

Smijete koristiti pisane materijale ili vlastite bilješke spremljene u HOME direktoriju, web-stranicu kolegija i linkove koji vode s nje.

Strogo je zabranjeno korištenje ChatGPT-a te će se isto provjeravati za svaki kod. Ne smijete se koristiti nikakvim sredstvima komunikacije (papirići, mobitel, mail, chat, messenger i sl.). Svaki pokušaj prepisivanja i dogovaranja rezultirat će trenutnim udaljavanjem sa kolokvija. Strogo je zabranjeno fotografiranje ili pretipkavanje teksta zadataka.

Iznimno, studenti koji kolokvij pišu u čitaonici, smiju upaliti hotspot na mobitelu, nakon čega mobitel moraju okrenuti licem prema stolu na prikladnoj udaljenosti. Mobitel se nakon toga više ne smije dirati do kraja izvođenja kolokvija.

Prvi zadatak (15 bodova)

Na početku sezone radi se organizacija igrača u košarkaškim klubovima.

Napišite parametriziranu strukturu *igrac* koja sadrži sljedeće informacije o igraču:

- generički tip **T** za oznaku igrača (*string* ako je oznaka prezime, *int* ako je broj na dresu)
- *string* za poziciju koju igra

Oznaka igrača je jedinstvena. Moguće pozicije igrača su sljedeći stringovi: "razigravac", "suter", "krilo", "krilni centar", "centar".

Igrača trebamo moći definirati naredbom oblika: npr. `igrac<string> player("Jordan","suter");`

Dodatno, trebamo moći deklarirati polje igrača `igrac<string> niz[20];`

Napišite parametriziranu strukturu *klub* koja sadrži sljedeće informacije o košarkaškom klubu:

- *string* za ime kluba
- *igrac<T>** za polje igrača koji igraju za klub
- *int* za trenutni broj igrača u klubu

Poznato je da svaki klub može imati najviše 20 igrača. Klub trebamo moći definirati naredbama oblika: npr. `klub<string> KK1("Chicago Bulls",niz,0);` ili `klub<string> KK2(KK1);`

Za strukturu *klub* implementirajte sljedeće funkcije:

- destruktor - oslobađa memoriju zauzetu za niz igrača
- `void dodaj_igraca (igrac<T> player)` - dodaje igrača player na kraj niza
- `void obrisi_igraca (igrac<T> player)` - briše igrača player iz niza
- `void zastupljene_pozicije (string* pozicije)` - preko varijabilnog argumenta vraća niz zastupljenih pozicija u klubu tako da svaku od njih zapiše točno jednom
- `void ispis_igraca ()` - ispisuje sve igrače iz kluba u formatu

oznaka1: pozicija1

oznaka2: pozicija2

oznaka3: pozicija3

Nije bitan poredak igrača u nizu. Kod implementacije funkcija pripazite na rubne slučajeve.

U funkciji `main()` deklarirajte klub Chicago Bulls te mu dodajte igrače (Jordan, suter), (Bird, krilo), (LeBron, krilni centar), (West, razigravac), (Iverson, centar). Ispišite popis svih igrača u klubu. Izbrišite igrače Bird i West te zatim ispišite popis zastupljenih pozicija u klubu.

Napomena: Rješenje rastavite na tri datoteke. Smijete dodavati po potrebi konstruktore i funkcije u definirane strukture da omogućite implementaciju rješenja. **Nije dozvoljeno koristiti STL.**

IME I PREZIME:

MATIČNI BROJ STUDENTA:

BODOVI:

Drugi zadatak (20 bodova)

Na manifestaciji Dan i noć na PMF-u sudjeluju studenti volonteri koji održavaju razne radionice. Na Matematičkom odsjeku svaka radionica smještena je u neku predavaonicu (možete pretpostaviti da je u svakoj predavaonici najviše jedna radionica), a svaka predavaonica nalazi se na nekom katu. Oznaka predavaonice sastoji se od nekoliko znakova, od kojih je prvi uvijek oznaka kata: 0 za predavaonice u prizemlju, 1 za prvi kat itd. Pretpostavljamo da ne postoje predavaonice s "negativnom" oznakom kata (podrum). Za uspješno završenu radionicu posjetiteljima volonteri dijele naljepnice. Svaka radionica počinje manifestaciju s nekim brojem naljepnica, a nakon nekoliko sati potrebno je dopuniti zalihe naljepnica budući da ih se puno podijelilo.

Informacije o radionicama i naljepnicama spremamo u varijablu tipa

```
map<string, tuple<string, int, multiset<int>>>>.
```

Ključevi su oznake predavaonica, a vrijednost je uređena trojka. Prvi element je ime radionice (`string`), drugi element broj naljepnica na početku manifestacije (`int`). Treći element je multiskup u kojem svaki broj predstavlja broj podijeljenih naljepnica za nekog od volontera u toj predavaonici (zbroj elemenata tog multiskupa odgovara ukupnom broju podijeljenih naljepnica u predavaonici).

Napišite funkciju `nedostaju_naljepnice` koja prima informacije o radionicama i naljepnicama (varijabla tipa `map<string, tuple<string, int, multiset<int>>>`) i ispisuje predavaonice i pripadne radionice u kojima su podijeljene sve naljepnice. Predavaonice ispišite odvojeno po katovima, dakle prvo sve takve predavaonice u prizemlju (označeno kao "prizemlje", a ne kao "0. kat"), pa na 1. katu itd. Pri ispisu poredak predavaonica na katu nije bitan, a ako na nekom katu ne postoji niti jedna predavaonica u kojoj nedostaju naljepnice, oznaka kata se ne ispisuje. Format ispisa ne mora biti identičan primjeru na sljedećoj stranici.

Napišite `main` koji učitava informacije o radionicama i naljepnicama te koristeći gornju funkciju ispisuje predavaonice i pripadne radionice u kojima su podijeljene sve naljepnice.

Informacije o radionicama i naljepnicama učitavamo s ulaza u sljedećem formatu:

<i>oznakaPredavaonice₁</i>	...
<i>imeRadionice₁</i>	<i>oznakaPredavaonice_n</i>
<i>pocetniBrojNaljepnica₁</i>	<i>imeRadionice_n</i>
<i>podijeljenoNaljepnica₁₁</i>	<i>pocetniBrojNaljepnica_n</i>
<i>podijeljenoNaljepnica₁₂</i>	<i>podijeljenoNaljepnica_{n1}</i>
...	...
−1	−1
<i>oznakaPredavaonice₂</i>	<i>kraj</i>
<i>imeRadionice₂</i>	
<i>pocetniBrojNaljepnica₂</i>	
<i>podijeljenoNaljepnica₂₁</i>	
...	
−1	

Ako je za oznaku predavaonice upisan "kraj" stajemo s unosom. Količinu podijeljenih naljepnica za određenu predavaonicu unosimo dok kao količinu ne unesemo −1.

IME I PREZIME:

MATIČNI BROJ STUDENTA:

BODOVI:

Primjer:Ulaz:

006

igraonica

50

20

20

-1

005

potjera

160

150

10

-1

201

kriptografija

40

10

10

10

10

-1

101

kombinatorna

100

30

50

20

-1

104

escaperoom

20

20

-1

kraj

Izlaz:

Prizemlje:

005 (potjera)

1. kat:

101 (kombinatorna)

104 (escaperoom)

2. kat:

201 (kriptografija)