9631407 - علی فرجی پور

تمرین سری چهارم

(1)

```
Nonrecursive_OS_Select (x, i)
r = x.left.size
while i ≠ r do
    if i > r then
        x = x.right
        i = i - r
    else
        x = x.left
    r = x.left.size
return x
```

---

(2)

```
Interval (T, i)
if overlaps i then
    Print T.root

if T.root.left ≠ T.null and T.root.left.max >= i.low then
    Interval (T.root.left, i)
if T.root.right ≠ T.null and T.root.right.max >= i.high then
    Interval (T.root.right, i)
```

---

(3) فرض کنید که heap یک BST کامل با $n=2^k-1$ باشد؛ پس 2^{k-1} برگ و $2^{k-1}-1$ گره غیربرگ (داخلی) داریم. 2^{k-1} المان اول heap را در نظر بگیرید، فرض کنید برگ ها سیاه و غیربرگ ها سفید باشند. گره های رنگی یک زیردرخت از heap هستند. از آن جایی که 2^{k-1} گره رنگی وجود دارد و ماکسیمم 2^{k-2} تا سیاه هستند. پس $2^{k-2}-1$ گره سفیدند.

گره های سیاه می توانند به عنوان گره در نظر گرفته شوند ولی برای گره های سفید باید swap انجام شود. مینیمم تعداد swap در حالتی است که 1- $2^{k-2}-1$ گره سفید باشد، 2- به صورت BST مرتب شده باشند. اگر w تا گره سفید داشته باشیم، $\lg w$ تا سطح داریم که هر کدام $2^{\lg w}$ گره دارند؛ پس تعداد swap برابر است با

$$\sum_{i=0}^{\lg w} i2^i = 2+(\lg w-2)2^{\lg w} = \Omega(w \lg w)$$

، می دانیم $T(n)=T(n/2)+\Omega(n\lg n)$

$$\Rightarrow T(n) = \Omega(n\lg n)$$

(۴) یک آرایه با سایز |V| .. ۰ در نظر بگیرید. برای یک لیست در لیست همسایگی برای راس v ، تمام المان ها را بررسی می کنیم ؛ اگر هر کدام با u برابر بود ، آن را حذف می کنیم. به ازای راس u ، A[u] را بررسی می کنیم ، اگر با v برابر نبود آن را برابر v قرار می دهیم و اگر برابر بود u را حذف می کنیم . از آن جا که زمان دسترسی Constant است (در آرایه) ، زمان الگوریتم برابر $O(V+E)$ می شود .

---

(5) زمان iterate روی یال ها از $O(E)$ به $O(V^2)$ تبدیل می شود؛ پس :

$$= O(V+V^2) = O(V^2)$$

DFS_Stack(G)


```
for every v ∈ G.V do
    v.color = WHITE
    v.parent = Null
time = 0
S = empty stack
while there is a white vertex v in G do
    S.push(v)
    while S is not empty do
        v = S.pop
        time++
        v.d = time
        for all neighbors w of v do
            if w.color == WHITE
                w.color = BLACK
                w.parent = v
                S.push(w)
        time++
        v.f = time
```

(7)

d-1

s	t	u	y	z
0	3	∞	5	∞
0	3	9	5	∞
0	3	9	5	11
0	3	9	5	11
0	3	9	5	11

d-2

s	t	u	y	z
3	∞	7	∞	0
3	6	7	8	0
3	6	7	8	0
3	6	7	8	0
3	6	7	8	0

π-1

s	t	u	y	z
Null	s	Null	Null	Null
Null	s	t	s	Null
Null	s	t	s	y
Null	s	t	s	y
Null	s	t	s	y

π-2

s	t	u	y	z
z	Null	z	Null	Null
z	s	z	s	Null
z	s	z	s	Null
z	s	z	s	Null
z	s	z	s	Null