Izpit iz predmeta Programiranje 1, 24. junij 2025 **Čas pisanja: ~~60 minut~~ → 90 minut**

Dovoljena je uporaba literature na poljubnih medijih, prepovedana pa je komunikacija in uporaba inteligentnih pripomočkov.

1. Soda in liha

Napišite funkcijo `soda_liha(s)`, ki kot argument prejme seznam števil `s`. Vrne naj nov seznam: vanj izmenično shranite soda in liha števila iz seznama `s`, začenši s sodim. Oboja naj bodo v enakem vrstnem redu, kot so bila prej. Če je sodih več kot lihih ali obratno, naj odvečna števila ignorira.

Klic `soda_liha([5, 8, 4, 2, 8, 17, 13, 10, 2, 4, 6, 8, 9])` vrne `[8, 5, 4, 17, 2, 13, 8, 9]`. Soda števila so v enakem zaporedju kot prej (8, 4, 2, 8), med njimi pa so liha v enakem zaporedju kot prej (5, 17, 13, 9). Odvečna soda števila (10, 2, 4, 6, 8) so izpuščena. (Pazi: izpuščeno je tudi število 10! V novem seznamu so vedno pari sodih in lihih števil!)

2. Prafaktorji

Za začetek napišite funkcijo `prafaktorji(n)`, ki razcepi podano število `n` na prafaktorje in vrne razcep v obliki slovarja. Če pokličemo `prafaktorji(1400)`, vrne slovar `{2: 3, 5: 2, 7: 1}`, saj je $1400 = 2^3 5^2 7^1$.

Za res pa napišite funkcijo `gcd(a, b)`, ki prejme dva slovarja, kakršna vrača prejšnja naloga, in vrne največji skupni delitelj števil, ki ju predstavljata tadva slovarja. Če pokličemo `gcd({2: 3, 5: 2, 7: 1}, {2: 2, 7: 2, 11:1})`, vrne 28. Prvi slovar namreč predstavlja število 1400 in drugi število 2156, njun največji skupni delitelj pa je 28. Nalogo rešite, ne da bi izračunali števili (npr. 1400 in 2156): delajte s slovarjema, ki ste ju dobili!

Namig: $1400 = 2^3 5^2 7^1$ in $2156 = 2^2 7^2 11^1$, zato je njun največji skupni delitelj enak $2^2 7^1$.

3. Preveri vsoto

Napišite rekurzivno funkcijo `preveri_vsoto(s, n)`, ki prejme seznam števil in *domnevno* vsoto teh števil. Funkcija vrne `True`, če je *domnevna* vsota točna in `False`, če ni.

Rekurzivna naj bo prav ta funkcija s prav temi argumenti. Ne pišite nobenih pomožnih funkcij!

4. Čakalnica

Napišite razred `Cakalnica` z naslednjimi metodami:

- `prihod(ime, cas)` zabeleži, da je ob podanem času v čakalnico vstopila oseba s podanim imenom. Predpostaviti smete, da se metoda pokliče v trenutku, ko oseba vstopi. To pomeni, da bo čas pri vsakem klicu večji od časa pri predhodnem klicu;
- `cakajocih()` vrne število oseb v čakalnici;
- `naslednji(cas)` vrne ime naslednje osebe, ki je na vrsti. Osebe pridejo na vrsto v takšnem vrstnem redu, v kakršnem so prihajale v čakalnico. Če je čakalnica trenutno prazna, vrne `None`. Metoda kot argument prejme trenutni čas; potrebuje ga zaradi naslednjih dveh metod;
- `skupni_cas_cakanja()` vrne skupni čas čakanja vseh oseb, ki so že bile na vrsti (ne pa tistih, ki so še v čakalnici);
- `povprecni_cas_cakanja()` vrne povprečni čas čakanja za vse osebe, ki so že bile na vrsti (ali 0, če ni bila na vrsti še nobena oseba).

Konstruktor napišite, če menite, da ga potrebujete. Argumentov pa ne sme imeti.

Čas je vedno podan oz. vrnjen v urah. Če nekdo pride ob 10:45, je `cas` enak 10.75 (10 in tri četrt). Če je skupni čas čakanja uro in pol, metoda `skupni_cas_cakanja` vrne 1.5 (ne 1.30). Skratka: ne kompliciraj s časi, mirno jih seštevajte in odštevajte.