

Сортировки

① Задача сортировки:

$\langle a_1; a_2; a_3; a_4; \dots a_n \rangle$ - набор объектов.
↓ алгоритм

$\langle a'_1; a'_2; a'_3; \dots a'_n \rangle$, где
 $a'_1 \leq a'_2 \leq a'_3 \leq \dots \leq a'_n$

② Видов "порядка"

		<u>10 ; 5 ; 7 ; -10 ; 5</u>
→	по убыванию	\leq -10 ; 5 ; 5 ; 7 ; 10
→	по возрастанию	$<$ -10 ; 5 ; 8 ; 7 ; 10
→	не по возрастанию	\geq 10 ; 7 ; 5 ; 5 ; -10
→	по убыванию	$>$ 10 ; 7 ; 5 ; 8 ; -10

③ Применение:

- архивизм
- бинар. поиск
- проектиру.

IV

Свойства

→ время выполнения

$T(N)$

TIME

кол-во
выходов

Δ APPROX

→ Требуемая память

$M(N)$

MEMORY

локальная

сортировки $\Rightarrow M(N) = O(1)$

→ Устойчивость

$a_1; a_2; a_3; a_4; \dots; a_7; \dots; a_n$

$\forall i, j$
 $i \neq j$

$a_i \neq a_j$, кроме

$a_4 = a_7$

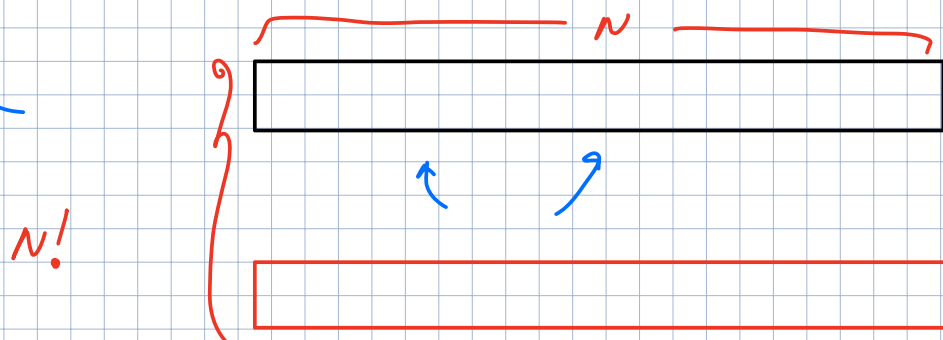
a_7, a_4 — не устойчивая.

$a_2; a_4; a_3; \dots; a_n; a_7; \dots; a_n$

(не поменялись
местами (один и
тот же элемент))

⑤ Тривиальная сортировка

перестановка

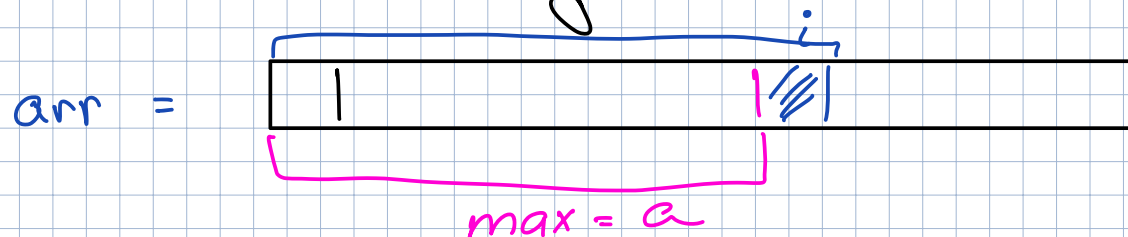


$$O(N! \cdot N)$$

$$N! \sim n^n$$

$$100 \rightarrow 100^{98}$$

⑥ Поиск максимума

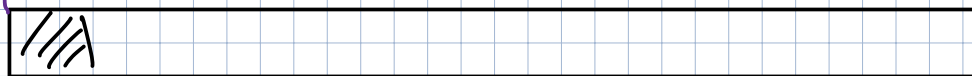


если $\max < arr[i]$:

$\max = arr[i]$

11

arr =



L

max = arr[0]

for
{

i = 1 ... N:

если arr[i] > max

↳ max = arr[i]

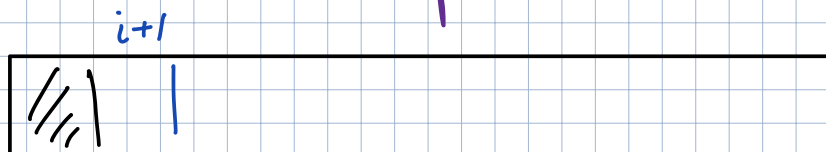
}

$T(N) =$
 $= \underline{O}(N)$

$M(N) = \underline{O}(1)$

⑦ Сортировка выбором (Selection Sort)

arr =



↑
max

max

не возрастающие
"удовольствие"

1	2	10	3
10	1	2	3

for
{

i = 0 ... N

{

for j = (i+1) ... N:

{

если arr[j] > arr[i]

↳

SWAP(arr[i], arr[j])

{

{

$$T(N) = \underline{O}(N)$$

$$M(N) = \underline{O}(1)$$

Асимптотика : $T(N) = \underline{O}(N^2)$

$$M(N) = \underline{O}(1)$$

SWAP :

$$\begin{aligned} t &= a \\ a &= b \\ b &= t \end{aligned}$$

Устойчива ли сортировка?

$$\underline{\Delta A}$$

$$a \xrightarrow{1}$$

$$b \xrightarrow{2}$$

$$a = b$$

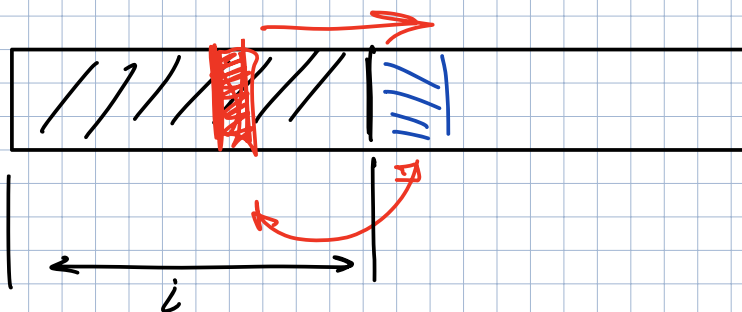
$$b = a$$

$$a - b = 2$$

VIII

Сортировка вставками

(Insertion Sort)



те возрастающего

$$\begin{array}{|c|c|} \hline j-1 & j \\ \hline \end{array}$$

```

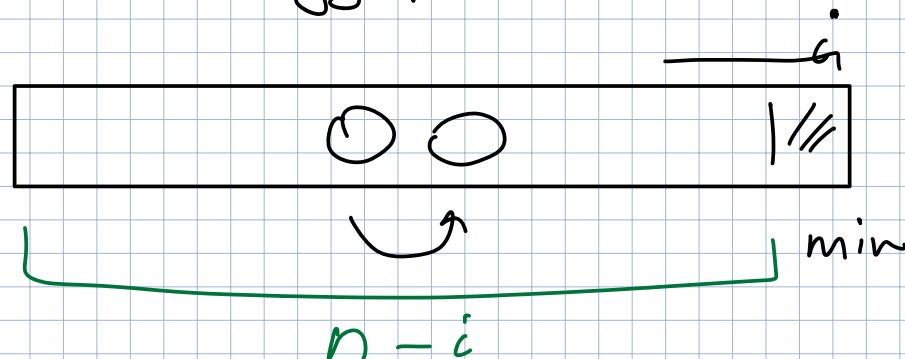
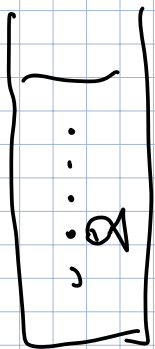
for i = 0 ... N
{
  j = i + 1
  while (j > 0) and (arr[j-1] < arr[j])
  {
    t = arr[j-1]
    arr[j-1] = arr[j]
    arr[j] = t
    j = j - 1
  }
}

```

} SWAP

$T(N) = O(N^2)$ Устойчивая
 $M(N) = O(1)$

IX) Сортировка Пузырьком (Bubble Sort)



```

for i = 0... N:
{
  for j = 0... N-i:
  {
    if arr[j] < arr[j+1]:
      t = arr[j]
      arr[j] = arr[j+1]
      arr[j+1] = t
  }
}

```

} SWAP

$$T(N) = \underline{O}(N^2)$$

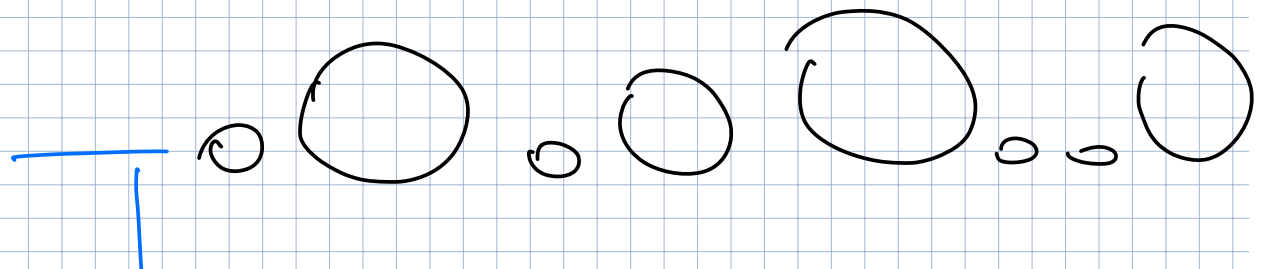
$$M(N) = \underline{O}(1)$$

Устойчивость: нет

(x)

Пример

$O(1)$



$$\underline{O(10)}$$

$$\underline{O(1)}$$

