

БИНАРНАЯ КУЧА

Пусть $A = a_1, a_2, \dots, a_n$ – массив.

Массив A называется **бинарной кучей**, если

$$(\forall i \mid 1 \leq i \wedge 2 \cdot i \leq n)(a_i \geq a_{2 \cdot i}) \wedge (\forall i \mid 1 \leq i \wedge 2 \cdot i + 1 \leq n)(a_i \geq a_{2 \cdot i + 1})$$

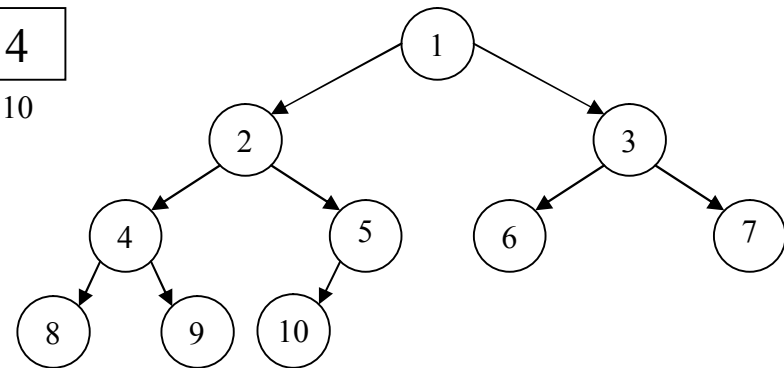
Левый и правый сыновья:

$$left_son(i) = \begin{cases} 2 \cdot i, 2 \cdot i \leq n \\ undefined, 2 \cdot i > n \end{cases}$$

$$right_son(i) = \begin{cases} 2 \cdot i + 1, 2 \cdot i + 1 \leq n \\ undefined, 2 \cdot i + 1 > n \end{cases}$$

A:

16	11	9	10	5	6	8	1	2	4
1	2	3	4	5	6	7	8	9	10



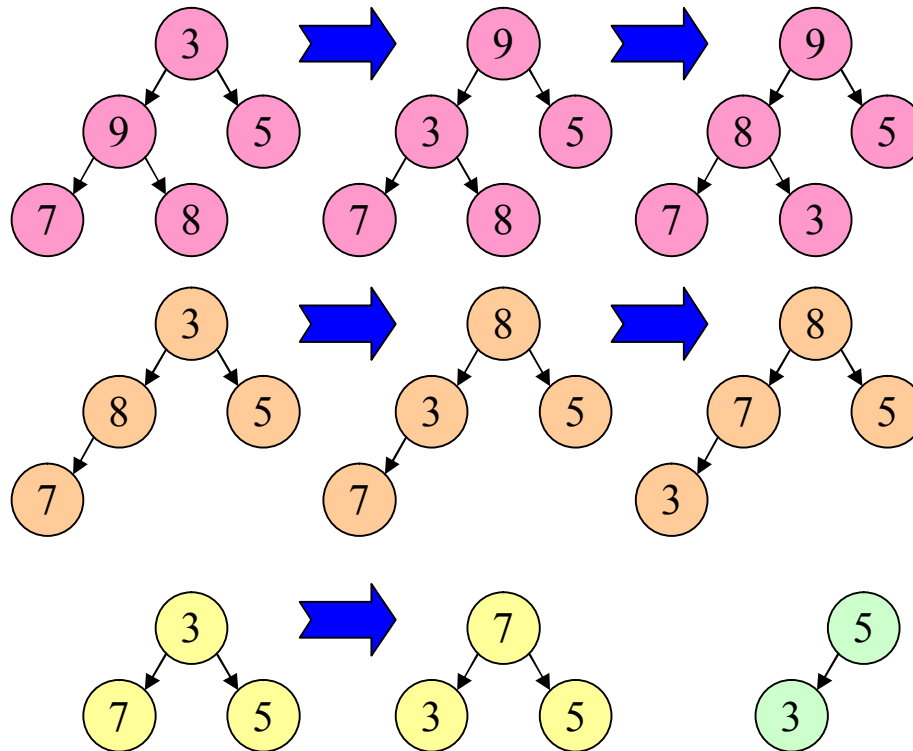
ПЕРЕСЫПКА (SHAKING)

```
shaking:= proc (  
    A, # исходный массив  
    n, # размер исходного массива  
    k # начало отрезка [k .. n]  
)  
    if 2*k <= n then  
        if 2*k + 1 <= n then  
            if A[k] < max(A[2*k], A[2*k + 1]) then  
                if A[2*k] < A[2*k + 1] then  
                    swp(k, 2*k + 1); shaking(A, n, 2*k + 1)  
                else  
                    swp(k, 2*k); shaking(A, n, 2*k)  
                end if;  
            end if;  
        else  
            if A[k] < A[2*k] then  
                swp(k, 2*k); shaking(A, n, 2*k)  
            end if;  
        end if;  
    end if;  
    return  
end proc
```

ПОСТРОЕНИЕ БИНАРНОЙ КУЧИ (BINARY_HEAP_CREATE)

```
binary_heap_create := proc (  
  A, # исходный массив  
  n # размер исходного массива  
)  
  for k from [n/2] by -1 to 1 do  
    shaking(A, n, k)  
  end do  
end proc
```

3	9	5	7	8
9	8	5	7	3
3	8	5	7	9
8	7	5	3	9
3	7	5	8	9
7	3	5	8	9
5	3	7	8	9
3	5	7	8	9



СОРТИРОВКА БИНАРНОЙ КУЧЕЙ (BINARY_HEAP_SORT)

```
binary_heap_sort := proc (  
  A, # исходный массив  
  n  # размер исходного массива  
)  
  binary_heap_create(A, n);  
  for k from n by -1 to 2 do  
    swp(A[1], A[k]);  
    shaking(A, 1, k - 1)  
  end do  
end proc
```

Теорема. Пусть $A = a_1, a_2, \dots, a_n$ – массив, тогда алгоритм `binary_heap_sort` сортирует его за время $\Theta(n \log n)$ в худшем случае.