

Лабораторна робота № 13

Сортування одновимірного масиву

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

Завдання 1. Відсортуйте масив кожним описаним методом(**Bubble sort**):

Листинг програми:

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>
using namespace std;
int main(){
    srand(time(0));
    float timetaken;
    clock_t start, finish;
    const int NUMBER = 10;
    int array[NUMBER];
    for (int index = 0; index < NUMBER; index++) {
        array[index] = rand() % 20 - 10 + 1;
        printf("%d ", array[index]);
    }
    start = clock();
    for (int j = 0; j < NUMBER; j++) {
        for (int index = 0; index < NUMBER - j + 1; index++) {
            if (array[index] > array[index + 1]) {
                int temp = array[index];
                array[index] = array[index + 1];
                array[index + 1] = temp;
            }
        }
    }
    finish = clock();
    printf("\n");
    for (int index = 0; index < NUMBER; index++) { printf("%d ", array[index]); }
    timetaken = (float)(finish - start) / CLOCKS_PER_SEC;
    printf("\nTime taken (seconds) : %f", timetaken);
    return 0;}
```

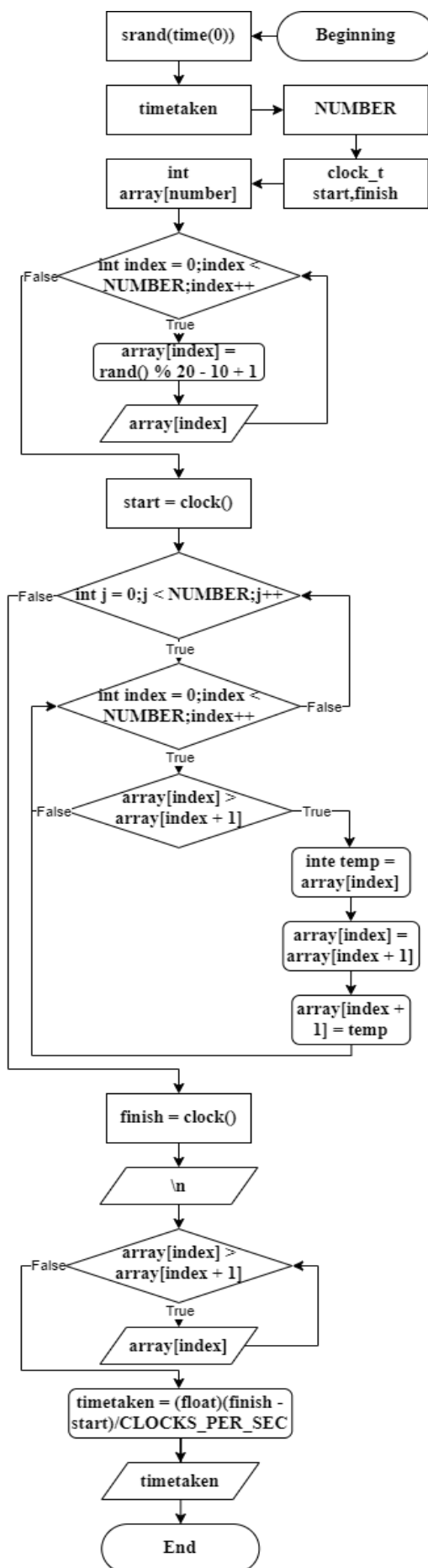
Результат виконання програми:

```
-7 4 4 0 -4 -6 -1 10 2 10
-7 -6 -4 -1 0 2 4 4 10 10
Time taken (seconds) : 0.000000
```

Рис. 1

					ДУ«Житомирська політехніка».23.121.10.000 – Лр13			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Семенчук О.А.			Звіт з лабораторної роботи		Літ.	Арк.
Перевір.		Чижмотря О. В						1
Керівник								11
Н. контр.							ФІКТ Гр. ІПЗ-23-1[2]	
Зав. каф.								

Блок-схема:



Табличний графік(Bubble sort):

Дано масив {2 ,4 ,1 ,7 ,8 ,4 ,1 ,6 ,3 ,10}

2	4	1	7	8	4	1	6	3	10
---	---	---	---	---	---	---	---	---	----

Перше проходження масиву

2	4	1	7	8	4	1	6	3	10
2	1	4	7	8	4	1	6	3	10
2	1	4	7	8	4	1	6	3	10
2	1	4	7	8	4	1	6	3	10
2	1	4	7	4	8	1	6	3	10
2	1	4	7	4	1	8	6	3	10
2	1	4	7	4	1	6	8	3	10
2	1	4	7	4	1	6	3	8	10
2	1	4	7	4	1	6	3	8	10

Друге проходження масиву

1	2	4	7	4	1	6	3	8	10
1	2	4	7	4	1	6	3	8	10
1	2	4	7	4	1	6	3	8	10
1	2	4	4	7	1	6	3	8	10
1	2	4	4	1	7	6	3	8	10
1	2	4	4	1	6	7	3	8	10
1	2	4	4	1	6	3	7	8	10
1	2	4	4	1	6	3	7	8	10

Третє проходження масиву

1	2	4	4	1	6	3	7	8	10
1	2	4	4	1	6	3	7	8	10
1	2	4	4	1	6	3	7	8	10
1	2	4	1	4	6	3	7	8	10
1	2	4	1	4	6	3	7	8	10
1	2	4	1	4	3	6	7	8	10
1	2	4	1	4	3	6	7	8	10

Четверте проходження масиву

1	2	4	1	4	3	6	7	8	10
1	2	4	1	4	3	6	7	8	10
1	2	1	4	4	3	6	7	8	10
1	2	1	4	4	3	6	7	8	10
1	2	1	4	3	4	6	7	8	10
1	2	1	4	3	4	6	7	8	10

П'яте проходження масиву

1	2	1	4	3	4	6	7	8	10
1	1	2	4	3	4	6	7	8	10
1	1	2	4	3	4	6	7	8	10
1	1	2	3	4	4	6	7	8	10
1	1	2	3	4	4	6	7	8	10

Відсортований масив

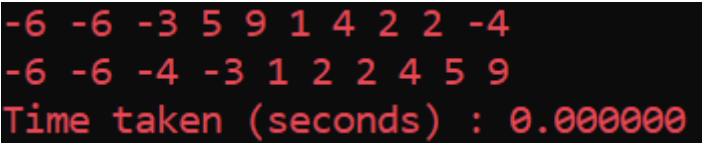
1	1	2	3	4	4	6	7	8	10
---	---	---	---	---	---	---	---	---	----

Завдання 2. Відсортуйте масив кожним описаним методом(Selection sort):

Листинг програми:

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>
int main() {
    srand(time(0));
    float timetaken;
    clock_t start, finish;
    const int NUMBER = 10;
    int array[NUMBER];
    for (int index = 0; index < NUMBER; index++) {
        array[index] = rand() % 20 - 10 + 1;
        printf("%d ", array[index]);
    }
    start = clock();
    for (int index = 0; index < NUMBER; index++) {
        int min = index;
        for (int j = index; j < NUMBER; j++) {
            if (array[j] < array[min]) { min = j; }
        }
        int temp = array[index];
        array[index] = array[min];
        array[min] = temp;
    }
    finish = clock();
    printf("\n");
    for (int index = 0; index < NUMBER; index++) { printf("%d ", array[index]); }
    timetaken = (float)(finish - start) / CLOCKS_PER_SEC;
    printf("\nTime taken (seconds) : %f", timetaken);
    return 0;}
```

Результат виконання програми:

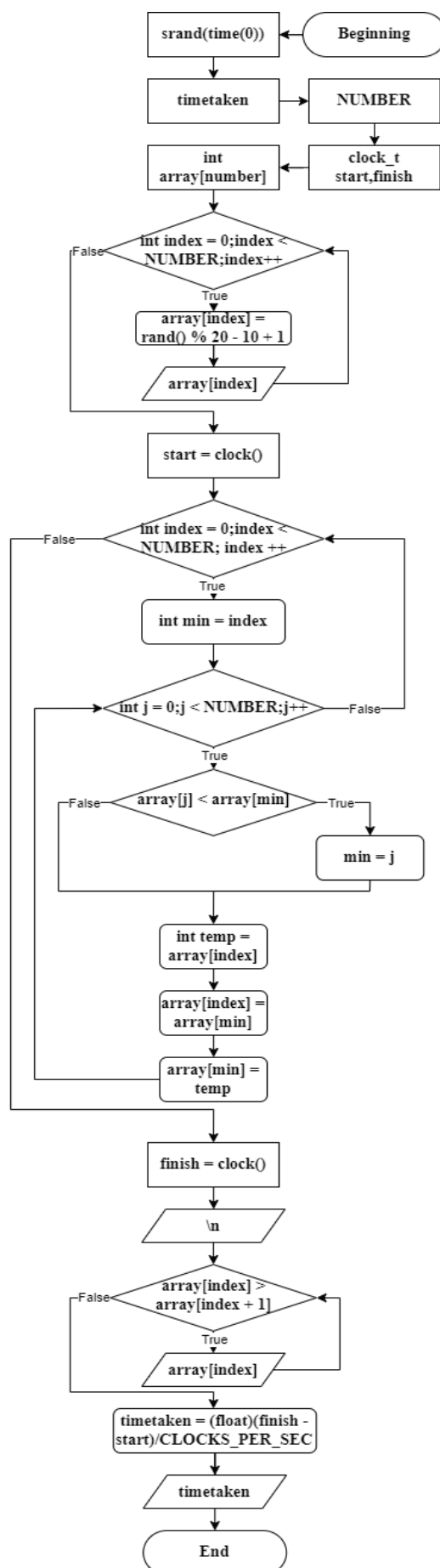


```
-6 -6 -3 5 9 1 4 2 2 -4
-6 -6 -4 -3 1 2 2 4 5 9
Time taken (seconds) : 0.000000
```

Рис 2.

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмотря О. В.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

Блок-схема:



Табличний графік(Selection sort):

Дано масив {2,4,1,7,8,4,1,6,3,10}

2	4	1	7	8	4	1	6	3	10
2	4	1	7	8	4	1	6	3	10
1	4	2	7	8	4	1	6	3	10
1	4	2	7	8	4	1	6	3	10
1	1	2	7	8	4	4	6	3	10
1	1	2	7	8	4	4	6	3	10
1	1	2	7	8	4	4	6	3	10
1	1	2	7	8	4	4	6	3	10
1	1	2	3	8	4	4	6	7	10
1	1	2	3	8	4	4	6	7	10
1	1	2	3	4	8	4	6	7	10
1	1	2	3	4	8	4	6	7	10
1	1	2	3	4	8	4	6	7	10
1	1	2	3	4	4	8	6	7	10
1	1	2	3	4	4	6	8	7	10
1	1	2	3	4	4	6	8	7	10
1	1	2	3	4	4	6	7	8	10
1	1	2	3	4	4	6	7	8	10

Відсортований масив

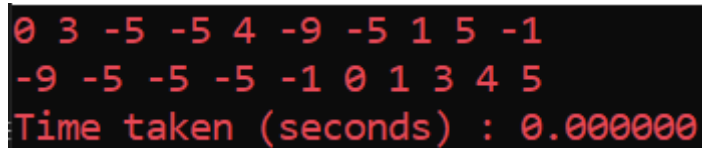
1	1	2	3	4	4	6	7	8	10
---	---	---	---	---	---	---	---	---	----

Завдання 3. Відсортуйте масив кожним описаним методом(Insertion sort):

Листинг програми:

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>
using namespace std;
int main() {
    srand(time(0));
    float timetaken;
    clock_t start, finish;
    const int NUMBER = 100000;
    int array[NUMBER];
    for (int index = 0; index < NUMBER; index++) {
        array[index] = rand() % 20 - 10 + 1;
        printf("%d ", array[index]);
    }
    start = clock();
    for (int index = 1; index < NUMBER; index++) {
        int key = array[index];
        int j = index - 1;
        while (j >= 0 && array[j] > key) {
            array[j + 1] = array[j];
            j = j - 1;
        }
        array[j + 1] = key;
    }
    finish = clock();
    printf("\n");
    for (int index = 0; index < NUMBER; index++) { printf("%d ", array[index]); }
    timetaken = (float)(finish - start) / CLOCKS_PER_SEC;
    printf("\nTime taken (seconds) : %f", timetaken);
    return 0;
}
```

Результат виконання програми:

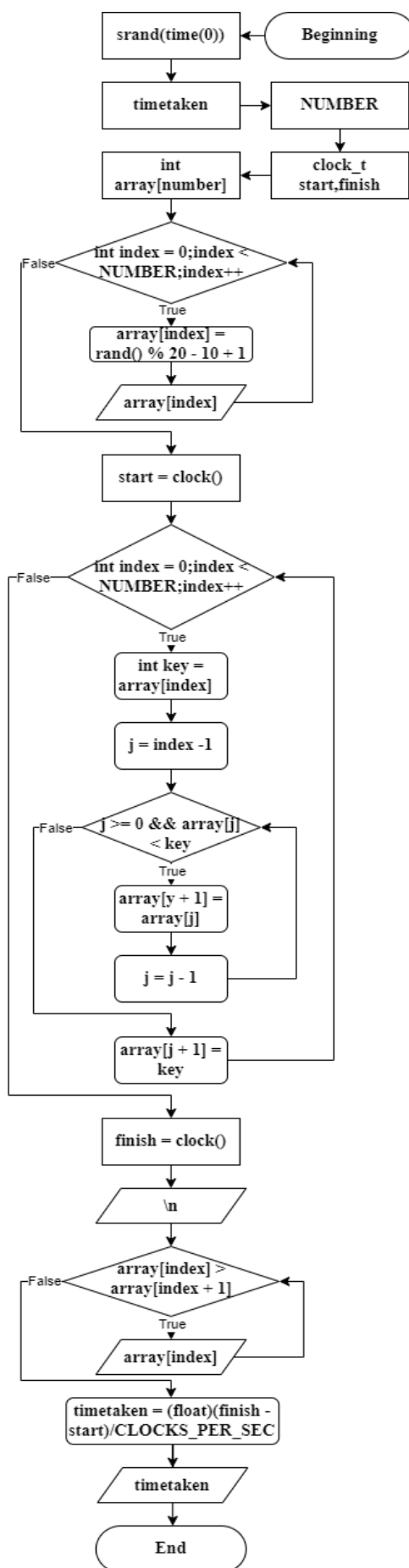


```
0 3 -5 -5 4 -9 -5 1 5 -1
-9 -5 -5 -5 -1 0 1 3 4 5
Time taken (seconds) : 0.000000
```

Рис. 3

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмоторя О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		7

Блок-схема:



Табличний графік(Selection sort):

Дано масив {2 ,4 ,1 ,7 ,8 ,4 ,1 ,6 ,3 ,10}

2	4	1	7	8	4	1	6	3	10
---	---	---	---	---	---	---	---	---	----

Перше проходження масиву

2	4	1	7	8	4	1	6	3	10
2	4	1	7	4	1	6	3	8	10

Друге проходження масиву

2	4	1	7	4	1	6	3	8	10
2	4	1	4	1	6	3	7	8	

Третє проходження масиву

2	4	1	4	1	6	3	7	8	10
2	4	1	4	1	3	6	7	8	10

Четверте проходження масиву

2	4	1	4	1	3	6	7	8	10
2	1	4	1	3	4	6	7	8	10

П'яте проходження масиву

2	1	4	1	3	4	6	7	8	10
2	1	1	3	4	4	6	7	8	10

Шосте проходження масиву

2	1	1	3	4	4	6	7	8	10
2	1	1	3	4	4	6	7	8	10

Сьоме проходження масиву

2	1	1	3	4	4	6	7	8	10
1	1	2	3	4	4	6	7	8	10

Відсортований масив

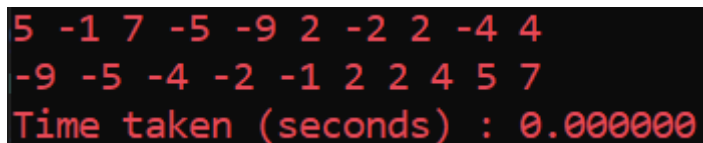
1	1	2	3	4	4	6	7	8	10
---	---	---	---	---	---	---	---	---	----

Завдання 4. Відсортуйте масив кожним описаним методом(Shell sort):

Листинг програми:

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>
using namespace std;
int main() {
    srand(time(0));
    float timetaken;
    clock_t finish, start;
    const int NUMBER = 10;
    int array[NUMBER];
    for (int index = 0; index < NUMBER; index++) {
        array[index] = rand() % 20 - 10 + 1;
        printf("%d ", array[index]);
    }
    start = clock();
    for (int j = NUMBER / 2; j > 0; j = j / 2) {
        for (int index = j; index < NUMBER; index++) {
            int temp = array[index];
            int atom;
            for (atom = index; atom >= j && array[atom - j] > temp; atom = atom - j) {
                array[atom] = array[atom - j];
            }
            array[atom] = temp;
        }
    }
    finish = clock();
    printf("\n");
    for (int index = 0; index < NUMBER; index++) { printf("%d ", array[index]); }
    timetaken = (float)(finish - start) / CLOCKS_PER_SEC;
    printf("\nTime taken (seconds) : %f", timetaken);
    return 0;
}
```

Результат виконання програми:

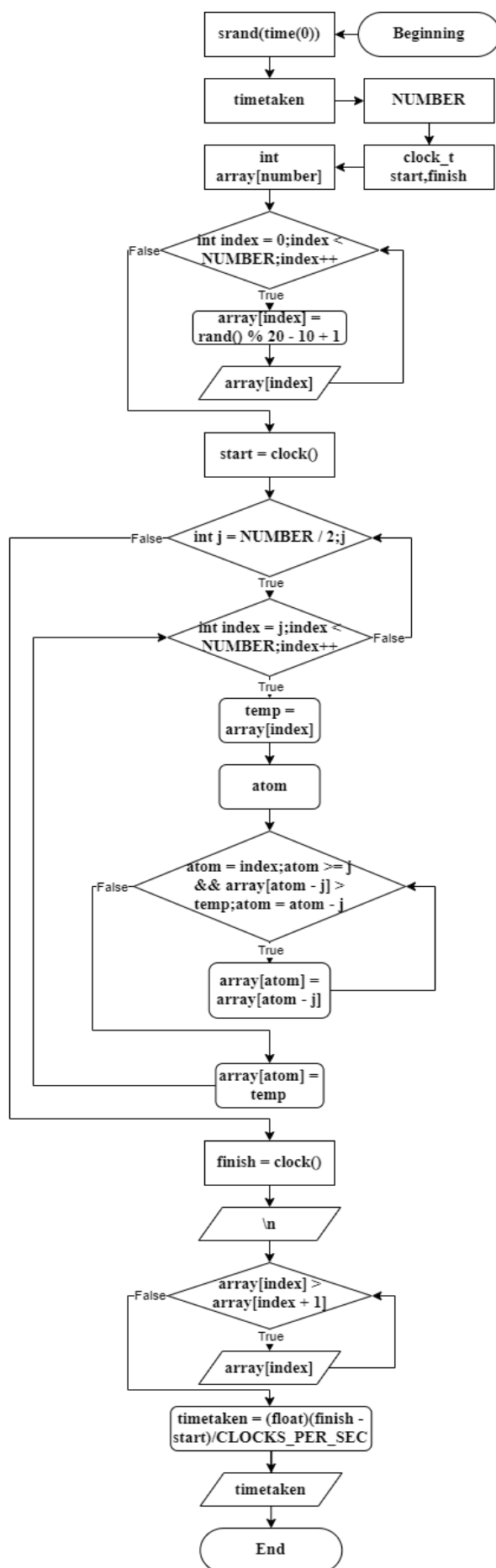


```
5 -1 7 -5 -9 2 -2 2 -4 4
-9 -5 -4 -2 -1 2 2 4 5 7
Time taken (seconds) : 0.000000
```

Рис. 4

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмоторя О. В.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

Блок-схема:



		Семенчук О. А.		
		Чижмоторя О. В.		
Змн.	Арк.	№ докум.	Підпис	Дата

ДУ «Житомирська політехніка».23.121.10.000 – Лр13

Арк.

11

Табличний графік(Selection sort):

Дано масив {2 ,4 ,1 ,7 ,8 ,4 ,1 ,6 ,3 ,10}

2	4	1	7	8	4	1	6	3	10
---	---	---	---	---	---	---	---	---	----

Перше проходження масиву

2	4	1	7	8	4	1	6	3	10
2	1	1	7	8	4	4	6	3	10
2	1	1	7	8	4	4	6	3	10
2	1	1	3	8	4	4	6	7	10
2	1	1	3	8	4	4	6	7	10

Друге проходження масиву

2	1	1	3	8	4	4	6	7	10
2	1	1	3	8	4	4	6	7	10
2	1	1	3	8	4	4	6	7	10
2	1	1	3	8	4	4	6	7	10
2	1	1	3	4	4	8	6	7	10
2	1	1	3	4	4	8	6	7	10
2	1	1	3	4	4	7	6	8	10
2	1	1	3	4	4	7	6	8	10

Третє проходження масиву

1	2	1	3	4	4	7	6	8	10
1	1	2	3	4	4	7	6	8	10
1	1	2	3	4	4	7	6	8	10
1	1	2	3	4	4	7	6	8	10
1	1	2	3	4	4	7	6	8	10
1	1	2	3	4	4	7	6	8	10
1	1	2	3	4	4	6	7	8	10
1	1	2	3	4	4	6	7	8	10
1	1	2	3	4	4	6	7	8	10

Відсортований масив

1	1	2	3	4	4	6	7	8	10
---	---	---	---	---	---	---	---	---	----

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13				Арк.
		Чижмоторя О. В.							12
Змн.	Арк.	№ докум.	Підпис	Дата					

Порівняння:

10 elements

	Sorting algorithm			
Time	0 (Bubble)	0 (Selection)	0 (Insertion)	0 (Shell)

Fastest: Bubble, Selection, Insertion, Shell;

Slowest:

100 elements

	Sorting algorithm			
Time	0 (Bubble)	0 (Selection)	0 (Insertion)	0 (Shell)

Fastest: Bubble, Selection, Insertion, Shell;

Slowest:

1000 elements

	Sorting algorithm			
Time	0.000200 (Insertion)	0.000400 (Selection)	0.000500 (Shell)	0.001600 (Bubble)

Fastest: Insertion;

Slowest: Bubble;

10.000 elements

	Sorting algorithm			
Time	0.002100 (Shell)	0.098800 (Insertion)	0.155900 (Selection)	0.319900 (Bubble)

Fastest: Shell;

Slowest: Bubble;

100.000 elements

	Sorting method			
Time	0.018900 (Shell)	8.728700 (Insertion)	15.523599 (Selection)	35.717102 (Bubble)

Fastest: Shell;

Slowest: Bubble;

Overall:

Fastest >	Shell	Insertion	Selection	Bubble	> Slowest
-----------	-------	-----------	-----------	--------	-----------

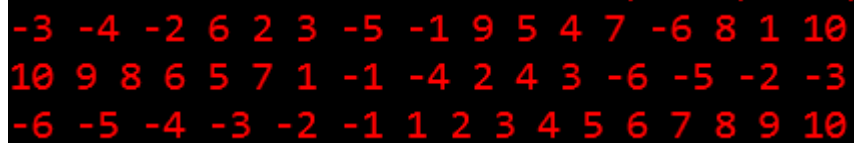
		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмоторя О. В.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдання на самостійну роботу(Heap sort):

Листинг програми:

```
#include <stdio.h>
void heapify(int array[], int length, int index) {
    int largest = index;
    int left = 2 * index + 1;
    int right = 2 * index + 2;
    if (left < length && array[left] > array[largest]) {largest = left;}
    if (right < length && array[right] > array[largest]) {largest = right;}
    if (largest != index){
        int temp = array[index];
        array[index] = array[largest];
        array[largest] = temp;
        heapify(array, length, largest);}
}
void heapsort(int array[], int length){
    for (int index = length / 2 - 1; index >= 0; index--){heapify(array, length, index);}
    for (int index = length - 1; index >= 0; index--){
        int temp = array[0];
        array[0] = array[index];
        array[index] = temp;
        heapify(array, index, 0);}
}
int main() {
    int array[] = {-3, -4, -2, 6, 2, 3, -5, -1, 9, 5, 4, 7, -6, 8, 1, 10};
    int length = 16;
    for (int index = 0; index < length; index++){printf("%d ", array[index]);}
    for (int index = length / 2 - 1; index >= 0; index--){heapify(array, length, index);}
    printf("\n");
    for (int index = 0; index < length; index++){printf("%d ", array[index]);}
    heapsort(array, length);
    printf("\n");
    for (int index = 0; index < length; index++){printf("%d ", array[index]);}
    return 0;}
```

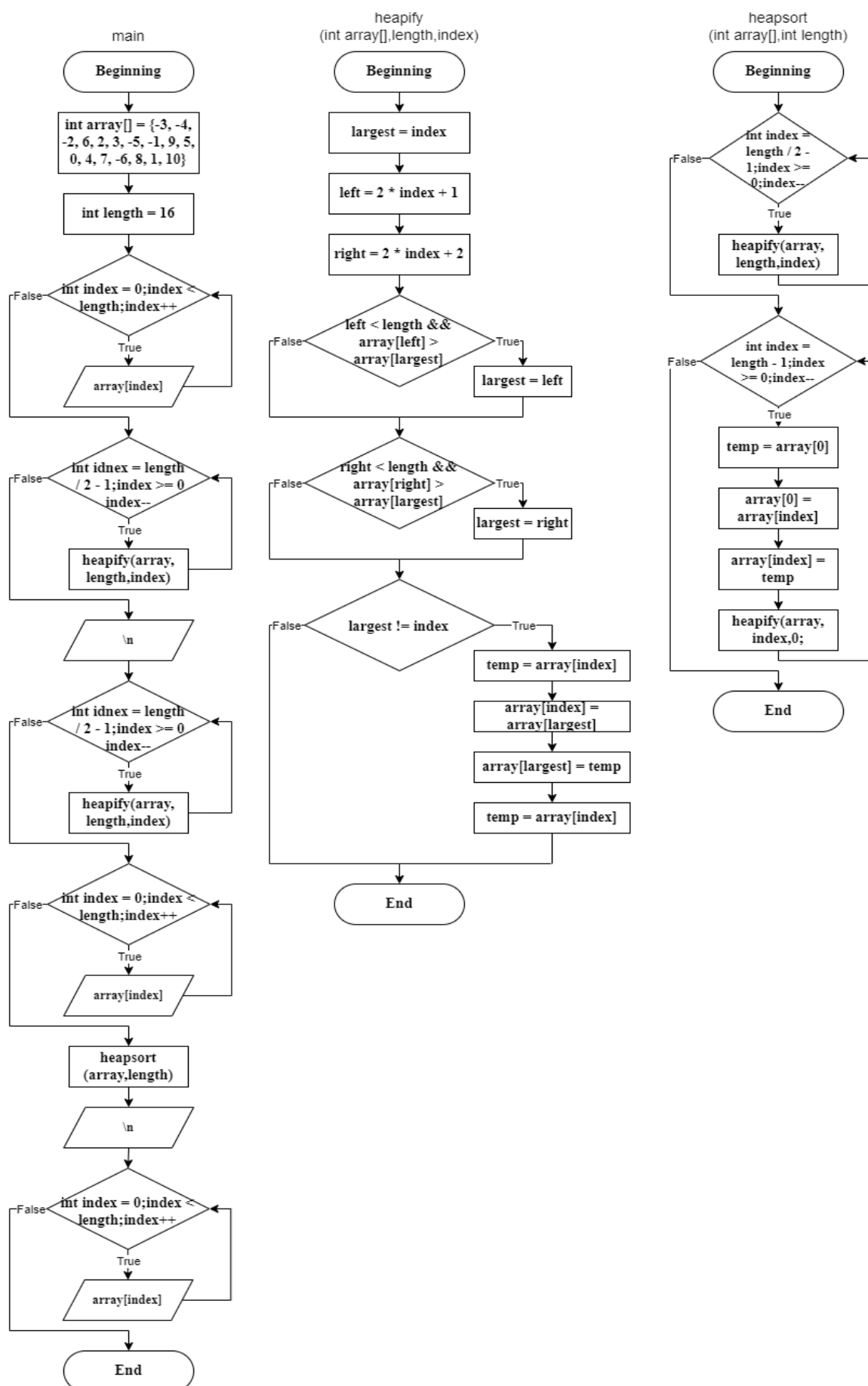
Результат виконання програми:



```
-3 -4 -2 6 2 3 -5 -1 9 5 4 7 -6 8 1 10
10 9 8 6 5 7 1 -1 -4 2 4 3 -6 -5 -2 -3
-6 -5 -4 -3 -2 -1 1 2 3 4 5 6 7 8 9 10
```

Рис. 5

Блок-схема:



Табличний графік(Heap sort):

Дано масив {-3, -4, -2, 6, 2, 3, -5, -1, 9, 5, 4, 7, -6, 8, 1, 10}

-3	-4	-2	6	2	3	-5	-1	9	5	4	7	-6	8	1	10
----	----	----	---	---	---	----	----	---	---	---	---	----	---	---	----

-3	-4	-2	6	2	3	-5	-1	9	5	4	7	-6	8	1	10
-3	-4	-2	6	2	3	-5	10	9	5	4	7	-6	8	1	-1
-3	-4	-2	6	2	3	-5	10	9	5	4	7	-6	8	1	-1
-3	-4	-2	6	2	3	8	10	9	5	4	7	-6	-5	1	-1
-3	-4	-2	6	2	3	8	10	9	5	4	7	-6	-5	1	-1
-3	-4	-2	6	2	7	8	10	9	5	4	3	-6	-5	1	-1
-3	-4	-2	6	2	7	8	10	9	5	4	3	-6	-5	1	-1
-3	-4	-2	6	5	7	8	10	9	2	4	3	-6	-5	1	-1
-3	-4	-2	6	5	7	8	10	9	2	4	3	-6	-5	1	-1
-3	-4	-2	10	5	7	8	6	9	2	4	3	-6	-5	1	-1
-3	-4	-2	10	5	7	8	6	9	2	4	3	-6	-5	1	-1
-3	-4	8	10	5	7	-2	6	9	2	4	3	-6	-5	1	-1
-3	10	8	-4	5	7	-2	6	9	2	4	3	-6	-5	1	-1
-3	10	8	-4	5	7	-2	6	9	2	4	3	-6	-5	1	-1
10	-3	8	-4	5	7	-2	6	9	2	4	3	-6	-5	1	-1
-1	-3	8	-4	5	7	-2	6	9	2	4	3	-6	-5	1	10
-1	-3	8	-4	5	7	-2	6	9	2	4	3	-6	-5	1	10
-1	-3	8	-4	5	7	1	6	9	2	4	3	-6	-5	-2	10
-1	-3	8	-4	5	7	1	6	9	2	4	3	-6	-5	-2	10
-1	-3	8	-4	5	7	1	6	9	2	4	3	-6	-5	-2	10
-1	-3	8	9	5	7	1	6	-4	2	4	3	-6	-5	-2	10
-1	-3	8	9	5	7	1	6	-4	2	4	3	-6	-5	-2	10
-1	-3	8	9	5	7	1	6	-4	2	4	3	-6	-5	-2	10

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижомотря О. В.				16
Змн.	Арк.	№ докум.	Підпис	Дата		

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмотря О. В.				17
Змн.	Арк.	№ докум.	Підпис	Дата		

-6	6	3	-1	5	-5	1	-3	-4	2	4	-2	7	8	9	10
-6	6	3	-1	5	-5	1	-3	-4	2	4	-2	7	8	9	10
-6	6	3	-1	5	-5	1	-3	-4	2	4	-2	7	8	9	10
6	-6	3	-1	5	-5	1	-3	-4	2	4	-2	7	8	9	10
-2	-6	3	-1	5	-5	1	-3	-4	2	4	6	7	8	9	10
-2	-6	3	-1	5	-5	1	-3	-4	2	4	6	7	8	9	10
-2	-6	3	-1	5	-5	1	-3	-4	2	4	6	7	8	9	10
-2	-6	3	-1	5	-5	1	-3	-4	2	4	6	7	8	9	10
-2	-6	3	-1	5	-5	1	-3	-4	2	4	6	7	8	9	10
-2	-6	3	-1	5	-5	1	-3	-4	2	4	6	7	8	9	10
-2	5	3	-1	-6	-5	1	-3	-4	2	4	6	7	8	9	10
-2	5	3	-1	-6	-5	1	-3	-4	2	4	6	7	8	9	10
5	-2	3	-1	-6	-5	1	-3	-4	2	4	6	7	8	9	10
4	-2	3	-1	-6	-5	1	-3	-4	2	5	6	7	8	9	10
4	-2	3	-1	-6	-5	1	-3	-4	2	5	6	7	8	9	10
4	-2	3	-1	2	-5	1	-3	-4	-6	5	6	7	8	9	10
4	-2	3	-1	2	-5	1	-3	-4	-6	5	6	7	8	9	10
4	-2	3	-1	2	-5	1	-3	-4	-6	5	6	7	8	9	10
4	-2	3	-1	2	-5	1	-3	-4	-6	5	6	7	8	9	10
4	2	3	-1	-2	-5	1	-3	-4	-6	5	6	7	8	9	10
4	2	3	-1	-2	-5	1	-3	-4	-6	5	6	7	8	9	10
4	2	3	-1	-2	-5	1	-3	-4	-6	5	6	7	8	9	10
4	2	3	-1	-2	-5	1	-3	-4	-6	5	6	7	8	9	10
-6	2	3	-1	-2	-5	1	-3	-4	4	5	6	7	8	9	10
-6	2	3	-1	-2	-5	1	-3	-4	4	5	6	7	8	9	10
-6	2	3	-1	-2	-5	1	-3	-4	4	5	6	7	8	9	10
-6	2	3	-1	-2	-5	1	-3	-4	4	5	6	7	8	9	10
-6	2	3	-1	-2	-5	1	-3	-4	4	5	6	7	8	9	10
3	2	-6	-1	-2	-5	1	-3	-4	4	5	6	7	8	9	10
-4	2	-6	-1	-2	-5	1	-3	3	4	5	6	7	8	9	10
-4	2	-6	-1	-2	-5	1	-3	3	4	5	6	7	8	9	10
-4	2	-6	-1	-2	-5	1	-3	3	4	5	6	7	8	9	10

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмоторя О. В.				18
Змн.	Арк.	№ докум.	Підпис	Дата		

-4	2	1	-1	-2	-5	-6	-3	3	4	5	6	7	8	9	10
-4	2	1	-1	-2	-5	-6	-3	3	4	5	6	7	8	9	10
-4	2	1	-1	-2	-5	-6	-3	3	4	5	6	7	8	9	10
2	-4	1	-1	-2	-5	-6	-3	3	4	5	6	7	8	9	10
-3	-4	1	-1	-2	-5	-6	2	3	4	5	6	7	8	9	10
-3	-4	1	-1	-2	-5	-6	2	3	4	5	6	7	8	9	10
-3	-4	1	-1	-2	-5	-6	2	3	4	5	6	7	8	9	10
-3	-1	1	-4	-2	-5	-6	2	3	4	5	6	7	8	9	10
-3	-1	1	-4	-2	-5	-6	2	3	4	5	6	7	8	9	10
1	-1	-3	-4	-2	-5	-6	2	3	4	5	6	7	8	9	10
-6	-1	-3	-4	-2	-5	1	2	3	4	5	6	7	8	9	10
-6	-1	-3	-4	-2	-5	1	2	3	4	5	6	7	8	9	10
-6	-1	-3	-4	-2	-5	1	2	3	4	5	6	7	8	9	10
-6	-1	-3	-4	-2	-5	1	2	3	4	5	6	7	8	9	10
-1	-6	-3	-4	-2	-5	1	2	3	4	5	6	7	8	9	10
-5	-6	-3	-4	-2	-1	1	2	3	4	5	6	7	8	9	10
-5	-6	-3	-4	-2	-1	1	2	3	4	5	6	7	8	9	10
-5	-2	-3	-4	-6	-1	1	2	3	4	5	6	7	8	9	10
-5	-2	-3	-4	-6	-1	1	2	3	4	5	6	7	8	9	10
-2	-5	-3	-4	-6	-1	1	2	3	4	5	6	7	8	9	10
-6	-5	-3	-4	-2	-1	1	2	3	4	5	6	7	8	9	10
-6	-5	-3	-4	-2	-1	1	2	3	4	5	6	7	8	9	10
-6	-4	-3	-5	-2	-1	1	2	3	4	5	6	7	8	9	10
-6	-4	-3	-5	-2	-1	1	2	3	4	5	6	7	8	9	10
-3	-4	-6	-5	-2	-1	1	2	3	4	5	6	7	8	9	10
-5	-4	-6	-3	-2	-1	1	2	3	4	5	6	7	8	9	10
-5	-4	-6	-3	-2	-1	1	2	3	4	5	6	7	8	9	10
-4	-5	-6	-3	-2	-1	1	2	3	4	5	6	7	8	9	10
-6	-5	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмоторя О. В.				19
Змн.	Арк.	№ докум.	Підпис	Дата		

-6	-5	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10
-4	-6	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10
-6	-5	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10
-6	-5	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр13	Арк.
		Чижмотря О. В.				20
Змн.	Арк.	№ докум.	Підпис	Дата		