1. Kreirati klase Datum, Posao i Radnik. Klasa datum ima tri cjelobrojna privatna polja dan, mjesec i godina. Kreirati konstruktore, uključujući i konstruktor kopiranja i odgovarajuće pristupne metode. U konstruktoru provjeravati da li je datum ispravan. Obratiti pažnju na prestupne godine (one koje su djeljive sa 4 a nisu sa 100, a jesu sa 400. Npr. 2000, 1996 i 1892 su prestupne a 1900, 1800 i 2007 nisu prestupne). Klasa Posao ima tri privatna polja: nazivPoslodavca (tipa String), datumZaposlenja (tipa Datum) i datumPrestankaZaposlenja (tipa Datum). Klasa Radnik ima polja maticniBroj (tipa String), ime (string dužine ne veće od 30), staz (broj godina staža), plata (pozitivan realan broj) i zaposlenja (niz objekata klase Posao). Sva ova polja su privatna. Kreirati konstruktore i pristupne metode. Klasa Datum implementira interfejs Comaparable.

Kreirati klasu DblLista koja predstavlja **dvostruko olančanu listu** objekata klase Radnik. Lista se zadaje referencom na prvi čvor. Za predstavljanje čvora liste koristiti klasu Node. Napisati konstruktore i pristupne metode za klasu DblLista. Napisati metod za dodavanje radnika u listu, na početak liste. Napisati metod koji štampa sadržaj liste. Napisati metod koji iz liste uklanja radnika sa datim matičnim brojem. Napisati metod koji za dati datum d u datoteku štampa imena i matične brojeve svih radnika koji su datuma d bili u radnom odnosu.

1. Napisati klase Vozac, Cvor, ListaVozaca i TestListe. Klasa Vozac opisuje vozače formule 1 i ima polja ime i tim (oba su String) i pobjede i trening (nizovi cijelih brojeva, broj bodova ostvarenih na 20 trka i pozicije ostvarene na 20 treninga). Napisati konstruktore i odgovarajuće pristupne metode. Klasa Vozac implementira interfejs Comparable, na sljedeci nacin: vozač a je „veći“ od vozača b ako a ima više bodova na trkama. Ako imaju jednak broj bodova, tada se posmatra broj prvih pozicija na treninzima, a ako je i to jednako, posmatra se abecedni poredak. **Jednostruko olančana uređena lista** vozača opisuje se pomoću klase ListaVozaca i ima polja front i rear koji su tipa Cvor i polje brojElemenata, koje je tipa int. Polje front je referenca na prvi čvor liste, polje rear je referenca na posljednji element liste, dok polje brojElemenata sadrži broj elemenata u listi. Klasa Cvor predstavlja jedan čvor liste i ima dva polja: info (tipa Vozac) i next (referenca na sljedeći čvor liste). U klasi ListaVozaca napisati sljedeće metode.
2. Napisati metod dodaj(Vozac v) koja u listu dodaje vozača opisanog zapisom v tako da lista ostane uređena.
3. Napisati metod Brisi(String ime) koja iz liste l briše čvor koji sadrži vozača čije je ime dato argumentom ime.
4. Napisati metod Stampaj() koja na standardni izlaz štampa imena vozača i broj pobjeda svakog od njih.
5. Napisati metod Stampaj(String fName, int x) koja u datoteku čije je ime dato argumentom fName štampa imena i broj pobjeda svakog vozača koji je osvojio bar x bodova.

U metodu main klase TestListe kreirati listu sa bar 10 vozača. Podatke o vozačima učitati iz tekstualne datoteke čije ime unosi korisnik. Datoteka je organizovana na sljedeći način: svaki vozač zauzima 4 uzastopna reda; u prvom redu je ime vozača, u drugom redu je tim, a treći i četvrti red sadrže po 20 cijelih brojeva razdvojenih jednim blankom, redom bodove i pozicije na treninzima.

1. Kreirati klasu Rijeci koja implementira jednostruko-olančanu uređenu listu u kojoj se čuvaju stringovi uređeni u leksikografskom poretku. Jedini član te klase je promjenljiva glava tipa Node. Klasa Node je ugnježdena privatna klasa u klasi Rijeci i ima sljedeće članove: data – tipa String i next – tipa Node (referenca na sljedeći čvor liste). Napisati konstruktor za klasu Node. U klasi Rijeci, napisati sljedeće metode: metod printList() koji štampa elemente liste, metod insert(String d) – koji u listu unosi string d, tako da lista ostane uređena, ako on već ne postoji u listi (ako postoji ne unositi ga) i metod delete(String d, int k) koji iz liste briše sve stringove koji razlikuju od stringa d na više od k karaktera. Kreirati klasu Test, sa metodom main. U main metodi kreirati jednu ovakvu listu sa barem 5 elemenata i pozvati metode insert, delete i printList.
2. Kreirati klasu MyList koja implementira jednostruko-olančanu uređenu listu u kojoj se čuvaju cijeli brojevi u rastućem poretku. Jedini član te klase je promjenljiva head tipa Node. Klasa Node je ugnježdena privatna klasa u klasi MyList i ima sljedeće članove: data – tipa int i next – tipa Node (referenca na sljedeći čvor liste). Napisati konstruktor za klasu Node. U klasi MyList, napisati sljedeće metode: metod printList() koji štampa elemente liste, metod insert(int d) – koji u listu unosi broj d, tako da lista ostane uređena, ako on već ne postoji u listi (ako postoji ne unositi ga), metod delete(int d, int k) koji iz liste briše sve čvorove koji se čija je apsolutna razlika sa brojem d manja od k i metod groupReverse(int k) koja okreće listu u grupama od po k elemenata k≥1. Na primjer, ako je data lista čiji su elementi:

2 4 6 8 10

nakon poziva funkcije groupReverse za k=3, lista će izgledati ovako:

6 4 2 8 10

Kreirati klasu Test, sa metodom main. U main metodi kreirati jednu ovakvu listu sa barem 5 elemenata i pozvati metode insert, delete, groupReverse i printList.

1. Kreirati klasu Paralelogram koja ima dva četvočlana niza realnih brojeva x i y koji predstavljaju redom koordinate tjemena paralelograma u smjeru kretanja kazaljke sata, tj. (x[i],y[i]) je koordinata i-tog tjemena. Napisati konstruktor sa dva argumenta. Klasa Paralelogram treba da implementira interfejs Comparable tako da manjim paralelogramom smatramo onaj koji je manje površine, a ako su površine jednake, tada je manji onaj čiji je obim manji. Redefinisati metod toString tako u jednom redu ispisuje koordinate sva 4 tjemena. Na primjer: (0,0)-(0,1)-(1,1)-(1,0).

Kreirati klasu ListaParalelograma koja implementira **dvostruko olančanu uređenu listu** objekata klase Paralelogram. Lista se zadaje referencom na prvi čvor. Napisati metod insert(Paralelogram p) za dodavanje paralelograma u listu tako da ona ostane u rastućem poretku. Napisati metod print() koji štampa sadržaj liste. Napisati metod delete(double a, double b) koji iz liste uklanja sve paralelograme čija je površina veća od a i manja od b.

Kreirati klasu Test, sa metodom main. U metodi main kreirati listu, unijeti bar 5 objekata klase Paralelogram u tu listu i pozvati metode delete i print.

1. Kreirati klasu Crtac1 izvedenu iz klase JPanel. Implementirati metod koji na klik miša crta ili krug ili kvadrat ili zaobljeni kvadrat sa centrom u tački u kojoj je kliknuto. Metod treba slučajno da izabere figuru, nacrta je i popuni plavom bojom. Stranica kvadrata i poluprečnik kruga treba da budu slučajan cio broj iz intervala [15,50]. Parametri koji određuju zaobljenost kvadrata jednaki su četvrtini stranice kvadrata.
2. Kreirati klasu Crtac2 izvedenu iz klase JPanel. Napisati metod koji prikazuje izborom prve dve tačke crta pravu, a zatim svakim sljedećim izborom tačke crta trougao sačinjen od prve dvije izabrane tačke i posljednje izabrane tačke.
3. Kreirati klasu DatumGenerator izvedenu iz klase JPanel koja sadrži jedno read-only tekstualno polje Datum i dugme Next. U polju Datum na početku je upisana vrijednost “14.04.2018.”. Klikom na dugme Next u polju Datum potrebno je upisati datum sljedećeg dana. Obavezno napisati klasu u kojoj se kreira prozor i iz koje se pokreće Java aplikacija.
4. Kreirati klasu NumberConverter izvedenu iz klase JPanel koja sadrži dva tekstualna polja Number1 i Number2, dugme Convert i dva radio button-a HexToDecimal i DecimalToHex. Klikom na dugme Convert, broj upisan u polju Number1 se pretvara iz heksadecimalnog u dekadni brojni sistem (ukoliko je radio button HexToDecimal selektovan) ili iz dekadnog u heksadecimalni brojni sistem (ukoliko je radio button DecimalToHex selektovan). Rezultat se upisuje u polje Number2. Obavezno napisati klasu u kojoj se kreira prozor i iz koje se pokreće Java aplikacija.
5. Marko je planirao da ide u Rusiju na Svjetsko prvenstvo u fudbalu. Kreirajte klasu WorldCup koja ima izgled kao na slici kako bi pomogli Marku da izračuna koliko novca će mu biti potrebno za ovo putovanje. Marko može da bira gradove koje će posjetiti, da li će putovati ekonomskom ili biznis klasom, kao i da li da pored avionskih karti rezerviše i smještaj u hotelu. Spisak svih gradova domaćina nalazi se u fajlu “russia.txt”. U jednom redu tog fajla su podaci o jednom gradu naziv grada i tri realna broja koja redom predstavljaju cijenu putovanja biznis klasom do tog grada, cijenu putovanja ekonomskom klasom do tog grada i cijenu hotelskog smještaja u tom gradu. Klikom na dugme Submit, u labelu Total se upisuje ukupna cijena putovanja. Za dati izgled možete koristiti četiri panela i BorderLayout.



1. Kreirati klasu PopProzor koja ima izgled kao na slici. Dimenzije prozora su 350x100. Polja Broj i Rezultat su veličine 80x25, dok su polja Pocetak i Kraj veličine 40x25. Za dati izgled možete koristiti FlowLayout. U polje Broj možete da upišete string s koji sa sastoji samo od cifara 0-9, bez vodećih nula, dužine ne veće od 1000. U polja Pocetak i Kraj upisujete 2 pozitivna cijela broja p i q, takva da važi 0<=p<=q<s.length(). Klikom na dugme Izracunaj, u polje Rezultat upisuje se broj koji se dobije kada se sve cifre u stringu s od pozicije p do pozicije q (uključujući i ove dvije pozicije) zamijene zbirom cifara s[p]+...+s[q]. Nije potrebno kontrolisati ispravnost ulaza.



1. Kreirati klasu TriangleDrawer izvedenu iz klase JPanel koja sadrži jedno dugme Load i jedan panel. Koristeći funkciju setPreferredSize postaviti dimenzije unutrašnjeg panela na 400x400. Klikom na dugme Load iz fajla “trouglovi.txt” čija svaka linija sadrži po 6 cijelih brojeva u sljedećem obliku x1,y1,x2,y2,x3,y3 učitati listu trouglova, a zatim ih iscrtati na unutrašnjem panelu. Svaki trougao ispuniti bojom koja je generisana na slučajan način. Izgled panela posle iscrtavanja treba da bude kao na slici. Možete koristiti BorderedLayout.



1. Kreirati klasu Prozor koja ima izgled kao na slici. Dimenzije prozora su 400x500. Za dati izgled možete koristiti tri panela i GridLayout. U polje Fajl možete da upišete string s koji je ime fajla. U polje prosjek Prosjek upisujete pozitivan cio broj p iz intervala [0,100]. Klikom na dugme Ucitaj, u listu se upisuju svi studenti iz datoteke čije je ime string s koji imaju prosjek bodova veći od p. Klikom na dugme Sortiraj, na standarni izlaz štampaju se svi elementi iz liste poređani po brojevima indeksa (prvo idu studenti sa većom godinom upisa, a ako je godina upisa ista, prvo idu studenti sa manjim indeksom). Podaci o jednom studentu u fajlu nalaze se u 3 reda: prvi red sadrži broj indeksa i godinu upisa, drugi red ime i prezime studenta a treći red listu bodova osvojenu na testovima (4 cijela broja iz intervala [0,100]).



1. Kreirati klasu ConvexWindow koja omogućava da se lijevim klikom miša unutar panela (dio prozora koji je bijele boje) definišu tačke ravni, koje se smještaju u niz. Tačke crtati kao krugove čiji je poluprečnik 3. Klikom na dugme Racunaj, treba generisati konveksni omotač datog skupa tačaka i nacrtati ga u datom panelu. Ispod su data dva algoritma za traženje konveksnog omotača.

algoritam Grejema: <http://en.wikipedia.org/wiki/Graham_scan>

algoritam monotonih lanaca (algoritam Endrjua): <http://in1.csie.ncu.edu.tw/~jrjiang/alg2007/Andrew%27s%20Monotone%20Chain%20Convex%20Hull%20Algorithm.ppt>

