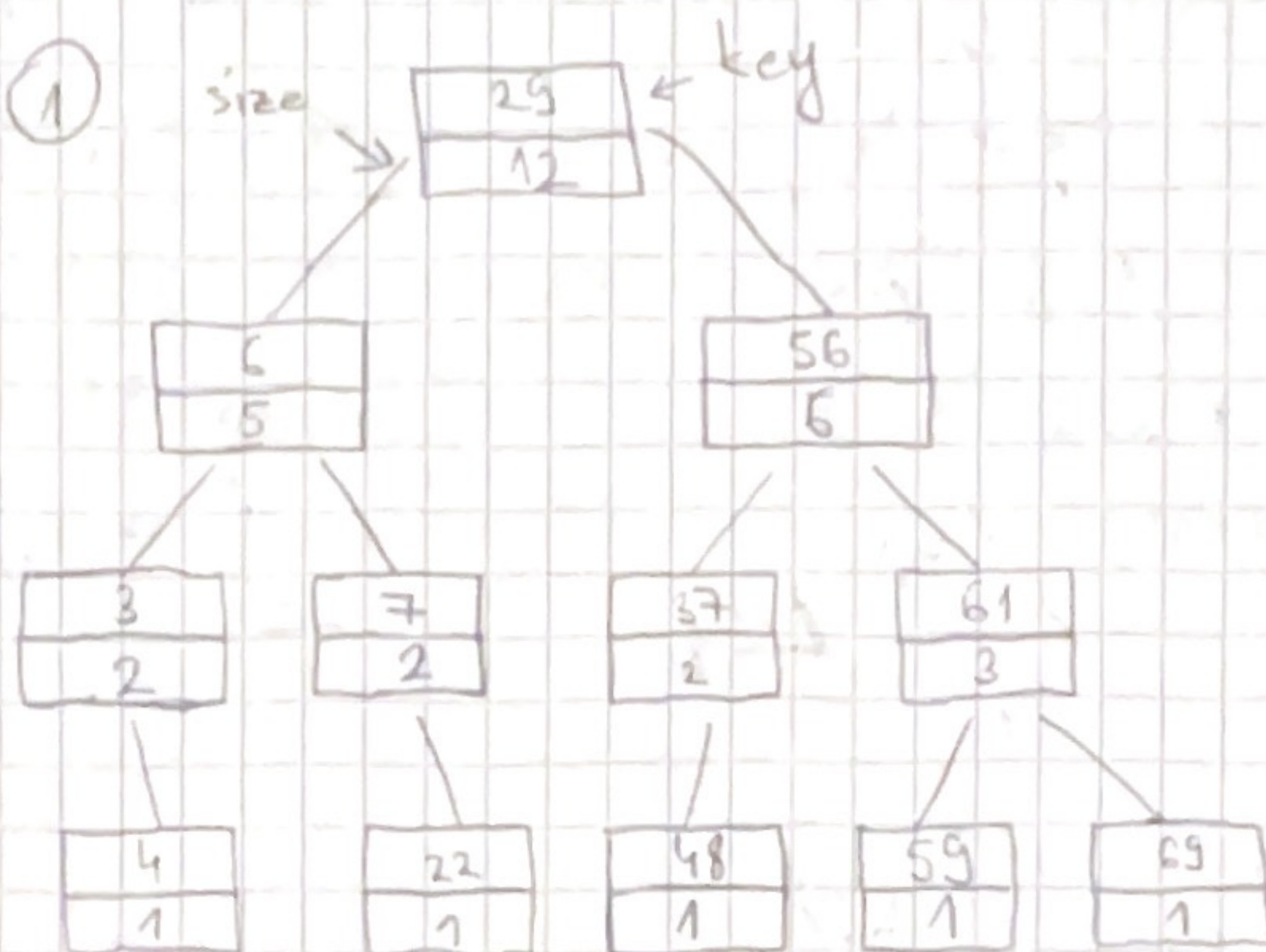


ZADACA 6 - SPA 2

Zad 1.

①



② Procedura OS-SELECT ($T.root, 48$) pronalazi čvor s ključem 48 u stablu. Početni poziv OS-SELECT ($T.root, 48$) provjerava rang lijevog podstabla korijena (tj. 6, 3, 7, 4 i 22), 48 je veći od tih 5 čvorova, pa se f-ja rekurzivno poziva na desno podstablo, tj. OS-SELECT ($T.root, right, 48-6$), gdje je 6 broj čvorova u desnom podstablu. Tada se f-ja poziva na lijevo podstablo čvora 56 i traži se 42. elem. u tom podstablu s obzirom da je rang lijevog podstabla čvora 56 jednak 2 (37 i 48 su u tom lijevom podstablu), je li 42 veći od 2, f-ja se poziva na desno podstablo čvora 56, tj. OS-SELECT ($T.root, right, 42-2$), a gdje je 2 broj čvorova u lijevom podstablu čvora 56. Tada dođe do čvora s ključem 48, f-ja se završava i vraća taj čvor.

- ③ Procedure OS-RANK (T, x) pronalazi rang čvora ' x ' u stablu s korjenom ' T '. Za čvor s bližem 59, f -ju se pravi korak prema korjenu, računajući rang svakog čvora u odnosu na njegovog roditelja. Kada f -ju dođe do čvora s bližem 59, računanje se prekida i vraća se rang tog čvora.

Zad 2. OS-SELECT (T, i):

```
1.  $x = T.root$ 
2. while  $x \neq T.nil$  and  $i \neq x.left.size + 1$ :
3.   if  $i < x.left.size + 1$ 
4.      $x = x.left$ 
5.   else:
6.      $i = i - x.left.size - 1$ 
7.      $x = x.right$ 
8. return  $x$ 
```

Funkcija prima stablo T i broj koji predstavlja redni broj traženog elem. u poretku bližem u stablu. Preduhvat računanje postavljanjem pokazivača x na korijen stabla T , a varijabla i se postavlja na 1 što je redni broj korijena. U svakoj iteraciji se upoređuje i s trenutnim rednim brojem i lijevim podstablom čvora x , što je i manji od veličine lijevog podstabla $x + 1$, petlja se nastavlja u lijevom podstablu od x , što je i veći, smanjuje se ova veličina lijevog podstabla $x + 1$ i petlja se nastavlja u desnom podstablu od x , što se i podudara s veličinom lijevog podstabla $+ 1$, pronalazi se traženi čvor i petlja se prekida, što čvor nije pronalazi do broja petlje, vraća se na korijen elem.

ne postoji u stablu. VSA je $O(\lg n)$, n je broj čvorova u stablu.

Zad 3. OS-KEY-RANK(T, k)

```
1. rank = 1
2. x = T.root
3. while x != null
4.   if k == x.key
5.     rank = rank + x.left.size
6.     break
7.   else if
8.     k < x.key
9.     x = x.left
10.  else
11.    rank = rank + x.left.size + 1
12.    x = x.right
13. return rank
```

VSA : $O(\lg n)$

Zad 4. (1) Koristi more operacije nad augmentiranim AVL-om :

SELECT ili RANK

i thSuccessor(T, x, i):

```
1. if x = NIL:
2.   return NIL
3. r = RANK(T, x.key)
4. if r + 1 == i:
5.   return x.right.minimum()
6. else if r + 1 < i:
7.   return i-thSuccessor(T, x.right, i)
8. else return i-thSuccessor(T, x.parent, i)
```

VSA : $O(\lg n)$

② Konstantne operacije:

i th Successor (x, i)

1. $count = 0$
2. $node = treeSuccessor(x)$
3. while $node \neq nullptr$
4. $count += 1$
5. if $count == i$
6. return $node$
7. $node = treeSuccessor$
8. return $nullptr$

VSA: $O(n \lg n)$