



2.

$\text{OS-SELECT}(x, i)$

1. $n = x.\text{left_size} + 1$
2. if $n = i$:
3. return x
4. else if $i < n$:
5. return $\text{OS-SELECT}(x.\text{left}, i)$
6. else:
7. return $\text{OS-SELECT}(x.\text{right}, i - n)$

Träume $\text{OS-SELECT}(T.\text{root}, 8)$.

OS-SELECT kann position n über i return
möglichkeit klausur

6. From bottom $i=8$, $n=6$. (So far we already have 6
keyed like odours known OS-SELECT (T.next right, 2)).

Number of no. items $56 + 1 = 2$. Total count in count 3
so far known OS-SELECT (96. left, 2). So far we have known

no. items $37 + 2$. Avg 37 more items object per
count $n=7$, $i=2$, $n=2$, total known so far known
OS-SELECT (37-right, 1). So far we have known no. items 48.
On average items object per item $n=7$, $i=1$ to check
the unknowns. So after no. items 1 known 48.

3. OS-RANK(T, X)

1. $n = X.bft. max + 1$

2. $\gamma = X$

3. while $\gamma \neq T.\text{root}$

4. if $\gamma = \gamma.p.\text{right}$

5. $n = n + \gamma.p.bft. max + 1$

6. $\gamma = \gamma.p$

7. return n .

OS-RANK^k means how many nodes known X a known path by selected
nodes selection.

From OS-RANK(T, X), $X.bft = G.Y$.

6. From bottom $n=1$ to 53 more items object $\gamma=99$. Union
a white path. γ my items object being path per known items
is $9=61$. γ to another item object per $n=6$ objects my
items object using $n=6$ $\Rightarrow n=1+2+1=4$. Known items $\gamma=56$.

On γ there are items object $n=4+5+1=10$. So far known items

3. OSKEY RANK (Γ_k)

$y = \text{SEARCH}(k)$

if $x = \text{nullptr}$

return - 1

N = N, left, right

47

while you're not:

if $y = 4$. P. right

$n = \text{C-17, P. Alt., Mx + 1}$

T = 4. p

return 10

for i in range(0, len(keys) - 1):
 if keys[i] > keys[i + 1]:
 return 10
 else:
 continue

if len(keys) == 1:
 return 10
else:
 for i in range(0, len(keys) - 1):
 if keys[i] > keys[i + 1]:
 return 10
 else:
 continue

OS_RANK_REC(i, k)

vector<int> keys = NODER(i)

for first in range(0, len(keys) - 1):

if keys[first] > keys[first + 1]:

return first + 1

else:

return -1

NODER(i) \neq i hyo next vector n't right u mrode
Portuguese, machine hyo won't know go forward u vector
i myper molers +1 slywsm work own 3 hndwn
k.

```
i-thSuccessor(T, x, i)
y = OS-RANK(T, x)
RETURN OS-SELECT(T, next, y+i)
```

Upward rank node x, i-th rank object in tree
of global rank 1-to ∞ algorithm: OS-SELECT procedure
i-th ranking node column i-th algorithm: OS-SELECT
OS-RANK node \cup O(km) \Rightarrow $O(km)$ time complexity \Rightarrow
 $O(km)$.

2. i^{th} Successor(T, x, i)

```
y = x
for (int j=0; j < i; j++)
    y = PROG-SUCCESSOR(T, x)
return y
```

TREE-SUCCESSOR node \cup $O(k+h)$ uprow, to new parent
node, $h = \text{lym} \Rightarrow O(k+lym)$. If x_i present by finding
a copy in binary g_i else move previous column left to
copy into resultant $O(m+lym)$