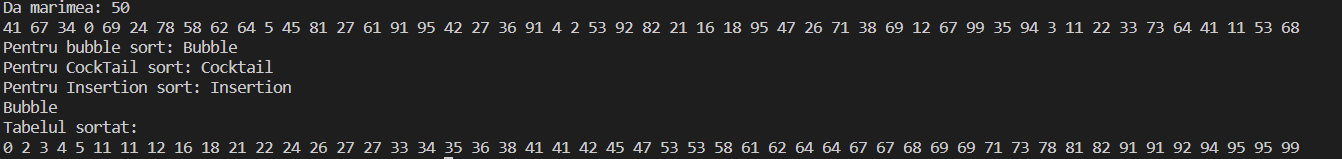
        printf("\n");

    }

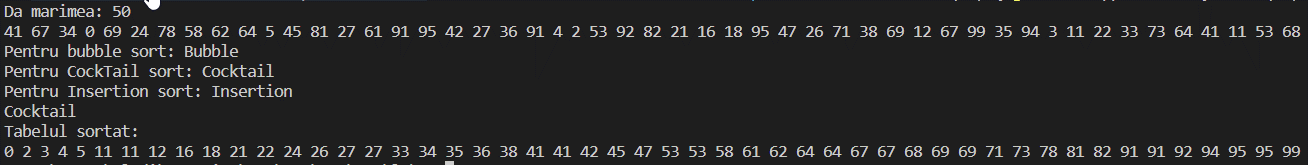
};

Rezultatele:

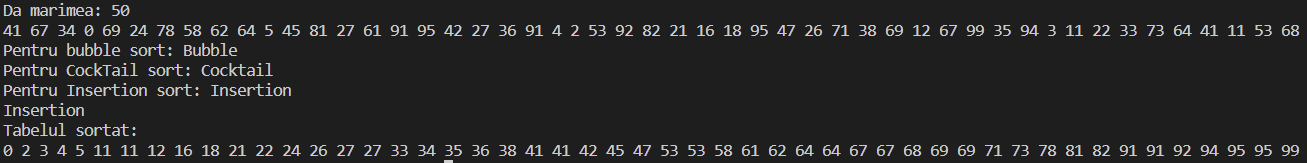
Bubble sort



Cocktail sort



Insertion sort



**Concluzii:**

Deci sa lucrat cu clasele si obiectele in limbajul C++ si sau folosit 3 algoritmi de sortare diferiti, bubble, cocktail si insertion sort, sa lucrat cu elaborarea functiilor, combinarea fisierelor de cod dintre main si fileul cu clasa noastra, generarea unui array folosind rand(). Neam facut cunostiinta cu bazele programarii orientate pe obiecte.