**ЛАБОРАТОРНА РОБОТА № 13-14**

**Сортування одновимірного масиву.**

***Мета:*** набуття практичних навиків роботи з одновимірними масивами, а саме сортування елементів масиву різними методами. Здійснення порівняння та аналізу ефективності використовуваних методів сортування.

**Хід роботи:**

**Завдання**:

1. Згенеруйте одновимірний масив і виведіть його на екран.

2. Відсортуйте масив кожним описаним методом.

3. Порівняйте швидкодію метода сортування при різних значеннях n (кількість елементів масиву) маленьких, середніх, великих. Доведіть розмір масиву до 100000 і більше елементів.

4. Запускаючи програму не менше десяти разів для кожного методу, отримати час t сортування. Побудувати залежність t=f(n) на одному графіку для різних методів сортування (у табличному і графічному виглядах).

5. Які з вивчених методів сортування дають найкращий результат для великих масивів? Який метод найгірший?.

***Сортування методу обміну:***

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <stdlib.h>

#include <time.h>

#include <stdio.h>

#include <locale.h>

#define MAX\_N 100000

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

int tmp, n, is = 0, count;

int arr[MAX\_N];

printf("Введiть кiлькiсть елементiв масиву\n");

scanf\_s("%d", &n);

printf("№Елемента Значення\n");

for (int i = 0; i < n; i++)

{

arr[i] = rand() % 10000;

printf("%d. %d\n", i, arr[i]);

}

printf("Сортований Масив: \n№Елементу\nЗначення\n");

count = n - 1;

clock\_t start, finish;

start = clock();

do

{ is = 0;

for(int i=0;i<count;i++)

if (arr[i] > arr[i + 1])

{

tmp = arr[i];

arr[i] = arr[i + 1];

arr[i + 1] = tmp;

is = 1;

}

count--;

} while (is);

finish = clock();

for (int i = 0; i < n; i++)

printf("%d. %d\n", i, arr[i]);

printf("Час виконання сортування: %f\n", (float)(finish - start) / CLOCKS\_PER\_SEC);

return 0;

}

Результат виконання програми:

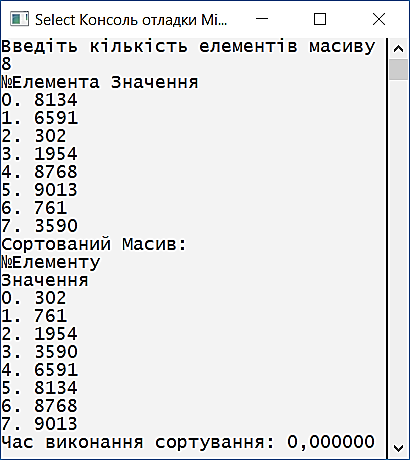


Рис.1(Сортування методом обміну)

***Метод Вибору:***

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <stdlib.h>

#include <time.h>

#include <stdio.h>

#include <locale.h>

#define MAX\_N 100000

int main()

{ setlocale(LC\_ALL, "Rus");

srand(time(NULL));

int tmp, n,min;

int arr[MAX\_N];

printf("Введiть кiлькiсть елементiв масиву\n");

scanf\_s("%d", &n);

printf("№Елемента Значення\n");

for (int i = 0; i < n; i++)

{

arr[i] = rand() % 10000;

printf("%d. %d\n", i, arr[i]);

}

printf("Сортований Масив: \n№Елементу\nЗначення\n");

clock\_t start, finish;

start = clock();

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

{ min = i;

for (int j = i + 1; j < n; j++)

if (arr[j] < arr[min]) min = j;

tmp = arr[i];

arr[i] = arr[min];

arr[min]=tmp;

}

finish = clock();

for (int i = 0; i < n; i++)

printf("%d. %d\n", i, arr[i]);

printf("Час виконання сортування: %f\n", (float)(finish - start) / CLOCKS\_PER\_SEC);

return 0;

}

Результат виконання програми:

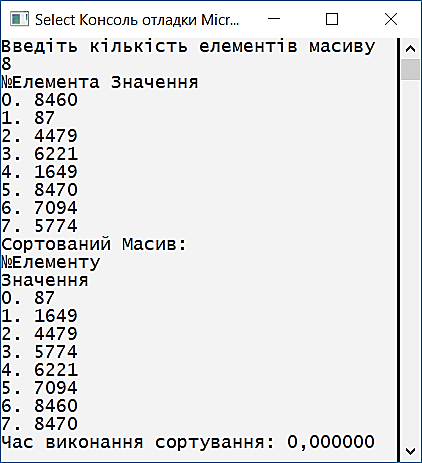


Рис.2(Сортування методом вибору)

***Сортування вставками:***

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <stdlib.h>

#include <time.h>

#include <stdio.h>

#include <locale.h>

#define MAX\_N 100000

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

int tmp, n;

int arr[MAX\_N];

printf("Введiть кiлькiсть елементiв масиву\n");

scanf\_s("%d", &n);

printf("№Елемента Значення\n");

for (int i = 0; i < n; i++)

{

arr[i] = rand() % 10000;

printf("%d. %d\n", i, arr[i]);

}

printf("Сортований Масив: \n№Елементу\nЗначення\n");

clock\_t start, finish;

start = clock();

for (int i = 1; i < n; i++)

{

tmp=arr[i];

for (int j = i - 1; j >= 0 && arr[j] / tmp; j--)

{

arr[j + 1] = arr[j];

arr[j] = tmp;

}

}

finish = clock();

for (int i = 0; i < n; i++)

printf("%d. %d\n", i, arr[i]);

printf("Час виконання сортування: %f\n", (float)(finish - start) / CLOCKS\_PER\_SEC);

return 0;

}

Результат виконання програми:

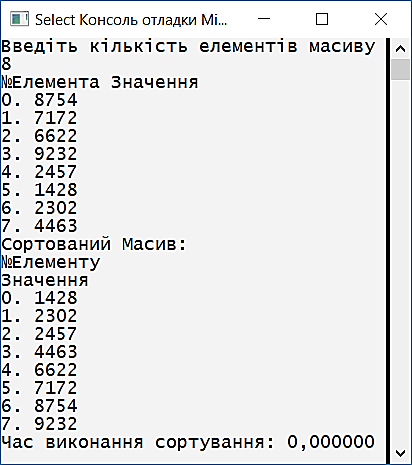


Рис.3(Сортування методом вставки)

***Сортування методом Шелла:***

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <locale.h>

#include <stdlib.h>

#include <time.h>

#include <stdio.h>

#define MAX\_N 100000

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

int tmp, n;

int arr[MAX\_N];

printf("Введiть кiлькiсть елементiв масиву\n");

scanf\_s("%d", &n);

printf("№Елемента Значення\n");

for (int i = 0; i < n; i++)

{ arr[i] = rand() % 10000;

printf("%d. %d\n", i, arr[i]);

}

printf("Сортований Масив: \n№Елементу\nЗначення\n");

clock\_t start, finish;

start = clock();

int step = n / 2;

while (step > 0)

{

for (int i = 1; i < (n - step); i++)

{

int j = i;

while (j >= 0 && arr[j] > arr[j + step])

{

tmp = arr[j];

arr[j] = arr[j + step];

arr[j + step] = tmp;

j--;

}

}

step = step / 2;

}

finish = clock();

for (int i = 0; i < n; i++)

printf("%d. %d\n", i, arr[i]);

printf("Час виконання сортування: %f\n", (float)(finish - start) / CLOCKS\_PER\_SEC);

return 0;

}

Результат виконання програми:

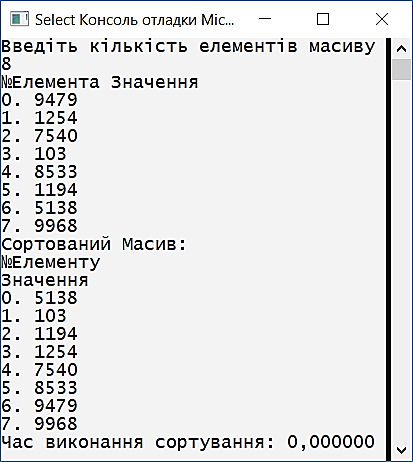
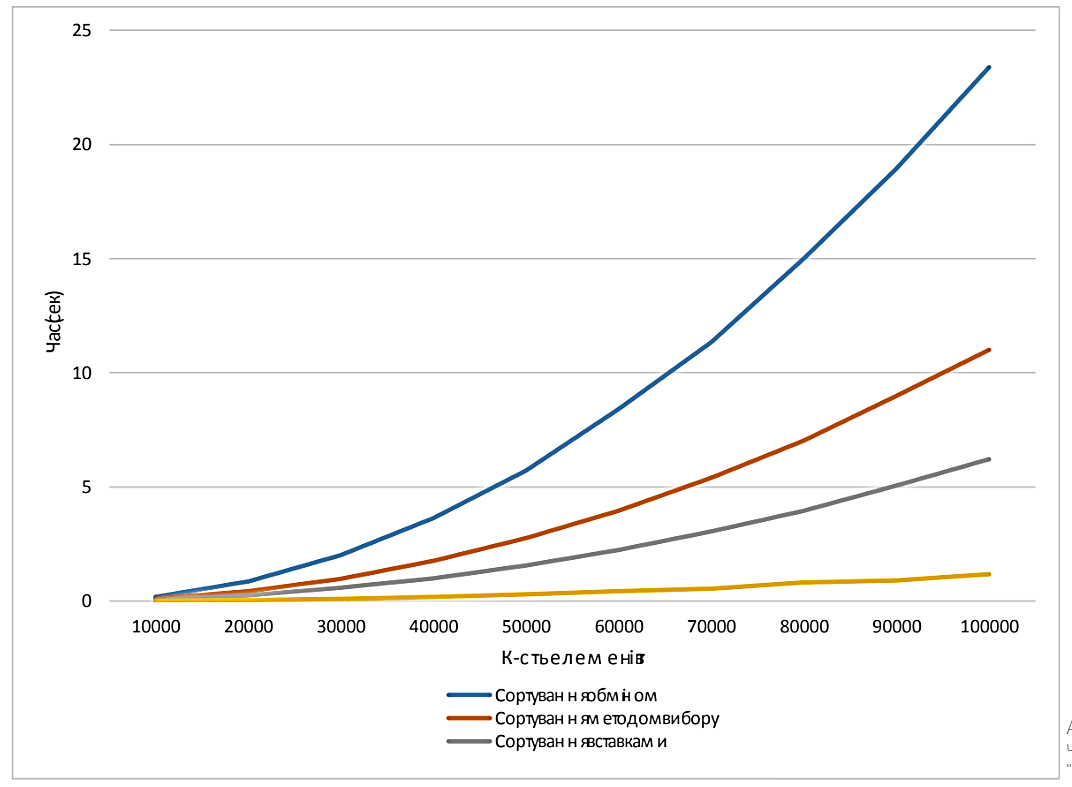


Рис.4(Сортування методом Шелла)

***Завдання на самостійну роботу:***





Тест швидкості сортування масиву різними методами.

Найшвидшим є метод Шелла. Найгіршим – метод обміну.

***Висновок:*** в ході виконання лабораторної роботи набуто практичних навиків роботи з одновимірними масивами, а саме сортування елементів масиву різними методами. Здійснено порівняння та аналізу ефективності використовуваних методів сортування.