

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

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 hostspot 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 (20 bodova)

Napišite parametriziranu strukturu **par** koja reprezentira uređene parove (x,y), gdje je x generičkog tipa **T**, a y generičkog tipa **S** (smijete pretpostaviti da su **T** i **S** jedan od tipova **char, int, float, double**). Par trebamo moći definirati naredbom oblika: npr. `par<int,double> tmp(1,12.3);`

Napišite parametriziranu strukturu prioritetni red tako da je moguće red definirati naredbom oblika: npr. `pqueue<int,double> R;`

Prioritetni red kao elemente prima parove definirane strukturom **par**. Red je parametriziran s dva parametra tipa koji odgovaraju parametrima tipa strukture **par**. Red implementirajte kao jednostruko povezanu listu koristeći pokazivače. Prioritetni red je sortiran silazno prema prioritetu (uređaju definiranim nad parovima): (a,b) > (c,d) akko (b>d) ili (b = d i a>c). Možete pretpostaviti da će svi elementi prioritetnog reda biti međusobno različiti.

Za strukturu **pqueue** implementirajte:

- konstruktor bez parametara
- `par<T,S> top()` - vraća element reda s najvećim prioritetom. Kod praznog reda vraća par konstruiran konstruktorom bez parametara.
- `int size()` - vraća broj elemenata reda (broj elemenata je spremljen u varijablu članicu strukture **pqueue**)
- `bool empty()` - vraća logičku vrijednost *istina* ukoliko je red prazan, inače *laž*
- `void push(par<T,S> el)` – dodaje element u prioritetni red prema prioritetima definiranim uređajem nad parovima, element s najvišim prioritetom se nalazi na početku reda.
- `void pop()` - izbacuje element s najvišim prioritetom iz reda
- `void print_pqueue()` - ispisuje elemente reda od elementa s najvišim prioritetom do elementa s najnižim prioritetom bez promjene reda.

Napomena: Smijete implementirati dodatne strukture ukoliko je potrebno za rješavanje zadatka. Smijete dodavati po potrebi konstruktore i funkcije u strukturu **par** da omogućite implementaciju strukture **pqueue**. Preporučeno je (ne obavezno) da testirate implementaciju glavnim programom. **Nije dozvoljeno koristiti STL.**

Primjer: Nakon ubacivanja parova: (1, 12.3), (12,10.0), (11,10.0), (45,17.4), red će biti: {(45,17.4), (1,12.3), (12,10.0), (11,10.0)}, veličina reda će biti 4 i funkcija **empty** će vratiti logičku vrijednost *laž*. Funkcija **top** će vratiti par (45,17.4).

PREZIME I IME:

MATIČNI BROJ STUDENTA:

BODOVI:

Drugi zadatak (15 bodova)

Veslačka sekcija PMF-a naručuje *merch* (predmeti s logom Veslačke sekcije PMF-a, npr. majica) za svoje veslače. Informacije o narudžbi učitavamo s ulaza u sljedećem obliku:

```
imeVeslača1
predmet1,1:velicina1,1
predmet1,2:velicina1,2
...
"tjt"
imeVeslača2
predmet2,1:velicina2,1
...
"tjt"
...
imeVeslačan
predmetn,1:velicinan,1
...
"tjt"
"kraj"
```

Dakle, ako je za ime veslača upisan string **"kraj"**, stajemo s unosom. Inače, unosimo informacije o narudžbi tog veslača sve dok se ne unese string **"tjt"**. Pretpostavljamo da se nitko ne zove **"kraj"**.

Informacije spremamo u varijablu tipa **map<string, vector<pair<string, string>>>**. Ključevi su imena veslača, a vrijednost je vektor parova pri čemu je prvi element para naručeni predmet, a drugi element je veličina (npr. S, M, L). Veličina može biti i broj, ali ju pamtimo kao string.

Napišite funkciju **stvariNarudzbu** koja prima informacije o narudžbi svakog veslača (varijabla tipa **map<string, vector<pair<string, string>>>**), a vraća **map<string, map<string, int>>**. Ključevi su svi naručeni predmeti. Svakom predmetu pridružen je **map** u kojoj je zapisana ukupna količina (**int**) pojedine veličine (**string**) za taj predmet.

Napišite **main** koji učitava narudžbe veslača te koristeći funkciju **stvariNarudzbu** stvara narudžbu i ispisuje ju (ispisuje predmet te koliko je koje veličine naručeno - vidi primjer). Format ispisa odaberite sami, ne mora biti kao u primjeru.

Ulaz:

Lucija
majica:M
carape:39
tjt
Tonka
kombinezon:M
carape:39
majica:M
tjt
Ivan
kombinezon:L
carape:42
tjt
Helena
carape:39
tjt
kraj

Ispis:

carape
39: 3
42: 1
kombinezon
L: 1
M: 1
majica
M: 2

PREZIME I IME:

JMBAG:

BODOVI: