



CS203 - DOMAĆI ZADATAK 08.

Prilikom slanja domaćeg zadatka svom asistentu neophodno je da ispunite sledeće:

- Subject mail-a mora biti CS203-DZbr. Za ovaj domaći zadatak - CS203-DZ08
- Sve fajlove, koji su deo rešenja zadataka, arhivirati (zip)
- **Obavezno kopirati tekst zadatka kao komentar u kodu resenja**
- U prilogu maila treba da se nalazi arhiva projekta koji se ocenjuje, imenovana na sledeći način: CS203-DZbr-BrojIndeksa-ImePrezime.

Na primer, CS203-DZ08-9999-VeljkoGrkovic.zip

Molimo sve studente da se pridržavaju navedenog, inače zadaci neće biti pregledani i ocenjeni.

Studenti iz Beograda zadatke na ocenjivanje šalju mail-om na adresu lazar.mrkela@metropolitan.ac.rs

Studenti iz Nisa zadatke na ocenjivanje šalju mail-om na adresu

- IT smer: jovana.jovanovic@metropolitan.ac.rs
- SI smer: uros.lazarevic@metropolitan.ac.rs

Online studenti zadatke na ocenjivanje šalju mail-om na adresu uros.lazarevic@metropolitan.ac.rs **Rok za slanje domaćih zadataka:**

- **Tradicionalni studenti:** sedam dana od dana održavanja vežbe na koju se odnosi domaći zadatak. Nakon toga, broj poena se umanjuje za 50%. Krajnji rok za slanje rešenja je deset dana pre termina ispita u kome želite da polazete ispit.
- **Studenti na online nastavi,** kao i studenti kojima je predmet prenet u narednu godinu ili je diferencijalni: najkasnije deset dana pre termina ispita u kome želite da polazete ispit.

Svaki student radi jedan zadatak sa spiska. Ostali zadaci mogu da posluže za vežbanje i pripremu ispita, ali ih ne šalžete na pregled asistentima. Redni broj zadatak dobijate tako što

vaš broj indeksa podelite sa 12, a dobijeni ostatak pri deljenju uvećate za 1. Na primer: **Broj indeksa 2378**

$2378 \% 12 + 1 = 3$ – student radi 3. zadatak.

1. Napisati program koji učitava niz karaktera u jednoj liniji (dok se ne pređe u novi red), a korišćenjem steka ispisuje karaktere u inverznom poretku. Koristiti jednostruko povezanu listu za implementaciju steka.
2. Napisati program koji učitava niz celih brojeva u rastućem poretku i korišćenjem steka kao jednostruko povezane liste ispisuje brojeve u opadajućem poretku. Unos brojeva prekinuti kada se naruši rastući poredak tj. kada se desi da je sledeći uneti broj manji od prethodnog.
3. Neka je Q neprazan red, a S prazan stek. Napisati program i algoritam koji korišćenjem steka S vrši inverziju članova reda Q.
4. Napisati program i metodu za implementaciju reda pomoću niza, korišćenjem dve pomoćne promenljive front i rear, takve da ukoliko rear dostigne kraj niza, ceo red se pomera ka početku, tj. dok se front ne pomeri do indeksa 0.
5. Napisati program i metod za implementaciju deka korišćenjem kružnog niza. Generisati klase Deque i Queue takve da je klasa Deque izvedena iz klase Queue.
6. Napisati program za rad sa stekovima i redovima korišćenjem povezanih listi (definirati klasu za stek i red i odgovarajuće metode). Napisati program i metode koje:
 - a. pomeraju sve članove steka u red.
 - b. pomeraju sve članove reda u stek.
7. Napisati program za rad sa stekovima korišćenjem povezanih listi (definirati klasu za stek i odgovarajuće metode). Napisati program i metode koje:
 - a. prebacuju podatke iz jednog steka u drugi stek tako da podaci ostanu u istom redosledu.
 - b. prebacuju podatke iz jednog steka u drugi stek tako da podaci ostanu u inverznom redosledu.
8. Napisati program za rad sa stekovima i redovima korišćenjem povezanih listi (definirati klasu za stek i red i odgovarajuće metode). Napisati program i metode koje:
 - a. korišćenjem lokalnog steka invertuju redosled članova u redu.
 - b. korišćenjem lokalnog reda invertuju redosled članova u steku.

9. Napisati metodu koja će iz steka izbaciti sve one brojeve koji se pojavljuju dva ili više puta.
10. Napisati metodu koja će iz steka izbaciti sva pojavljivanja broja n. Metoda mora u novi stek, koji je inicijalno prazan, ubaciti brojeve koji predstavljaju koliko je broj bio udaljen od originalnog vrha steka.
11. Napisati metodu koja će na osnovu ulaznih stekova stek1 i stek2 stvoriti novi stek koristeći pritom sledeće pravilo da se za novi element u novom steku uvek odabere manji od elemenata na vrhu stekova stek1 i stek2. Ukoliko se jedan od stekova isprazni, onda uzeti preostale elemente iz drugog. Ulazni stekovi moraju nakon završetka metode ostati nepromenjeni.
12. Zadane su metode za stavljanje i skidanje elemenata sa stek i za stavljanje i skidanje elemenata iz reda (realizovanih listom):
dodajNaStek, skiniSaSteka, dodajURed, skinilzReda
a. Implementirati gore navedene metode.
b. Napišite rekurzivnu metodu, koristeći gore navedene metode, koja prebacuje elemente sa steka u red. Početni stek mora ostati očuvan. Redosled elemenata u redu isti je kao na steku (onaj element koji prvi izlazi sa steka je prvi element koji izlazi i iz reda). Red je na početku prazan.
13. U kontekstu strukture/generičke klase stek napišite metod insert_second koji ubacuje element x u stek s, tako da x bude drugi element odozdo (tj. drugi element od dna steka). Međusobni redosled ostalih elemenata ostaje nepromenjen.
14. Napišite ne-rekurzivnu metodu **public int SearchQ(Queue q);** koji pretražuje red od početka do kraja i pritom izbacuje duplikate elemenata koji su se već pre pojavili u redu. Uz to, broji koliko puta se koji element pojavio u redu i vraća najveći takav broj. Međusobni redosled ostalih (ne-izbačenih) elemenata ostaje nepromenjen.
15. Dane su metode za stavljanje i skidanje elemenata iz reda realizovanog listom:
a. dodajURed, skinilzReda
b. Napišite metodu, koristeći gore navedene metode, koja stvara novi red koji sadrži samo negativne elemente iz zadanog reda. Početni red mora ostati očuvan. Možete koristiti pomoćni red.
16. Posmatrajmo implementaciju "dinamičkog" steka pomoću niza promenljive dužine. Stek sadrži celobrojne podatke. Potrebno je napisati metode za stavljanje

elementa na stek i skidanje elementa sa steka. Ako na steku nema mesta za novi element, stek treba dinamički povećati za 10 novih mesta i upisati novi element. Takođe, kod skidanja elementa sa steka, potrebno je proveriti koliko ima praznih mesta. Ako ima više od 10 praznih mesta, veličinu steka potrebno je smanjiti za 10. Metode trebaju vratiti 1 ako je operacija uspeła, a 0 inače. Napišite i deklaraciju tipa podatka Stack.

17. U kontekstu strukture stack napraviti genericki metod: **void insert_N (T x, Stack s, int n);** koji ubacuje element x u stek s, tako da x bude n-ti element odozdo (tj. n-ti element od dna steka). Međusobni redosled ostalih elemenata ostaje nepromenjen.
18. Dat je stek S u kojem se čuvaju podaci tipa char. Na žalost, neki od elemenata steka su specijalni karakteri (npr. *). Napišite ne-rekurzivnu metodu sa prototipom **void ukoni_spec (Stack s);** koja će iz steka S ukloniti sve zvezdice, a ostale elemente poređati u "naopakom" poretku. Na primer, ako su elementi u S redom od vrha prema dnu steka (*, D, *, C, B, A), onda nakon poziva metode stek treba izgledati ovako: (A, B, C, D).