

Programiranje 2 - drugi kolokvij, 27. 6. 2014.

Rezultati i uvidi u kolokvije: Rezultati u utorak 8. srpnja 2014. u 23:59, a uvidi u srijedu 9. srpnja 2014. u 12:00.

Napomene: Sva rješenja napišite isključivo na papir sa zadatkom jer jedino njega predajete. Pomoćne račune smijete raditi na drugim papirima koje će vam dati dežurni asistent. U svim zadacima nije dopušteno korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Dozvoljeno je korištenje isključivo pribora za pisanje i brisanje te službenog podsjetnika. Kalkulatori i razne tablice, papiri i sl. nisu dozvoljeni. **Mobitele ugasite i spremite;** nisu dozvoljeni kao zamjena za sat!

Zadatak 1 (25 bodova) Elementi vezane liste trebaju sadržavati po jedan string `grad` s najviše 20 znakova, dva prirodna broja `dan` i `mjesec` i jedan realan broj dvostruke preciznosti `temperatura`. Broj `temperatura` predstavlja temperaturu izmjerenu u gradu `grad` u 12 sati na dan čiji datum je `dan.mjesec`, pa zato možete pretpostaviti da ne postoje dva elementa s istim gradom i istim datumom.

- Napišite definiciju tipa podataka za pojedini element liste, na način da bude moguće definirati varijable naredbom `element item`; i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.
- Napišite funkciju `element *ubaci(element *first, element *new)`, gdje je `first` pokazivač na prvi element liste sortirane uzlazno po imenu grada, pa prema datumu, koja ubacuje element čiji je pokazivač `new` tako da lista i dalje bude sortirana na isti način te vraća pokazivač na početak liste.
- Napišite funkciju `element *odvoji(element **first, double t)` gdje je `first` pokazivač na adresu prvog elementa liste, koja formira novu listu u kojoj se nalaze elementi za koje je `temperatura` veća od `t`, a početnu listu modificira tako da se u njoj ne pojavljuju više elementi koji su prebačeni u novu listu. Funkcija treba vratiti pokazivač na prvi element nove liste.

Nije dozvoljeno koristiti polja!

Zadatak 2 (20 bodova) Tajna poruka je napisana kao niz linija, svaka od kojih završava znakom `\n`, i tako je upisana (kopirana) u $1 \leq n \leq 9$ tekstualnih datoteka imena `1.part`, `2.part`, `...`. Problem je u tome što datoteke nisu cjelovite: svaka ima točno toliko linija koliko ima tajna poruka, i i -ta linija je uvijek iste duljine u svim datotekama, ali u svakoj su datoteci neki znakovi zamijenjeni razmacima. Napišite program koji će rekonstruirati tajnu poruku i zapisati je u datoteku `secret.message`. Primjer:

1.part	2.part	3.part	secret.message
Ov je t jna	0 o e aj	v a a	Ovo je tajna
p .	oru .	o ka	poruka.

Ne smijete ništa pretpostavljati o broju linija, niti o duljini pojedine linije tajne poruke. Smijete koristiti jedan pomoćni niz. Program može otkriti n tako da redom otvara datoteke `i.part`, sve dok ne naiđe na neku koju ne može otvoriti.

Zadatak 3 (25 bodova)

(10 bodova) Definirajte strukture kojima ćete pamtit i informacije o vođenju bolnice koja radi u tri smjene.

- Svaka smjena ima ime voditelja, te liječnike (za svakog je dovoljno pamtit i ime) indeksirane brojem od 1 do 99 (neiskorišteni brojevi imaju "" za ime liječnika).
- U bolnici se događa niz aktivnosti, za čiji ukupan broj ne postoji ograničenje. Svaka aktivnosti ima tip (broj bez predznaka, 0 do 9: primjerice, operacija srca, pregled uha, operacija želuca, mjerenje tlaka,...) te indeks kirurga koji je izvršio operaciju ili -1 ako aktivnost nije operacija.

Pretpostavite da imena liječnika i voditelja smjene neće imati više od 50 znakova. U ovom zadatku je dopušteno korištenje dodatnih polja i stringova.

(10 + 5 bodova) Napišite funkciju `char **kirurzi(...)`. Funkcija prima smjenu bolnice (sami definirajte ulazne parametre) i vraća polje stringova: za svakog kirurga tip aktivnosti, razmak, i njegovo ime. Svakog kirurga treba navesti onoliko puta u koliko aktivnosti se pojavljuje. Duljinu povratnog polja vratite kao varijabilni argument. Za dodatnih 5 bodova, sortirajte to polje stringova uzlazno po tipu aktivnosti, te za isti tip, po imenu kirurga.

Programiranje 2 - drugi kolokvij, 27. 6. 2014.

Rezultati i uvidi u kolokvije: Rezultati u utorak 8. srpnja 2014. u 23:59, a uvidi u srijedu 9. srpnja 2014. u 12:00.

Napomene: Sva rješenja napišite isključivo na papir sa zadatkom jer jedino njega predajete. Pomoćne račune smijete raditi na drugim papirima koje će vam dati dežurni asistent. U svim zadacima nije dopušteno korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavljje `math.h`), osim ako je u zadatku drugačije navedeno.

Dozvoljeno je korištenje isključivo pribora za pisanje i brisanje te službenog podsjetnika. Kalkulatori i razne tablice, papiri i sl. nisu dozvoljeni. **Mobitele ugasite i spremite;** nisu dozvoljeni kao zamjena za sat!

Zadatak 1 (25 bodova) Elementi vezane liste trebaju sadržavati po jedan string `bakterija` s najviše 20 znakova, dva prirodna broja `sat` i `minute` i jedan realan broj dvostruke preciznosti `kolicina`. Broj `kolicina` predstavlja količinu bakterija imena `bakterija` izmjerenu u vrijeme `sat:minute`. Možete pretpostaviti da su sva mjerenja učinjena istog dana i da ne postoje dva elementa s istom bakterijom i istim vremenom mjerenja.

- Napišite definiciju tipa podataka za pojedini element liste, na način da bude moguće definirati varijable naredbom `element item;` i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.
- Napišite funkciju `element *ubaci(element *first, element *novi)`, gdje je `first` pokazivač na prvi element liste sortirane uzlazno po imenu bakterija, pa prema vremenu mjerenja, koja ubacuje element čiji je pokazivač `novi` tako da lista i dalje bude sortirana na isti način te vraća pokazivač na početak liste.
- Napišite funkciju `element *odvoji(element **first, double q)` gdje je `first` pokazivač na adresu prvog elementa liste, koja formira novu listu u kojoj se nalaze elementi za koje je `kolicina` manja od `q`, a početnu listu modificira tako da se u njoj ne pojavljuju više elementi koji su prebačeni u novu listu. Funkcija treba vratiti pokazivač na prvi element nove liste.

Nije dozvoljeno koristiti polja!

Zadatak 2 (20 bodova) Tajna poruka je napisana kao niz linija, svaka od kojih završava znakom `\n`, i tako je upisana (kopirana) u $1 \leq n \leq 26$ tekstualnih datoteka imena `part.A`, `part.B`, `part.C` Problem je u tome što datoteke nisu cjelovite: svaka ima točno toliko linija koliko ima tajna poruka, i i -ta linija je uvijek iste duljine u svim datotekama, ali u svakoj su datoteci neki znakovi zamijenjeni točkama (`'.'`). Napišite program koji će učitati n s tipkovnice, rekonstruirati tajnu poruku i ispisati je na ekran. Primjer:

<code>part.A</code>	<code>part.B</code>	<code>part.C</code>	ispis:
<code>Ov. je t.jna</code>	<code>O.o .e .aj..</code>	<code>.v.....a..a</code>	Ovo je tajna
<code>p.....</code>	<code>.oru...</code>	<code>.o..ka.</code>	poruka.

Ne smijete ništa pretpostavljati o broju linija, niti o duljini pojedine linije tajne poruke. Smijete koristiti jedan pomoćni niz.

Zadatak 3 (25 bodova)

(10 bodova) Definirajte strukture kojima ćete pamtit i informacije o školi s četiri razreda.

- Svaki razred ima ime učiteljice, te učenike (za svakog je dovoljno pamtit i ime) indeksirane brojem od 1 do 99 (neiskorišteni brojevi imaju "" za ime učenika).
- U školi se događa niz aktivnosti, za čiji ukupan broj ne postoji ograničenje. Svaka aktivnosti ima tip (broj bez predznaka, 0 do 9: primjerice, ispit iz matematke, roditeljski sastanak, ispit iz hrvatskog, popravak prozora,...) te indeks učenika koji je usmeno odgovarao ili -1 ako aktivnost nije usmeni ispit.

Pretpostavite da imena učenika i učiteljica neće imati više od 50 znakova. U ovom zadatku je dopušteno korištenje dodatnih polja i stringova.

(10 + 5 bodova) Napišite funkciju `char **ispitanici(...)`. Funkcija prima razred škole (sami definirajte ulazne parametre) i vraća polje stringova: za svakog učenika koji je usmeno odgovarao tip aktivnosti, razmak, i njegovo ime. Svakog učenika treba navesti onoliko puta u koliko aktivnosti se pojavljuje. Duljinu povratnog polja vratite kao varijabilni argument. Za dodatnih 5 bodova, sortirajte to polje stringova uzlazno po tipu aktivnosti, te za isti tip, po imenu učenika.

Programiranje 2 - drugi kolokvij, 27. 6. 2014.

Rezultati i uvidi u kolokvije: Rezultati u utorak 8. srpnja 2014. u 23:59, a uvidi u srijedu 9. srpnja 2014. u 12:00.

Napomene: Sva rješenja napišite isključivo na papir sa zadatkom jer jedino njega predajete. Pomoćne račune smijete raditi na drugim papirima koje će vam dati dežurni asistent. U svim zadacima nije dopušteno korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Dozvoljeno je korištenje isključivo pribora za pisanje i brisanje te službenog podsjetnika. Kalkulatori i razne tablice, papiri i sl. nisu dozvoljeni. **Mobitele ugasite i spremite;** nisu dozvoljeni kao zamjena za sat!

Zadatak 1 (25 bodova) Elementi vezane liste trebaju sadržavati po jedan string `drzava` s najviše 20 znakova i tri prirodna broja `mjesec`, `godina` i `stanovnik`. Broj `stanovnik` predstavlja broj stanovnika u državi `drzava` prvog dana mjeseca `mjesec` godine `godina`, pa zato možete pretpostaviti da ne postoje dva elementa s istom državom i istim mjesecom i godinom.

- Napišite definiciju tipa podataka za pojedini element liste, na način da bude moguće definirati varijable naredbom `element item`; i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.
- Napišite funkciju `element *ubaci(element *first, element *new)`, gdje je `first` pokazivač na prvi element liste sortirane uzlazno po imenu države, pa prema godini i mjesecu, koja ubacuje element čiji je pokazivač `new` tako da lista i dalje bude sortirana na isti način te vraća pokazivač na početak liste.
- Napišite funkciju `element *odvoji(element **first, int s)` gdje je `first` pokazivač na adresu prvog elementa liste, koja formira novu listu u kojoj se nalaze elementi za koje je `stanovnik` veći od `s`, a početnu listu modificira tako da se u njoj ne pojavljuju više elementi koji su prebačeni u novu listu. Funkcija treba vratiti pokazivač na prvi element nove liste.

Nije dozvoljeno koristiti polja!

Zadatak 2 (20 bodova) Tajna poruka je napisana kao niz linija, svaka od kojih završava znakom `\n`, i tako je upisana (kopirana) u $1 \leq n \leq 99$ tekstualnih datoteka imena `dio.01`, `dio.02`, ... Problem je u tome što datoteke nisu cjelovite: svaka ima točno toliko linija koliko ima tajna poruka, i i -ta linija je uvijek iste duljine u svim datotekama, ali u svakoj su datoteci neki znakovi zamijenjeni crticama ('-'). Napišite program koji će učitati n s tipkovnice, rekonstruirati tajnu poruku i zapisati je u datoteku `tajna-poruka`. Primjer:

<code>dio.01</code>	<code>dio.02</code>	<code>dio.03</code>	<code>tajna-poruka</code>
<code>0v- je t-jna</code>	<code>0-o -e--aj--</code>	<code>-v-----a--a</code>	<code>Ovo je tajna</code>
<code>p-----.</code>	<code>-oru---</code>	<code>-o--ka-</code>	<code>poruka.</code>

Ne smijete ništa pretpostavljati o broju linija, niti o duljini pojedine linije tajne poruke. Smijete koristiti jedan pomoćni niz.

Zadatak 3 (25 bodova)

(10 bodova) Definirajte strukture kojima ćete pamtit i informacije o utakmici između dvije reprezentacije.

- Svaka reprezentacija ima ime, te igrače (za svakog je dovoljno pamtit i ime) indeksirane brojem na dresu (koji može biti 1 do 99; neiskorišteni brojevi imaju "" za ime igrača).
- Na svakoj utakmici dogodi se niz akcija, za čiji ukupan broj ne postoji ograničenje. Svaka akcija ima tip (broj bez predznaka, 0 do 9: primjerice, gol glavom, stativa, gol nogom, obranjen udarac,...) te broj dresa strijelca ili -1 ako akcijom nije postignut gol.

Pretpostavite da imena nogometaša i reprezentacija neće imati više od 50 znakova. U ovom zadatku je dopušteno korištenje dodatnih polja i stringova.

(10 + 5 bodova) Napišite funkciju `char **strijelci(...)`. Funkcija prima reprezentaciju na utakmici (sami definirajte ulazne parametre) i vraća polje stringova: za svakog strijelca tip akcije, razmak, i njegovo ime. Svakog strijelca treba navesti onoliko puta u koliko akcija se pojavljuje. Duljinu povratnog polja vratite kao varijabilni argument. Za dodatnih 5 bodova, sortirajte to polje stringova uzlazno po tipu akcije, te za isti tip, po imenu strijelca.

Programiranje 2 - drugi kolokvij, 27. 6. 2014.

Rezultati i uvidi u kolokvije: Rezultati u utorak 8. srpnja 2014. u 23:59, a uvidi u srijedu 9. srpnja 2014. u 12:00.

Napomene: Sva rješenja napišite isključivo na papir sa zadatkom jer jedino njega predajete. Pomoćne račune smijete raditi na drugim papirima koje će vam dati dežurni asistent. U svim zadacima nije dopušteno korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavljje `math.h`), osim ako je u zadatku drugačije navedeno.

Dozvoljeno je korištenje isključivo pribora za pisanje i brisanje te službenog podsjetnika. Kalkulatori i razne tablice, papiri i sl. nisu dozvoljeni. **Mobitele ugasite i spremite;** nisu dozvoljeni kao zamjena za sat!

Zadatak 1 (25 bodova) Elementi vezane liste trebaju sadržavati po jedan string rijeka s najviše 20 znakova, dva prirodna broja mjesec i godina i jedan realan broj dvostruke preciznosti koncentracija. Broj koncentracija predstavlja koncentraciju štetnih tvari u rijeci rijeka u mjesecu mjesec godine godina, pa zato možete pretpostaviti da ne postoje dva elementa s istom rijekom i istim mjesecom i godinom.

- Napišite definiciju tipa podataka za pojedini element liste, na način da bude moguće definirati varijable naredbom `element item;` i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.
- Napišite funkciju `element *ubaci(element *first, element *novi)`, gdje je `first` pokazivač na prvi element liste sortirane uzlazno po imenu rijeke, pa prema godini i mjesecu, koja ubacuje element čiji je pokazivač `novi` tako da lista i dalje bude sortirana na isti način te vraća pokazivač na početak liste.
- Napišite funkciju `element *odvoji(element **first, double c)` gdje je `first` pokazivač na adresu prvog elementa liste, koja formira novu listu u kojoj se nalaze elementi za koje je koncentracija veća od `c`, a početnu listu modificira tako da se u njoj ne pojavljuju više elementi koji su prebačeni u novu listu. Funkcija treba vratiti pokazivač na prvi element nove liste.

Nije dozvoljeno koristiti polja!

Zadatak 2 (20 bodova) Tajna poruka je napisana kao niz linija, svaka od kojih završava znakom $\backslash n$, i tako je upisana (kopirana) u $1 \leq n \leq 26$ tekstualnih datoteka imena `a.dio`, `b.dio`, `c.dio` . . . Problem je u tome što datoteke nisu cjelovite: svaka ima točno toliko linija koliko ima tajna poruka, i i -ta linija je uvijek iste duljine u svim datotekama, ali u svakoj su datoteci neki znakovi zamijenjeni znakovima '#'. Napišite program koji će rekonstruirati tajnu poruku i ispisati je na ekran. Primjer:

<code>a.dio</code>	<code>b.dio</code>	<code>c.dio</code>	ispis:
<code>Ov# je t#jna</code>	<code>O#o #e #aj##</code>	<code>#v#####a##a</code>	<code>Ovo je tajna</code>
<code>p#####.</code>	<code>#oru##.</code>	<code>#o##ka#</code>	<code>poruka.</code>

Ne smijete ništa pretpostavljati o broju linija, niti o duljini pojedine linije tajne poruke. Smijete koristiti jedan pomoćni niz. Program može otkriti n tako da redom otvara datoteke `l.dio`, sve dok ne nađe na neku koju ne može otvoriti.

Zadatak 3 (25 bodova)

(10 bodova) Definirajte strukture kojima ćete pamtit i informacije o fakultetu sa šest odsjeka.

- Svaki odsjek ima ime, te profesore (za svakog je dovoljno pamtit i ime) indeksirane brojem od 1 do 99 (neiskorišteni brojevi imaju "" za ime profesora).
- Na fakultetu se događa niz aktivnosti, za čiji ukupan broj ne postoji ograničenje. Svaka aktivnosti ima tip (broj bez predznaka, 0 do 9: primjerice, usmeni ispit, odlazak u mirovinu, kolokvij, popravak prozora,...) te indeks profesora koji je zadavao ispit ili -1 ako aktivnost nije ispit.

Pretpostavite da imena profesora i odsjeka neće imati više od 50 znakova. U ovom zadatku je dopušteno korištenje dodatnih polja i stringova.

(10 + 5 bodova) Napišite funkciju `char **ispitivaci(...)`. Funkcija prima odsjek fakulteta (sami definirajte ulazne parametre) i vraća polje stringova: za svakog ispitivača tip aktivnosti, razmak, i njegovo ime. Svakog profesora treba navesti onoliko puta u koliko aktivnosti se pojavljuje. Duljinu povratnog polja vratite kao varijabilni argument. Za dodatnih 5 bodova, sortirajte to polje stringova uzlazno po tipu aktivnosti, te za isti tip, po imenu profesora.