

САА – Упражнение 10

1) Алгоритъм за сортиране чрез пряк избор

```
void selection_sort(int A[], int n)
{
    int i, j, x, min;
    for (i = 0; i < n-1; i++)
    {
        min = i;
        for (j = i+1; j <= n-1; j++)
            if (A[j] < A[min]) min = j;
        if (min != i) {x = A[min]; A[min] = A[i]; A[i] = x;}
    }
}
```

2) Алгоритъм за сортиране чрез пряко вмъкване

```
void insertion_sort(int A[], int n)
{
    int i, j, x;
    for (i = 1; i < n; i++)
    {
        x = A[i];
        j = i - 1;
        while (j >= 0 && x < A[j])
        {
            A[j+1] = A[j];
            j--;
        }
        A[j+1] = x;
    }
}
```

3) Алгоритъм за сортиране чрез пряка размяна

```
void bubble_sort(int A[], int n)
{
    int i, j, x;
    for (i = 0; i < n-1; i++)
        for (j = n-1; j > i; j--)
            if (A[j] < A[j-1])
            {
                x = A[j];
                A[j] = A[j-1];
                A[j-1] = x;
            }
}
```

4) Алгоритъм за сортиране чрез пряка размяна с флаг

```
void bubble_sort_flag( int A[], int n)
{
    int i, x, flag;
    do
    {
        flag = 1;
        for(i=0; i<n-1; i++)
            if (A[i]>A[i+1])
            {
                x =A[i];
```

```

        A[i] = A[i+1];
        A[i+1] = x;
        flag = 0;
    }
    while (!flag);
}

```

5) Алгоритъм за сортиране чрез сливане

6) Алгоритъм за бързо сортиране

7) Цифрова сортировка

Задачи:

1. Да се реализира конзолно приложение, което сравнява времето за изпълнение на всеки от описаните по-горе седем алгоритъма за сортиране. За целта да се използва масив от случайно генерирани числа в интервала (0, 10000) с:

- A) 1000 елемента;
- Б) 2000 елемента;
- В) 5000 елемента;
- Г) 10000 елемента;
- Д) 20000 елемента;
- Е) 50000 елемента.

Упътване:

- За отчитане на времето за изпълнение на всеки от алгоритмите да се използва функцията GetTickCount(), която връща като резултат времето в милисекунди от началото на стартиране на системата (библиотека windows.h).
 - В главната функция, преди и след извикване на съответната функция за сортиране, да се извика функцията GetTickCount() и време за изпълнение на съответния алгоритъм за сортиране да се определи като разликата от двете стойности.
2. Да се анализират и обобщат резултатите, получени от проведените според задача 1 експерименти и да се направят изводи относно бързодействието на всеки от сравняваните алгоритми за сортиране.