

Data Structures and Algorithms

Алгоритмы. Гномья сортировка.



При чем тут гномы???

Гномья сортировка основана на технике, используемой обычным голландским садовым гномом (нидерл. *tuinkabouter*). Это метод, которым садовый гном сортирует линию цветочных горшков. По существу он смотрит на текущий и предыдущий садовые горшки: если они в правильном порядке, он шагает на один горшок вперёд, иначе он меняет их местами и шагает на один горшок назад. Граничные условия: если нет предыдущего горшка, он шагает вперёд; если нет следующего горшка, он закончил.

Дик Грун



Сведение о алгоритме

Алгоритм гномьей сортировки.

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

Затраты памяти $O(n)$

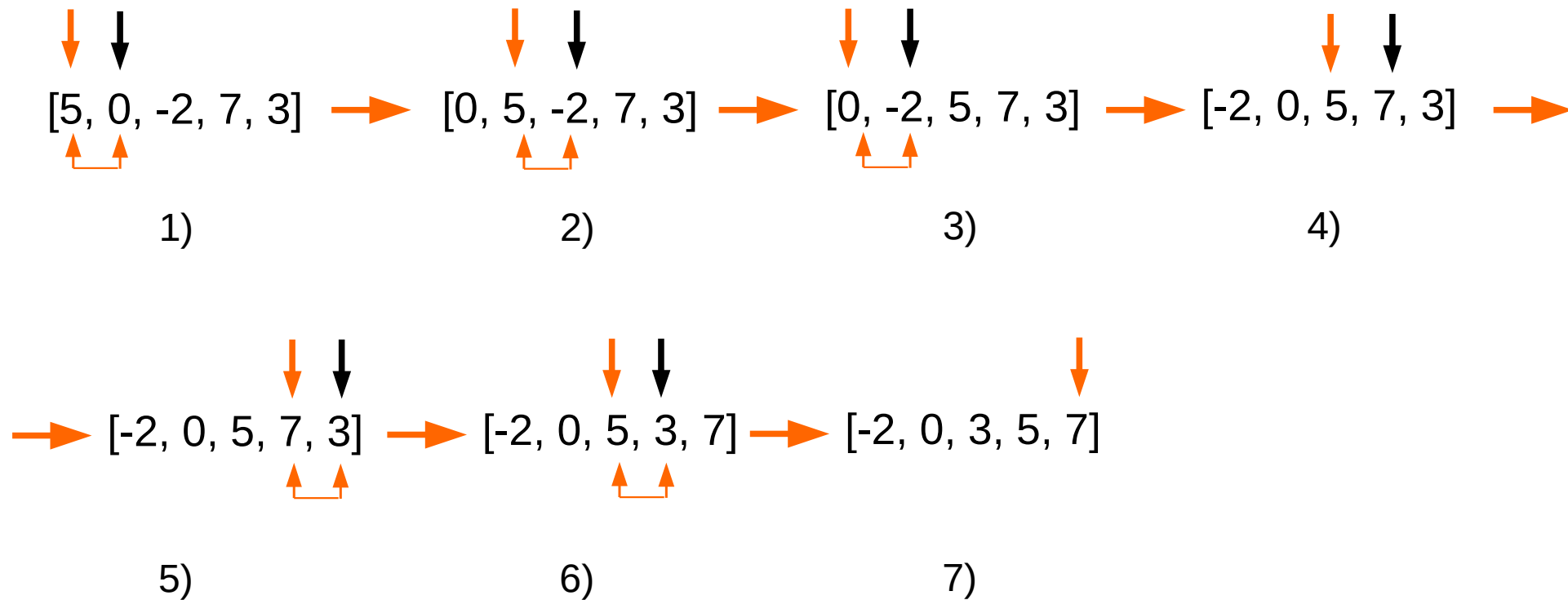


Принцип работы алгоритма

- 1) В качестве опорного берем первый элемент последовательности. Вводим дополнительную переменную, для хранения индекса. Устанавливаем ее равной индексу первого элемента.
- 2) Выполняем сравнение опорного и следующего за ним элемента. Если опорный элемент является последним элементом последовательности заканчиваем алгоритм. Если опорный элемент меньше или равен следующему элементу, то в качестве опорного выбирают следующий элемент и повторяют пункт 2.
- 3) Если опорный элемент больше следующего, то устанавливаем значение переменной для хранения индекса равным значению индекса следующего элемента. Производим обмен опорного и следующего элемента. В качестве опорного выбираем предыдущий. Повторяем до тех пор пока или не дойдем до начала последовательности или опорный элемент не станет меньше следующего. После чего в качестве опорного выбираем элемент индекс которого хранится во вспомогательной переменной. И повторяем пункт 2.

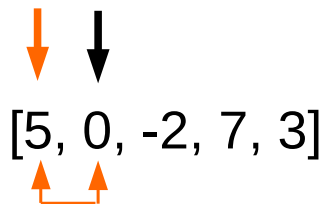


Графическая иллюстрация работы алгоритма





Графическая иллюстрация работы алгоритма

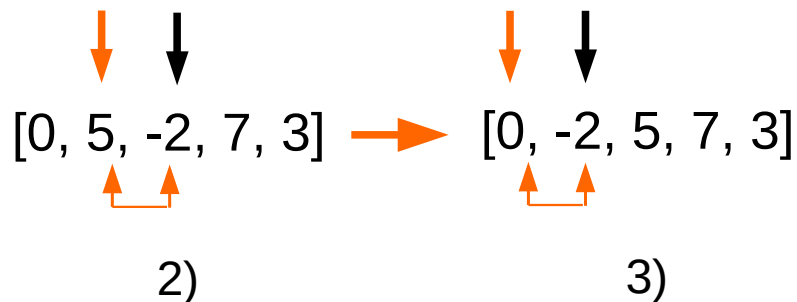


1)

На первом шаге в качестве опорного элемента выбирают первый элемент последовательности. Следующий элемент меньше опорного, поэтому производится их обмен.






Графическая иллюстрация работы алгоритма



На втором шаге опорный элемент больше следующего, поэтому устанавливаем значение переменной для хранения индекса равной индексу следующего элемента. И выполняем обмен. После этого на 3 шаге оказывается, что требуется еще один обмен, который и выполняется.



Графическая иллюстрация работы алгоритма

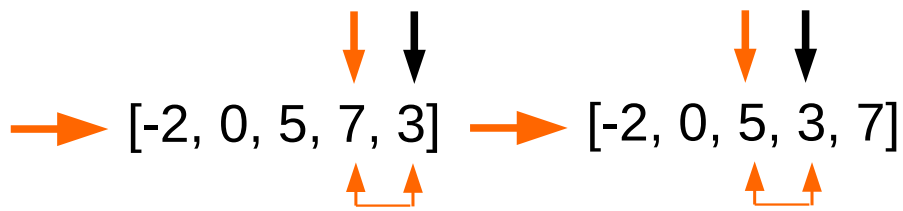
 
[-2, 0, 5, 7, 3] 

4)

В качестве опорного выбирается элемент с сохраненным индексом. Опорный меньше следующего. В качестве опорного выбираем следующий элемент.



Графическая иллюстрация работы алгоритма



5)

6)

На шаге 5, опорный больше следующего, поэтому устанавливаем значение переменной для хранения индекса равной индексу следующего элемента. И выполняем обмен. После этого на 6 шаге оказывается, что требуется еще один обмен, который и выполняется.



Графическая иллюстрация работы алгоритма



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

7)

Опорный элемент выступает в качестве последнего элемента в последовательности. Алгоритм окончен.



Реализация алгоритма на Python



Реализация алгоритма на Python

```
def gnome_sort(sequence):  
    index = 1  
    i = 0  
    n = len(sequence)  
    while i < n-1:  
        if sequence[i] <= sequence[i+1]:  
            i, index = index, index + 1  
        else:  
            sequence[i], sequence[i+1] = sequence[i+1], sequence[i]  
            i = i - 1  
            if i < 0:  
                i, index = index, index + 1
```



Java

Реализация алгоритма на Java



Реализация алгоритма на Java

```
public static void gnomeSort(int[] array) {  
    int index = 1;  
    int i = 0;  
    for (; i < array.length - 1;) {  
        if (array[i] <= array[i + 1]) {  
            i = index;  
            index += 1;  
        } else {  
            int temp = array[i];  
            array[i] = array[i + 1];  
            array[i + 1] = temp;  
            i = i - 1;  
            if (i < 0) {  
                i = index;  
                index += 1;  
            }  
        }  
    }  
}
```



Список литературы

- 1) Sarbazi-Azad, Hamid (2 October 2000). "Stupid Sort: A new sorting algorithm". Newsletter. Computing Science Department, Univ. of Glasgow (599)
- 2) <https://dickgrune.com/Programs/gnomesort.html>