

## Uputstvo za izradu zadatka:

Nenulti broj poena može dobiti samo zadatak koji *nešto* radi. Zadaci koji ne rade *ništa* automatski se boduju sa 0 poena, bez obzira na količinu otkucanog programskog koda. Maksimalan broj poena (10) se može dobiti pod uslovom da kandidat ispoštuje specifikaciju zadatka do kraja i da pri rešavanju ni jednom ne zatraži pomoć. Svako neopravdano odstupanje od specifikacije zadatka odnosi po 1 poen. Svaka suštinska pomoć dežurnog pri rešavanju nekog dela zadatka odnosi po 1 poen. Pojašnjenje formulacije zadatka se može tražiti proizvoljan broj puta i ne odnosi poene. Za izradu zadatka kandidati na raspolaganju imaju najviše 105 minuta, a za samu odbranu najviše 15 minuta.

### NAPOMENE:

- § Potprogrami ne smeju da direktno koriste globalne promenljive
- § Obratiti pažnju na pravilno oslobađanje memorije i zatvaranje datoteka
- § *nešto* znači da:
  - § barem jedan od potprograma radi ono što se u postavci traži
  - § izlazne datoteke se formiraju ili ispis podataka na ekran se vrši onako kako se u postavci traži

B00	Zadatak 1
-----	-----------

Podaci o svim studentima koji su izabrali Praktikum iz Programiranja kao izborni predmet se nalaze u prvoj ulaznoj tekst datoteci, za svakog studenta u zasebnom redu. Format jednog reda je: `ggbbbb Prezime Ime`, gde `gg` označava poslednje dve cifre godine upisa, `bbbb` redni broj indeksa. U drugoj ulaznoj datoteci se nalazi spisak studenata koji su prijavili izlazak na ispit, pri čemu je jedino poznato da su podaci o studentima u zasebnim redovima, i da svaki počinje sa `ggbbbb`, dok je format ostatka reda različit od reda do reda. Program treba da prepíše podatke o studentima koji su prijavili izlazak na ispit, i to tako da format bude isti kao kod prve ulazne datoteke. Broj studenata u datoteci nije unapred poznat. Potrebno je realizovati određene delove opisanog programa.

- § [1] Glavni program koji repetitivno učitava imena datoteka sa kojima će se raditi i poziva potrebne potprograme sve dok se za ime bilo koje od datoteka ne unese prazan string, te sadrži sve potrebne definicije tipova podataka i deklaracije promenljivih.
- § [1] Potprogram koji čita podatke o jednom studentu i smešta ih u zapis. Čitanje počinje od početka tekućeg reda prve ulazne datoteke i završava se na početku sledećeg reda.
- § [2] Potprogram koji koristi prethodni potprogram da učitava podatke o svim studentima u jednostruko ulančanu listu, takvu da ima elemente sa tačno dva polja. Prvo polje elementa je zapis sa podacima o studentu, drugo polje elementa je pokazivač na sledeći element liste. Novi elementi se umeću na kraj liste.
- § [1] Potprogram koji iz druge ulazne datoteke čita podatke o studentima prijavljenim za polaganje i broj indeksa svakog studenta smešta u jednostruko ulančanu listu stringova. Novi elementi se umeću na početak liste.
- § [2] Potprogram koji stvara izlaznu datoteku prema ranije opisanom pravilu, na osnovu podataka u navedenim listama. Za studente koji se nalaze u listi prijavljenih za ispit, a ne nalaze se u listi svih studenata na predmetu, na standardnom izlazu ispisati broj indeksa u zasebnom redu.
- § [2] Potprogram koji na osnovu dotadašnjeg rada programa<sup>1</sup> ispisuje broj prijavljenih studenata, i na osnovu broja raspoloživih termina, te vremena početka i maksimalnog broja studenata za svaki od termina (unose se sa standardnog ulaza) ispisuje raspored polaganja za svakog prijavljenog studenta na standardnom izlazu. Termina je najviše 10.
- § [1] Potprogram koji dealocira korišćenu dinamičku memoriju.

<sup>1</sup> izlazna datoteka, odnosno lista prijavljenih, zavisno od toga dokle se stiglo u realizaciji navedenih zahteva

## Uputstvo za izradu zadatka:

Nenulti broj poena može dobiti samo zadatak koji *nešto* radi. Zadaci koji ne rade *ništa* automatski se boduju sa 0 poena, bez obzira na količinu otkucanog programskog koda. Maksimalan broj poena (10) se može dobiti pod uslovom da kandidat ispoštuje specifikaciju zadatka do kraja i da pri rešavanju ni jednom ne zatraži pomoć. Svako neopravdano odstupanje od specifikacije zadatka odnosi po 1 poen. Svaka suštinska pomoć dežurnog pri rešavanju nekog dela zadatka odnosi po 1 poen. Pojašnjenje formulacije zadatka se može tražiti proizvoljan broj puta i ne odnosi poene. Za izradu zadatka kandidati na raspolaganju imaju najviše 105 minuta, a za samu odbranu najviše 15 minuta.

### NAPOMENE:

- § Potprogrami ne smeju da direktno koriste globalne promenljive
- § Obratiti pažnju na pravilno oslobađanje memorije i zatvaranje datoteka
- § *nešto* znači da:
  - § barem jedan od potprograma radi ono što se u postavci traži
  - § izlazne datoteke se formiraju/ispis podataka na ekran se vrši onako kako se u postavci traži

B03	Zadatak 2
-----	-----------

Podaci o svim demonstratorima Katedre za RTI se nalaze u tekst datoteci `demo_pr.txt`, za svakog demonstratora u zasebnom redu. Format jednog reda je: `ggggbbbb s prosek e-mail`, gde je `gggg` godina upisa, `bbbb` redni broj indeksa, a `s` godinu studija u koju je demonstrator upisan. U tekst datoteci `demo_akt.txt` se nalazi spisak aktivnosti demonstratora. Format jednog reda je: `bbbb/gg sifra_predmeta br_sati tip_aktivnosti`, na primer `0321/03 001PP1 2 1`. Program treba da prepíše podatke o demonstratorima, i to tako da za svakog demonstratora ispiše ukupan broj radnih sati i e-adresu. Broj demonstratora o kojima se podaci nalaze u datoteci nije unapred poznat. Potrebno je realizovati određene delove opisanog programa.

- § [1] Glavni program koji repetitivno učitava imena datoteka sa kojima će se raditi i poziva potrebne potprograme sve dok se za ime bilo koje od datoteka ne unese prazan string, te sadrži sve potrebne definicije tipova podataka i deklaracije promenljivih.
- § [1] Potprogram koji čita podatke o jednom demonstratoru i smešta ih u zapis. Čitanje počinje od početka tekućeg reda prve ulazne datoteke i završava se na početku sledećeg reda.
- § [2] Potprogram koji koristi prethodni potprogram da učitava podatke o svim demonstratorima u jednostruko ulančanu listu, takvu da ima elemente sa tačno dva polja. Prvo polje elementa je zapis sa podacima o demonstratoru, drugo polje elementa je pokazivač na sledeći element liste. Novi elementi se umeću na početak liste.
- § [1] Potprogram koji iz druge ulazne datoteke čita podatke o aktivnostima, i sve podatke smešta u jednostruko ulančanu listu, organizovanu analogno listi opisanoj u prethodnom potprogramu. Novi elementi se umeću na kraj liste.
- § [2] Potprogram koji stvara izlaznu datoteku, na osnovu podataka u navedenim listama. Format jednog reda: `ggggbbbb e-mail zbir_radnih_sati`. Za demonstratore koji nisu imali aktivnosti, ispisati broj indeksa i e-adresu na standardnom izlazu.
- § [1] Potprogram koji dealocira korišćenu dinamičku memoriju.
- § [2] Potprogram koji svakom demonstratoru dodeljuje redni broj (prema redosledu pojavljivanja u listi demonstratora), pa na standardnom izlazu za svakog od demonstratora ispisuje indeks i ukupan broj radnih sati. Potprogram treba da uzima u obzir samo one demonstratore čiji se redni brojevi nalaze u skupu čiji se elementi unose sa standardnog ulaza. Jednostavnosti radi, isključivo prilikom realizacije ovog potprograma, smatrati da demonstratora ima najviše 100.