

resitev

January 28, 2024

0.1 Izboljšave

Oddelek za gospodarstvo in (motorni) promet MOL ve povedati, da so kolesarske poti v Ljubljani višek udobja in varnosti ter da jih uporabljajo tudi njihovi uslužbenci. V želji dokazati, da je res tako, so v svojih vrstah našli nekaj takšnih, ki znajo voziti kolo in so se pripravljene popeljati po Ljubljani. Žal pa nimajo vseh potrebnih veščin, zato bo morda potrebno (začasno) odstraniti kako oviro ali popraviti kako stezo.

Napisati bo potrebno dve funkciji.

- Argumenti obeh so enaki. Prvi argument je zemljevid, katerega ključi so povezave, vrednosti pa množice veščin, ki jih povezava zahteva (podobno kot dvosmerni zemljevid v prejšnji analogi). Drugi argument je množica veščin, ki jih obvlada ne uslužbenec MOL.
- Obe morata **spreminjati množice v prejetem slovarju**.
- Funkciji ne smeta vračati nobenega rezultata.

Napiši funkcijo `poenostavi(zemljevid, vescine)`, ki spremeni zemljevid tako, da ga bo lahko prevozil uslužbenec, ki obvlada le podane veščine. Z drugimi besedami, funkcija mora spremeniti množice tako, da bodo vsebovale le še veščine, ki jih najdemo med veščinami v množici, podani z drugim argumentom.

Če imamo

```
zemljevid = {
    ('A', 'B'): {'gravel', 'trava'},
    ('A', 'V'): {'pešci', 'lonci'},
    ('B', 'C'): {'bolt', 'lonci'},
    ('B', 'V'): set(),
    ('C', 'R'): {'pešci', 'stopnice', 'lonci'}}
```

ga klic `poenostavi(zemljevid, {"gravel", "lonci", "pešci"})` spremeni v

```
{
    ('A', 'B'): {'gravel'},
    ('A', 'V'): {'pešci', 'lonci'},
    ('B', 'C'): {'lonci'},
    ('B', 'V'): set(),
    ('C', 'R'): {'pešci', 'lonci'}}
```

pri čemer so množice **iste kot prej**, le drugačne.

Napiši funkcijo `odstraniti(zemljevid, vescine)`, ki spremeni množice tako, da vsebujejo veščine, ki jih je potrebno "odstraniti". Če neka množica vsebuje bolt, pešce in lonce, uslužbenec

pa obvlada le lonce in stopnice, mora množica v slovarju po klicu funkcije vsebovati bolt in pešce, torej stvari, ki jih na povezavi ne sme biti, da jo bo uslužbenec lahko prevozil.

Če bi imeli takšen zemljevid kot prej, ga klic `odstraniti(zemljevid, {"gravel", "lonci", "pešci"})` spremeni v

```
{
    ('A', 'B'): {'trava'},
    ('A', 'V'): set(),
    ('B', 'C'): {'bolt'},
    ('B', 'V'): set(),
    ('C', 'R'): {'stopnice'}}
```

0.2 Rešitev

Najpreprosteje je tako.

```
[1]: def poenostavi(zemljevid, vescine):
      for val in zemljevid.values():
          val &= vescine

      def odstraniti(zemljevid, vescine):
          for val in zemljevid.values():
              val -= vescine
```

Pomembno je, da spreminjamo množico vrednosti, torej `val &= vescine`, ne pa, recimo `val = val & vescine`, saj to naredi novo množico in ji priredi ime, ki se je prej nanašalo na staro množico. Množica, ki jo vsebuje slovar, pa se pri tem ne spremeni.

Nalogo je skoraj težko rešiti narobe. Deluje tudi, na primer,

```
for odsek in zemljevid:
    zemljevid[odsek] &= vescine
```