

Семинар 1. Базовые алгоритмы сортировки

Сортировка вставками

Допустим, у нас есть массив чисел: [5, 2, 7, 1, 3]

1. Сначала мы берем первый элемент (5) и считаем его отсортированным подмассивом.
2. Далее берем второй элемент (2) и сравниваем его с первым элементом (5). Так как 2 меньше 5, мы меняем их местами и получаем подмассив [2, 5].
3. Теперь берем третий элемент (7) и сравниваем его с предыдущими элементами (5 и 2). Так как 7 больше 5 и 2, он остается на своем месте и мы получаем подмассив [2, 5, 7].
4. Далее берем четвертый элемент (1) и начинаем сравнивать его с предыдущими элементами (7, 5 и 2). Так как 1 меньше всех этих элементов, мы сдвигаем каждый из них на одну позицию вправо и вставляем 1 на первое место. Получаем подмассив [1, 2, 5, 7].
5. Наконец, берем последний элемент (3) и сравниваем его с предыдущими элементами (7, 5 и 2). Так как 3 меньше 7 и больше 2, мы сдвигаем только 7 на одну позицию вправо и вставляем 3 на место 5. Получаем отсортированный массив [1, 2, 3, 5, 7].

Таким образом, мы прошли по всем элементам массива и отсортировали его по возрастанию.

Сортировка выбором

Допустим, у нас есть массив чисел: [8, 3, 1, 6, 4]

1. Сначала мы ищем наименьший элемент в массиве и меняем его местами с первым элементом. Таким образом, получаем массив [1, 3, 8, 6, 4].
2. Далее ищем наименьший элемент среди оставшихся (3, 8, 6, 4) и меняем его местами со вторым элементом. Получаем массив [1, 3, 8, 6, 4].
3. Продолжаем этот процесс до тех пор, пока не пройдем по всем элементам массива. В результате получаем отсортированный массив [1, 3, 4, 6, 8].

Таким образом, мы прошли по всем элементам массива и отсортировали его по возрастанию, каждый раз находя наименьший элемент и помещая его на нужную позицию.

Сортировка обменом

Дан массив чисел: [5, 2, 8, 4, 9, 1]

Шаг 1: сравниваем первые два элемента массива (5 и 2). Так как 2 меньше 5, меняем их местами. Получаем массив [2, 5, 8, 4, 9, 1].

Шаг 2: сравниваем следующие два элемента (5 и 8). Так как они уже стоят в правильном порядке, оставляем их на месте. Массив остается без изменений.

Шаг 3: сравниваем следующие два элемента (8 и 4). Так как 4 меньше 8, меняем их местами. Получаем массив [2, 5, 4, 8, 9, 1].

Шаг 4: сравниваем следующие два элемента (8 и 9). Так как они уже стоят в правильном порядке, оставляем их на месте. Массив остается без изменений.

Шаг 5: сравниваем следующие два элемента (9 и 1). Так как 1 меньше 9, меняем их местами. Получаем массив [2, 5, 4, 8, 1, 9].

Шаг 6: продолжаем сравнивать и менять элементы до тех пор, пока массив не будет отсортирован полностью. В результате получим отсортированный массив [1, 2, 4, 5, 8, 9].

Сортировка пузырьком

1. Начать сравнение элементов массива с первого элемента до предпоследнего.
2. Если текущий элемент больше следующего, то поменять их местами.
3. Продолжать сравнение до конца массива.
4. После первого прохода, на последнем месте будет находиться наибольший элемент.
5. Повторять процесс для оставшихся элементов, исключая уже отсортированные.
6. Продолжать процесс до тех пор, пока все элементы не будут отсортированы по возрастанию.

Шейкерная сортировка

Алгоритм шейкерной сортировки (также известный как двунаправленная сортировка или коктейльная сортировка) является усовершенствованной версией пузырьковой сортировки. Он работает путем сравнения и обмена элементов попарно, двигаясь в обе стороны по массиву.

1. Начинаем проход по массиву с начала и сравниваем каждую пару соседних элементов. Если текущий элемент больше следующего, то они меняются местами. Таким образом, наибольший элемент будет перемещен в конец массива.

Пример:

Начальный массив: [5, 2, 8, 3, 1]

[2, 5, 3, 1, 8] - наибольший элемент 8 перемещен в конец массива.

2. После прохода по всему массиву начинаем проход с конца и сравниваем каждую пару соседних элементов. Если текущий элемент меньше предыдущего, то они меняются местами. Таким образом, наименьший элемент будет перемещен в начало массива.

Пример:

[2, 1, 3, 5, 8] - наименьший элемент 1 перемещен в начало массива.

3. После этого мы снова начинаем проход с начала массива и повторяем шаги 1 и 2 до тех пор, пока все элементы не будут отсортированы по возрастанию.

4. Когда при очередном проходе по массиву не происходит ни одной замены, это означает, что массив уже отсортирован и алгоритм может быть остановлен.

Отсортированный массив: [1, 2, 3, 5, 8]

Варианты

- 1) [17, 3, 0, 12, 6, 9, 19, 1]
- 2) [5, 10, 18, 0, 2, 7, 11, 14]
- 3) [20, 8, 4, 16, 1, 13, 6, 3]
- 4) [9, 19, 15, 2, 10, 7, 0, 4]
- 5) [14, 6, 12, 18, 5, 1, 11, 3]
- 6) [0, 8, 2, 17, 9, 13, 7, 16]
- 7) [3, 19, 10, 1, 15, 4, 11, 20]
- 8) [12, 6, 0, 14, 5, 18, 2, 8]
- 9) [7, 17, 9, 3, 13, 1, 16, 10]
- 10) [4, 0, 15, 6, 12, 2, 14, 8]
- 11) [16, 9, 5, 18, 1, 11, 7, 3]
- 12) [2, 10, 19, 0, 6, 13, 8, 4]

- 13) [11, 7, 14, 20, 5, 1, 15, 9]
- 14) [0, 3, 17, 8, 2, 12, 6, 19]
- 15) [13, 9, 4, 16, 1, 11, 5, 20]
- 16) [7, 15, 10, 2, 18, 0, 14, 6]
- 17) [1, 8, 19, 3, 12, 16, 9, 5]
- 18) [20, 11, 7, 4, 17, 2, 13, 0]
- 19) [6, 14, 9, 1, 15, 8, 5, 19]
- 20) [3, 10, 18, 0, 7, 12, 2, 16]
- 21) [4, 11, 5, 20, 1, 15, 8, 3]
- 22) [17, 0, 6, 12, 2, 9, 4, 19]
- 23) [8, 16, 1, 13, 7, 3, 20, 10]
- 24) [5, 14, 2, 18, 0, 6, 12, 9]
- 25) [11, 1, 15, 8, 4, 20, 10, 3]
- 26) [14, 11, 5, 8, 4, 2, 15, 3]
- 27) [4, 13, 10, 8, 1, 7, 10, 5]
- 28) [5, 2, 15, 20, 4, 8, 3, 10]
- 29) [17, 15, 10, 4, 8, 11, 13, 3]