**Kaj sem se na vajah naučil:**

Na vajah sem se naučil rekurzivno pregledovati dvojisko drevo. Menim, da je tukaj izjemno pomembno znanje rekurzije. Zato menim, da bi zdaj veliko lažje delal rekurzivne stvari, ker sem jih tako povadil z drevesi. Pri vsaki nalogi sem uporabil rekurzijo. Opazil sem, da se implementacije dvojiskih dreves zelo razlikujejo in tukaj smo imeli ze na vajah tezavo, saj testni primeri niso delala takrat ko smo uporabili dolocene implementacije.

**Prevzete naloge:**

V neki mafijski organizaciji so člani urejeni hierarhično. Vsakdo razen botra (vrhovnega šefa) ima natanko enega nadrejenega. Vsak mafijec ima lahko pod seboj največ dva podrejena (levega in desnega). Primer takšne mafijske organizacije:

mafija = Drevo('Salvatore', 320,

levo=Drevo('Bernardo', 200,

levo=Drevo('Matteo', 50),

desno=Drevo('Carlo', 100,

levo=Drevo('Rosalia', 70),

desno=Drevo('Tommaso', 50))),

desno=Drevo('Francesco', 120,

levo=Drevo('Giuseppe', 70),

desno=Drevo('Antonio', 60)))

Mafijci morajo zbirati denar s kriminalnimi dejavnostmi. Tisti, ki imajo podrejene, pa ga poleg tega poberejo še od svojih podrejenih. Ves “prisluženi” in prejeti denar morajo oddati svojemu nadrejenemu.

Boter je posumil, da nekateri člani goljufajo. Nekaj denarja, ki ga poberejo od podrejenih, zadržijo zase. Od vsakega člana je pridobil podatek o tem, koliko denarja je oddal naprej. Podatke je shranil v podatkovno strukturo Drevo, ki [je že implementirana](http://www.fmf.uni-lj.si/~basic/rac1/mafija.py).

1. **podnaloga**

Botra zanima, koliko denarja zaradi goljufov “ponikne”. Želi, da sestavite metodo koliko\_ponikne(**self**), ki vrne skupno vsoto denarja, ki ponikne. Primer (če mafija ustreza zgornjemu drevesu):

>>> mafija.koliko\_ponikne()

30

**Ideja rešitve:**

Pri metodi, je ideja da v vsakem koraku, ko stojimo v neki točki v dvojiškem drevesu pogledamo, ali je oseba na kateri smo trenutno zbrala več denarja kakor oba podrejena, ki sta v korenu v levem in desnem poddrevesu.

Ker metoda deluje rekurzivno najprej navedemo pogojni stavek, ki pove kaj se zgodi, če smo v postavljeni v prazno drevo. V našem primeru bomo tako vrnili enostavno 0.

Zatem metoda preveri koliko denarja je zbral podrejeni v levem poddrevesu in koliko denarja je izbral desni v desnem poddrevesu. Ker imamo lahko tudi primer v katerem oseba nima levega ali desnega podrejenega to zmeraj preverimo in če ga nima si lahko mislimo kot, da bi imel nekega podrejenega, ki ni zbral nič denarja.

Ko imamo podatke o izbranem denarju od podrejenih lahko preverimo, če je trenutna oseba nabrala manj denarja in v tem primeru vemo, da je oseba odtujila neko količino. Koliko ugotovimo tako, da od denarja trenutne osebe odšejemo obe vrednosti denarja njegovih podrejenih.

Metoda se nato rekurzivno sprehodi po vseh poddrevesih in tako prištevamo ves odtujen denar.

1. **Podnaloga**

Ko je boter dognal, koliko denarja ponikne, je totalno po████. Pri priči hoče imeti imena vseh goljufov! Napišite metodo goljufi(**self**), ki vrne množico goljufov. Vsak goljuf naj bo predstavljen z naborom. Prva kompotenta naj bo ime goljufa, druga komponenta pa količina denarja, ki ga je utajil. Primer:

>>> mafija.goljufi()

{(’Carlo’, 20), (’Francesco’, 10)}

**Ideja rešitve:**

Pri metodi, je ideja zelo podobna kot v prejšni podnalogi. V vsakem koraku stojimo v neki točki v drevesu in pogledamo, ali je oseba na kateri smo trenutno zbrala več denarja kakor oba podrejena, ki sta v korenu v levem in desnem poddrevesu. V primeru, da sta podrejena zbrala več denarja pomeni, da je oseba v korenu odtujila denar, zato jo dodamo v množico.

Ker metoda deluje rekurzivno najprej navedemo pogojni stavek, ki pove kaj se zgodi, če smo postavljeni v prazno drevo. V našem primeru, ker bomo imelo pravka z množicami, bomo vrnili prazno množico.

Zatem metoda preveri koliko denarja je zbral podrejeni v levem poddrevesu in koliko denarja je izbral desni v desnem poddrevesu. Ker imamo lahko tudi primer v katerem oseba nima levega ali desnega podrejenega to zmeraj preverimo in če ga nima si lahko mislimo kot, da bi imel nekega podrejenega, ki ni zbral nič denarja.

Ko imamo podatke o izbranem denarju od podrejenih lahko preverimo, če je trenutna oseba nabrala manj denarja in v tem primeru vemo, da je oseba odtujila neko količino. Koliko ugotovimo tako, da od denarja trenutne osebe odšejemo obe vrednosti denarja njegovih podrejenih.

Sedaj imamo vse potrebne podatke in jih zato vstavimo v množico kot par in sicer (ime, količina ukradenega denarja).

Nato metoda rekurzivno deluje v globino drevesa in tako vrača množice vseh kradljivcev. Te množice združimo z metodo unija.

**Povezave do objav v forumu:**

[Besedilo 1. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25521)

[Testni primeri 1. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25527)

[Ideja rešitve 1. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25529)

[Koda 1. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25533)

[Opomba 1. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25535)

[Besedilo 2. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25588)

[Testni primeri 2. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25589)

[Ideja rešitve 2. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25590)

[Koda 2. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25591)

[Opomba 2. podnaloge](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14166#p25592)

[Ideja za izboljšavo dane kode kolegice](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14169#p25593)

[Opozorilo kolegu, da rešitev ni prava](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14163#p25594)

[Ideja za skrajšanje kode kolega](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14162#p25595)

[Predlog za preprečevanje kopiranja implementacij v isto datoteko (dedovanje)](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14160#p25596)

[Predlog obrnitve if stavka, da se izogne pass ukazu](https://ucilnica.fmf.uni-lj.si/mod/forum/discuss.php?d=14148#p25597)