

Data Structures and Algorithms

Алгоритмы. Сортировка выбором.



Сведение о алгоритме

Алгоритм сортировки выбором.

Сложность по времени в наихудшем случае $O(n^2)$

Затраты памяти $O(n)$

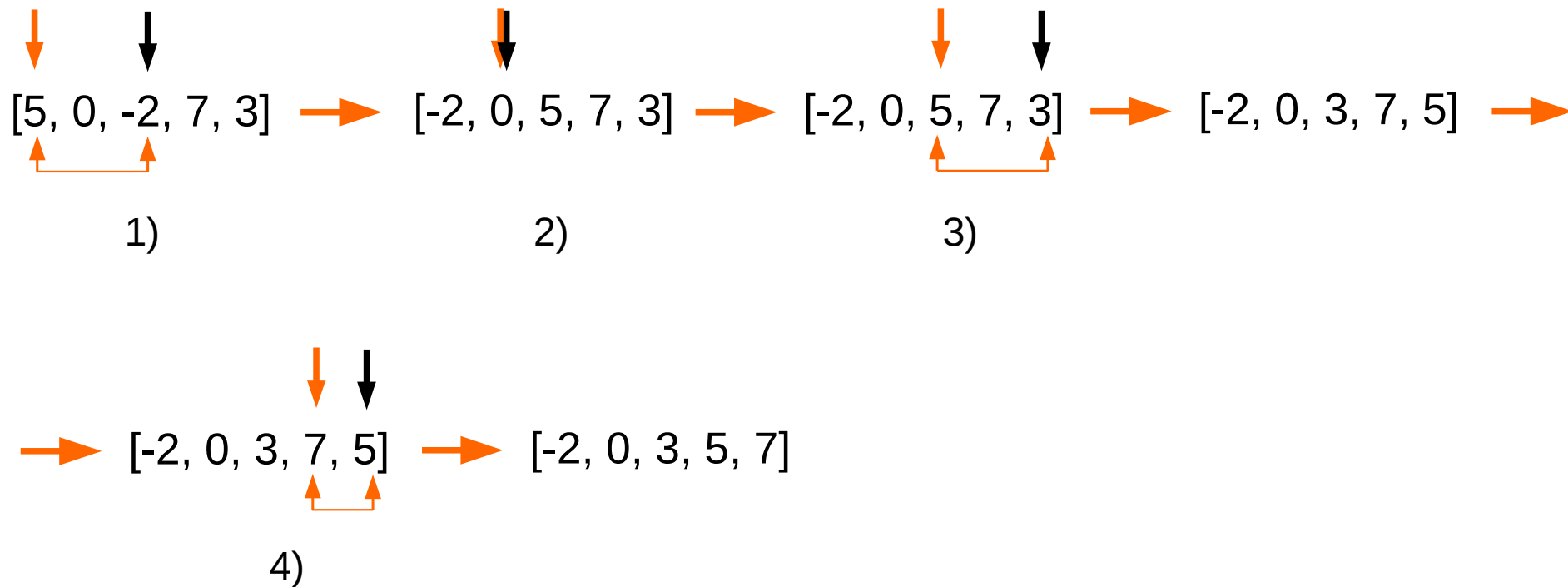


Принцип работы алгоритма

- 1) Выполняется проход по сортируемой последовательности.
- 2) Первый элемент не отсортированной подпоследовательности принимается за минимальный и сохраняется его индекс.
- 3) Выполняется проход по не отсортированной части последовательности в поиске минимального элемента.
- 4) Происходит обмен найденного минимального члена и первого члена не отсортированной подпоследовательности.

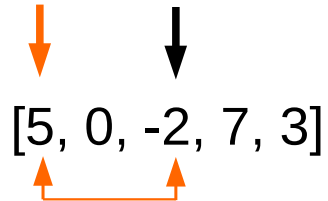


Графическая иллюстрация работы алгоритма





Графическая иллюстрация работы алгоритма



1)

На первом шаге, вся последовательность является не отсортированной. Выполняется поиск минимального элемента на всей последовательности. В примере это число -2. Потом производится обмен первого члена не сортированной подпоследовательности и найденного минимума.



Графическая иллюстрация работы алгоритма



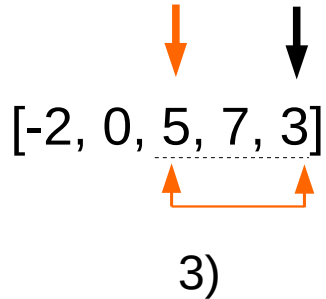
[-2, 0, 5, 7, 3]

2)

На втором шаге опять производится поиск минимума в не отсортированной подпоследовательности (в примере она подчеркнута). В этом примере минимальный элемент совпадает с первым элементом. Поэтому обмен не производится.



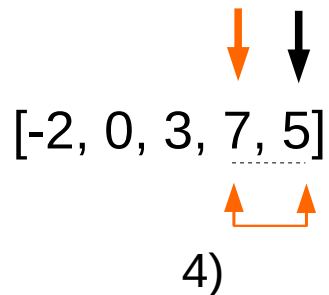
Графическая иллюстрация работы алгоритма



И опять производится поиск минимума в не отсортированной подпоследовательности (в примере она подчеркнута). В этом примере минимальный элемент равен 3. Производим обмен с первым элементом не отсортированной подпоследовательности.



Графическая иллюстрация работы алгоритма



И опять производится поиск минимума в не отсортированной подпоследовательности (в примере она подчеркнута). В этом примере минимальный элемент равен 5. Производим обмен с первым элементом не отсортированной подпоследовательности.



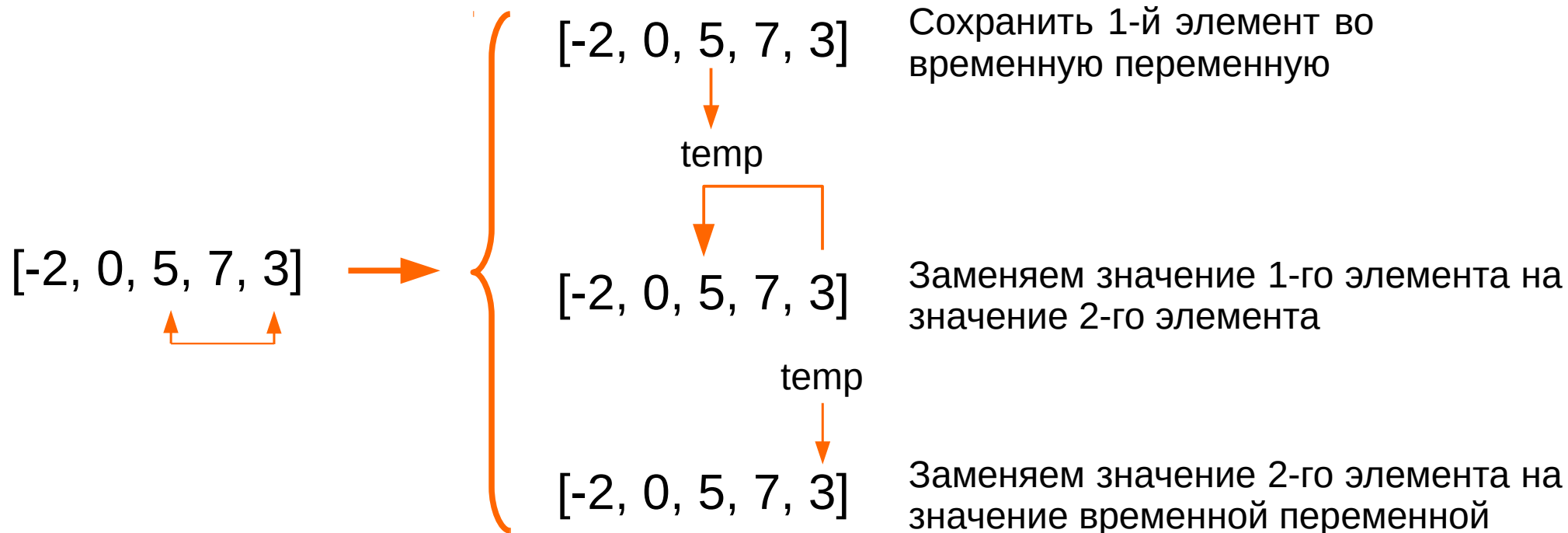
Графическая иллюстрация работы алгоритма

[-2, 0, 3, 5, 7]

Длина не отсортированной подпоследовательности равна 0. Следовательно алгоритм считается законченным. И действительно данные массива расположены в возрастающей последовательности.



Как выполнить обмен двух элементов последовательности





Реализация алгоритма на Python



Реализация алгоритма на Python

```
list_1 = [5, 0, -2, 7, 3]
```

```
for i in range(0, len(list_1)-1):
```

 ← Проход по все индексам последовательности

```
    min_index = i
```

 ← Индекс минимального элемента

```
        for j in range(i+1, len(list_1)):
```

 ← Проход по индексам не отсортированной подпоследовательности

```
            if list_1[min_index] > list_1[j]:
```

 ← Поиск минимального члена подпоследовательности

```
                min_index = j
```

```
        if min_index != i:
```

```
            temp = list_1[i]
```

```
            list_1[i] = list_1[min_index]
```

```
            list_1[min_index] = temp
```

 ← Обмен элементов

```
print(list_1)
```



Java

Реализация алгоритма на Java



Реализация алгоритма на Java

```
int[] array = new int[] { 5, 0, -2, 7, 3 };
```

```
for (int i = 0; i < array.length - 1; i++) { ← Проход по все индексам последовательности
```

```
    int minIndex = i; ← Индекс минимального элемента
```

```
    for (int j = i + 1; j < array.length; j++) { ← Проход по индексам не отсортированной подпоследовательности
```

```
        if (array[minIndex] > array[j]) { } ← Поиск минимального члена подпоследовательности  
            minIndex = j;  
        }
```

```
    }  
    if (minIndex != i) { } ← Обмен элементов  
        int temp = array[i];  
        array[i] = array[minIndex];  
        array[minIndex] = temp;  
    }
```

```
System.out.println(Arrays.toString(array));
```



Список литературы

- 1) Ананий Левитин. Алгоритмы: введение в разработку и анализ. : Пер. с англ. — М. : Издательский дом "Вильямс", 2006. — 576 с. : ил. — Парал. тит. Англ. ISBN 5-8459-0987-2. Стр. [143-144]
- 2) Стивенсон Род. Алгоритмы. Теория и практическое применение — М: Издательство «Э», 2016 — 544. ISBN 978-5-699-81729-0. Стр. [138-139]