

Zadatak za završni ispit

Grupa A

Pažljivo pročitati zadatak i slediti sva uputstva i napomene navedene u postavci zadatka i na ovom listu papira.

Napraviti direktorijum na Dekstop-u i nazvati ga **GA_BROJ_INDEKSA**, npr. **GA_2008213514**.

Otvoriti IDLE za Python 3.5 i sve datoteke Python programa čuvati u prethodno napravljenom direktorijumu.

Na samom početku izvršavanja, program treba na standardni izlaz (na konzolu) da ispiše ime i prezime, broj indeksa i datum izrade zadatka, tj. današnji datum, sve u jednom redu, npr:

Miloš Mravik, 2014200501, 23.12.2019.

Postavka zadatka:

Program uzima podatke za rad iz datoteke koja se zove "ulaz.dat", raspoređuje ih u listu adekvatnih struktura u obliku rečnika koji sadrži ključeve za čuvanje tih podataka, a zatim vrši obradu na način koji je naveden u postavci i kao rezultat vraća filtriranu listu. Podatke o tim filtriranim strukturama zatim upisuje u datoteku "izlaz.txt".

1. Napraviti modul `industrijski_robot` koji u sebi ima konstruktorsku funkciju *IndustrijskiRobot* koja uzima argumente za *naziv proizvođača*, *model*, *godinu proizvodnje*, *broj stepeni slobode kretanja*, *potrošnju električne energije*, *maksimalni garantovani broj sati rada* i *vrstu pogona pokretnih delova*. Funkcija treba da napravi strukturu u obliku rečnika koja sadrži ključeve za sve navedene vrednosti.

Za sve broježane argumente funkcije proveriti da li su vrednosti moguće, npr. da su pozitivne vrednosti.

Samo za argument za vrstu pogona pokretnih delova proveriti da li je jedna od tri moguće vrednosti, kao tekst. Moguće vrednosti su: "hidraulični pogon", "elektromotorni pogon" ili "magnetni linearni aktuator".

Ukoliko neki od podataka koji se proveravaju, a pogotovu vrsta pogona pokretnih delova, nemaju dozvoljenu vrednost, umesto rečnika, funkcija treba da vrati vrednost -1.

2. U modulu `industrijski_robot` definisati funkciju koja se zove *efikasnost* koja vraća realnu vrednost, a kao argument uzima strukturu u obliku rečnika koji sadrži podatke o industrijskom robotu.

Funkcija vraća vrednost po formuli:

$$Efikasnost = \frac{Potrošnja\ električne\ energije * L}{(Broj\ stepeni\ slobode\ kretanja - 2) * \sqrt{Maksimalan\ broj\ sati\ rada}}$$

- L je 1.00 ako je vrsta pogona pokretnih delova magnetni linearni aktuator;
- L je 2.00 ako je vrsta pogona pokretnih delova elektromotorni pogon;
- L je 1.33 ako je vrsta pogona pokretnih delova hidraulični pogon.

3. U modulu `industrijski_robot` definisati funkciju `load` koji uzima jedan argument tekstualnog tipa koji predstavlja ime datoteke koju treba da pročita, a kao rezultat vraća listu struktura u obliku rečnika koje sadrže podatke o industrijskim robotima, dobijene na osnovu podataka iz datoteke. Funkcija treba da pročita sadržaj datoteke i da na osnovu podataka iz te datoteke formira listu rečnika sa podacima i industrijskim robotima.

Podaci su upisani u formatu prikazanom ispod. Podaci o jednom industrijskom robotu se prostiru u ukupno dva reda teksta u datoteci. Format prikazuje raspored podataka za samo jednog robota.

8 mesta	8 mesta	5 m.	9 mesta
2753.12	95000	2007	ACME
6	MLA	Cookie maker	5

┘ označava prelazak u novi red teksta u datoteci.

- 1. podatak u 1. redu je potrošnja električne energije.
- 2. podatak u 1. redu je maksimalan broj sati rada.
- 3. podatak u 1. redu je godina proizvodnje.
- 4. podatak u 1. redu je string koji se prostire do samog kraja tog reda i predstavlja naziv proizvođača robota.
- 1. podatak u 2. redu je broj stepeni slobode kretanja.
- 2. podatak u 2. redu je string napisan velikim slovima i predstavlja akronim vrste pogona pokretnih delova:
 - MLA - magnetni linearni aktuator;
 - EMP - elektromagnetni pogon;
 - HP - hidraulični pogon.
- 3. podatak u 2. redu je string koji se prostire do kraja tog reda i predstavlja naziv modela robota.

Svaka sledeća dva reda u datoteci sadrže podatke za još po jednog industrijskog robota.

Pre nego što vrati listu, funkcija mora da je sortira listu u rastući poredak prema iznosu efikasnosti, a tek onda da vrati tako sortiranu listu.

Ukoliko prilikom rada sa datotekom dođe do greške, funkcija treba na ekran da ispiše adekvatnu poruku o grešci.

4. U glavnom modulu programa, pozivom funkcije `load` iz modula `industrijski_robot` pribaviti listu koju ta funkcija vraća nakon čitanja podataka iz datoteke pod nazivom "ulaz.dat". Postupkom selekcije formirati novu listu koja će sadržati samo one industrijske robote čija efikasnost je u rasponu od 50% do 150% srednje vrednosti efikasnosti svih industrijskih robota u prvobitnoj listi.
5. U datoteku sačuvati podatke o svim elementima nove, filtrirane, liste, dobijene izvršavanjem koda opisanog u prethodnoj stavci zadatka. Novu listu upisati u JSON formatu u datoteku čije ime treba da bude formirano na sledeći način: "izlaz_N_M.json", gde je N broj elemenata prvobitne liste, a M je broj elemenata filtrirane liste, npr. "lista_10_4.json" ako je prvobitna lista imala 10 elemenata, a u filtriranoj listi su ostala samo 4 elementa.

Kada završite projekat **obavezno** sačuvajte sve datoteke, a zatim **ceo direktorijum** Vašeg projekta kopirajte u **mrežni direktorijum KOLOKVIJUM** koji se nalazi na Desktop-u, pa u pod-direktorijum **PROGRAMIRANJE 1 - PYTHON**.

Ako ne iskopirate svoj projekat u mrežni direktorijum **KOLOKVIJUM** i napustite učionicu **nećete moći naknadno** da predate svoj rad, a oni radovi koji nisu prekopirani u mrežni direktorijum **KOLOKVIJUM** neće biti pregledani ako ostanu samo na računaru na kojem ste radili. Samo oni radovi koji su kopirani se smatraju predati na ocenjivanje!