НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка массивов

Формулировка задачи:

- Дан массив A из n элементов:
  a[0], a[1], a[2],… a[n-1]
- Требуется переставить элементы этого массива так, чтобы после перестановки они были упорядочены по неубыванию:
  $a[0] \le a[1] \le a[2] \le \ldots \le a[n-1]$

Примечание. Хотя, в нашем случае, массивы имеют не очень большой размер, будем считать, что заводить вспомогательный массив недопустимо – расходуется “лишняя память”

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка массивов

Алгоритмы сортировки:

- простые
- усовершенствованные

Мы рассмотрим простые методы:

- Сортировка выбором
- Сортировка обменами
- Сортировка вставками

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка обменами

Шаги алгоритма:

- Алгоритм состоит из повторяющихся проходов по сортируемому массиву.
- За каждый проход элементы последовательно сравниваются попарно и, если порядок в паре неверный, выполняется обмен элементов.
- Проходы по массиву повторяются N-1 раз или до тех пор, пока на очередном проходе не окажется, что обмены больше не нужны, что означает — массив отсортирован.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка обменами

Шаг 1:	7	15	33	42	12	19
Шаг 2:	7	12	15	33	42	19
Шаг 3:	7	12	15	19	33	42
Шаг 4:	7	12	15	19	33	42
Шаг 5:	7	12	15	19	33	42
	7	12	15	19	33	42

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка массивов

```
#include <iostream>

using namespace std;

void bubbleSort(int a[], int size);

void printArr(int a[], int size)
{
    for(int i=0; i<size; i++) {
        cout << a[i] << " ";
    }
    cout << "\n";
}

int main()
{
    int arr[] = {1, -5, 7, -2, 0, 11, 6, 8, -3, -4};
    printArr(arr, 10);
    bubbleSort(arr, 10);
    printArr(arr, 10);
    return 0;
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка обменами

```
void bubbleSort(int a[], int size)
{
    for (int i=0; i< size; i++) {
        bool swap = false;
        for (int j= size-1; j>i; j--){
            if (a[j] < a[j-1]) {
                int t = a[j];
                a[j] = a[j-1];
                a[j-1] = t;
                swap = true;
            }
        }
        if (!swap) break;
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка вставками

Шаги алгоритма:

- Выбираем один из элементов массива и вставляем его на нужную позицию в уже отсортированной части массива
- Повторяем до тех пор, пока не достигнут конец массива.
- Метод выбора очередного элемента из исходного массива произволен; может использоваться практически любой алгоритм выбора. Обычно, элементы вставляются по порядку их появления во входном массиве.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка вставками

Шаг 1:

15	33	42	07	12	19

Шаг 2:

15	33	42	07	12	19

Шаг 3:

15	33	7	42	12	19
7	15	33	42	12	19

Шаг 4:

7	15	33	42	12	19
7	12	15	33	42	19

т.к. на 3-й позиции было min число

Шаг 5:

7	12	15	33	42	19
7	12	15	19	33	42

т.к. на 4-й позиции было min число

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка массивов

```
#include <iostream>

using namespace std;

void insertSort(int a[], int size);

void printArr(int a[], int size)
{
    for(int i=0; i<size; i++) {
        cout << a[i] << " ";
    }
    cout << "\n";
}

int main()
{
    int arr[] = {1, -5, 7, -2, 0, 11, 6, 8, -3, -4};
   printArr(arr, 10);
    insertSort(arr, 10);
    printArr(arr, 10);
    return 0;
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка вставками

```
void insertSort(int a[], int size)
{
    for (int i=1; i<size; i++)
    {
        int curr = a[i];
        int j = i-1;
        while (j>=0 && a[j]>curr) {
            a[j+1]=a[j];
            j--;
        }
        a[j+1] = curr;
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка вставками

Демонстрация

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка выбором

Шаги алгоритма:

- находим номер минимального значения в массиве
- производим обмен этого значения со значением первой неотсортированной позиции (обмен не нужен, если минимальный элемент уже находится на данной позиции)
- теперь сортируем хвост массива, исключив из рассмотрения уже отсортированные элементы

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка массивов

```
#include <iostream>

using namespace std;

void chooseSort(int a[], int size);

void printArr(int a[], int size)
{
    for(int i=0; i<size; i++) {
        cout << a[i] << " ";
    }
    cout << "\n";
}

int main()
{
    int arr[] = {1, -5, 7, -2, 0, 11, 6, 8, -3, -4};
    chooseSort(arr, 10);
    printArr(arr, 10);
    return 0;
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка выбором

```
void chooseSort(int a[], int size)
{
    for (int i=0; i<size-1; i++) {
        int nmin = i;
        for (int j=i+1; j<size; j++){
            if (a[j]<a[nmin]) nmin = j;
        }
        if (i != nmin) {
            int t = a[nmin];
            a[nmin] = a[i];
            a[i] = t;
        }
    }
}
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Сортировка выбором

Демонстрация

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Спасибо!

Вопросы?