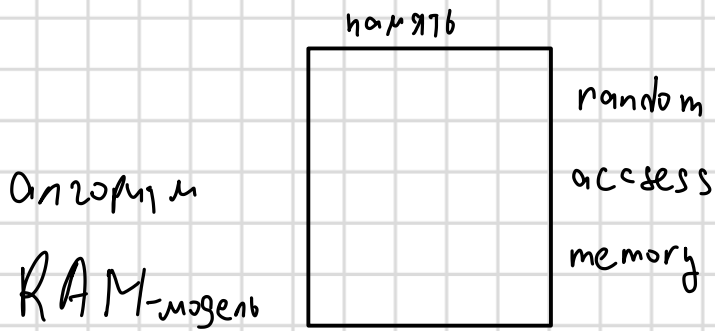


Сортировка



Сортировки:

1) С. пузырьком ^{всего $O(n^2)$}

for ($i=0, i < n, i++$)

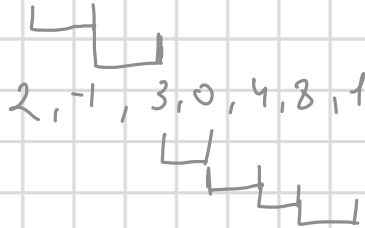
for ($j=0, j < n-i-1, j++$)

if ($a[j] > a[j+1]$): $O(n^2)$

swap($a[j], a[j+1]$)

$O(n^2)$

{ 2, 3, -1, 0, 4, 8, 1 }



т.е. ↓

{ 2, -1, 0, 3, 4, 1, 8 }

2) С. выбором ^{всего $O(n^2)$}

for ($i=0 \dots n$)

$mn = i$

for ($j=i, j < n$)

if ($a[j] < a[mn]$): $O(n^2)$

$mn = j$

swap($a[mn], a[j]$) $O(n)$

5 4 -1 3 2 4

1: -1 1 5 3 2 4

2: -

3: -1 1 2 3 5 4

4:

5: -1 1 2 3 4 5

всего $O(n^2)$

3) С. Вставками не все $O(n^2)$

for ($i = 1 \dots n-1$) $O(n+k)$ k -кон-во
универс

pos = i

while (pos > 0):

if ($a[pos] < a[pos-1]$):

swap ($a[pos], a[pos-1]$)

pos --

else

break

1: -1 1 ...

2: -1 1 3 ...

3: -1 1 3 5

4: -1 1 2 3 5

...

n: -1 -1 1 2 3 4 5

a = 3 7 5 1 4

;

sort 1 3 4 5 7

idx

4) С. Слиянием

def sort ($a[0 \dots n]$)

l = sort ($a[0 \dots \frac{n}{2}]$)

r = sort ($a[\frac{n}{2} \dots n]$)

merge (l, r)

2 1 4 8 | 7 5 6 3

$$T(n) = 2T\left(\frac{n}{2}\right) + n = O(n \log n)$$

a

a_1	a_2
-------	-------

b_1	..
-------	----

 b

$\min(a_1, b_1)$

c

a_1

merge(a, b)

$\Theta(n+m)$

$n = |a|$

$m = |b|$

$i, j = 0, 0$

while ($i \neq n$ or $j \neq m$):

if ($(j=m) \text{ or } (a_i < b_j)$):

c.add(a_i)

$i++$

else:

c.add(b_j)

$j++$

return c



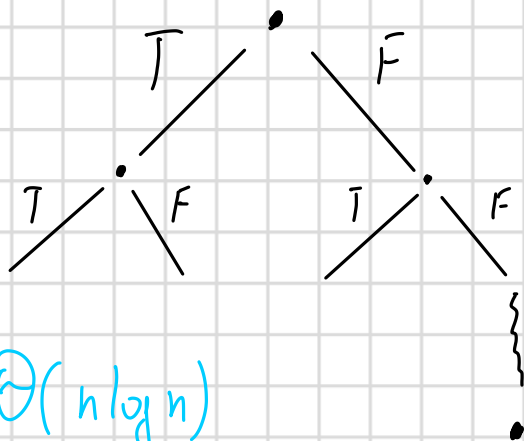
$n-1$

Анализ

$a_i \geq a_j$

$\Omega(n \log n)$

Мерге Мер



$K = \Theta(n \log n)$

K-мук. Кн-то онеpayи

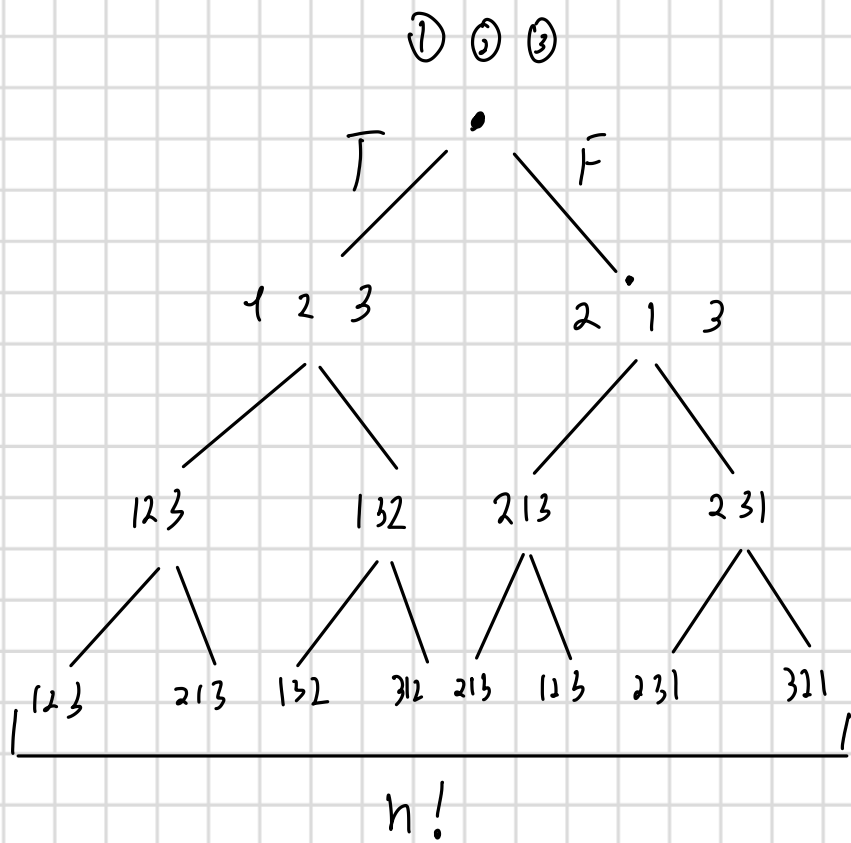
нзгирек:

0 0 0

$I < II$

$II < III$

$I \leq II$

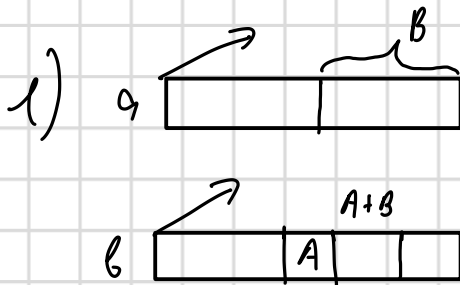


$$\log(n!) = K$$

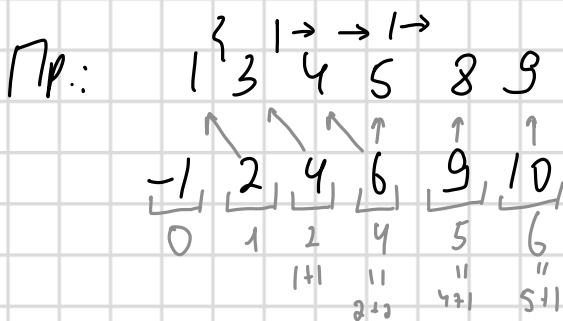
$$2^{K-1} \leq n! \leq 2^K$$

Визуализация всех вариантов перестановки
 71-ов в массиве иная ил
 Это - то что нужно

Метод двух указателей



$$\forall b_i \neq a_j < b_i$$



pred_ans = 0

$O(n+m)$

for (i = 0 ... n)

while (pred_ans < |a| && a[pred_ans] < b[i])

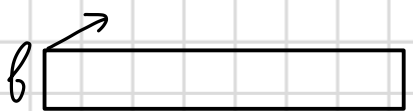
pred_ans ++

ans[i] = pred_ans

2)



$$a_i + b_j = S$$



j = m - 1

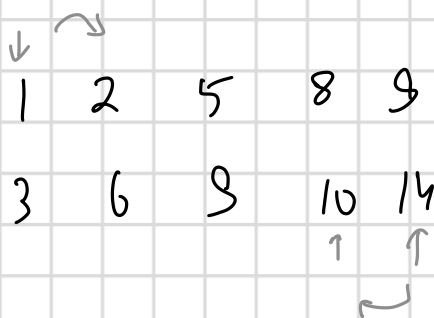
$O(n+m)$

for (i = 0 ... n)

while (a_i + b_j > S)

j -- 1

if (a_i + b_j = S) ok



14