

Železniške povezave so podane v obliki slovarja. Ključni so imena postaj (npr. Zidani most), pripadajoče vrednosti pa sezname povezav na neposredno povezane postaje. Povezava je trojka z imenom ciljne postaje (npr. Laško), časom odhoda (v minutah od polnoči) s postaje v ključu (Zidani most) in časom prihoda na ciljno postajo (npr. Laško). Če ključu "Zidani Most" pripada seznam [('Laško', 364, 382), ('Hrastnik', 365, 386), ('Laško', 375, 390), ('Radeče', 379, 395)], iz Zidanega Mostu odhaja vlak v Laško ob 6:04 in prispe tja ob 6:22, nato v Hrastnik ob 6:05 s prihodom ob 6:26, potem spet v Laško ob 6:15 s prihodom ob 6:30, potem v Radeče ob 6:19 s prihodom ob 6:35. **Povezave so urejene po časih odhodov.** Časi so izmišljeni in vsaka povezava z resničnimi prihodi in odhodi vlakov je zgolj naključna. V tem smislu so podobni dejanskim voznim redom Slovenskih železnic.

1. Postaje

Napišite funkcijo `postaje(povezave)`, ki prejme slovar s povezavami in vrne tri sezname: seznam križišč, seznam prehodnih postaj in seznam *končnih*. Postaja je križišče, če z nje vozijo vlaki na tri ali več različnih postaj. Postaja je prehodna, če je povezana z dvema različnima postajama, in *končna*, če vlaki z nje vozijo na eno samo postajo. Vse povezave so dvosmerne: če vozi vlak iz A v B, vozi tudi iz B v A.

2. Prihod

Napišite funkcijo `naslednja_povezava(povezave, odkod, kam, cas)`, ki poišče prvi vlak, ki odide iz odkod v kam ob podanem času `cas` ali po njem. Funkcije naj vrne čas, ko bo ta vlak prispel v kam. Če ob tem času (oziroma kasneje) ni več nobenega vlaka iz odkod v kam, ali pa takšne povezave sploh ni, funkcije vrne `None`. Klic `naslednja_povezava(povezave, "Zidani Most", "Laško", 362)` vrne 390.

Napišite funkcijo `prihod(povezave, pot, zacetek)`, ki vrne čas, ob katerem bo potnik, ki prepotuje podano pot (seznam imen postaj), prispel na končno postajo `poti`, če se nanjo odpravi ob času `zacetek`. Potnik na vsaki postaji uporabi prvi možni vlak do naslednje postaje. Če poti ni možno opraviti, ker določena povezava ne obstaja ali pa na tisti dan ni več vlaka na tej povezavi, funkcije vrne `None`. Predpostavite, da vlaki Slovenskih železnic nimajo zamud.

Klic `prihod(povezave, ["Kranj", "Škofja Loka", "Ljubljana", "Litija", "Hrastnik"], 420)` vrne 585. Potnik ob 439 sede na vlak Kranj – Škofja Loka, kamor prispe ob 464 in takoj nadaljuje v Ljubljano s prihodom ob 480. Do 542 čaka na vlak, ki ga ob 565 pripelje v Litijo, takoj nadaljuje v Hrastnik in prispe ob 585.

3. Vozni redi

Napišite funkcijo `vozni_red(povezave, postaja, ime_datoteke)`, ki v datoteko s podanim imenom izpiše odhode vlakov s podane postaje. Datoteka mora biti oblikovana, kot kaže okvirček: imenu ciljne postaje sledijo odhodi z nje, nato prazna vrstica in naslednja ciljna postaja. Izpis kaže del odhodov s postaje Grosuplje.

Ivančna Gorica
06:43
07:39
08:44
09:41
10:31
Ribnica
05:58
07:00
08:03
08:56
Škofljica
07:43
08:29
09:42
10:42

4. Rentabilnost

Naslednje funkcije bodo prejele dve numpyjevi tabeli: `postaje` je tabela z imeni postaj, `potniki[i, j]` pa vsebuje število potnikov, ki se je na določen dan prepeljalo s postaje `[i]` na postaje `[j]`. Napišite naslednje funkcije.

- `naj_odhodov(postaje, potniki)` vrne ime postaje, s katere se je odpeljalo največ potnikov.
- `naj_prihodov(postaje, potniki)` vrne ime postaje, na katero je prispelo največ potnikov.
- `prometnost(postaje, potniki, n)` vrne tabelo z imeni `n` postaj z največjim skupnim številom prihodov in odhodov. Tabela naj bo urejena padajoče; najbolj prometna postaja naj bo na prvem mestu.

5. Vozovnice

Ko boste prvič pognali teste, se bo pojavil direktorij z imenom "vozovnice", v katerem bodo shranjeni podatki o vozovnicah nekaj potnikov. Obliko datotek prepoznajte sami.

Napišite funkcijo `preberi_vozovnice(direktorij)`, ki prejme ime direktorija in vrne slovar, katerega ključni so imena potnikov, pripadajoče vrednosti pa slovar z vsemi drugimi podatki, zapisanimi v datoteki. SŽ predpostavljajo, da se imena potnikov ne ponavljajo.

Pomoč: funkcija `os.listdir(dir)` vrne imena vseh datotek v podanem direktoriju. Funkcija, ki naloži podatke v formatu XYZ, se navadno nahaja v modulu XYZ in se imenuje `load`, kot argument pa prejme (odprto) datoteko, iz katere mora naložiti podatke.