

Primer pitanja za obavezni zadatak na ispitu iz Objektno orijentisanog programiranja (2014)

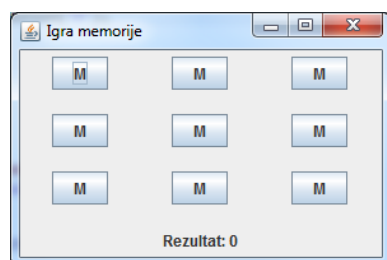
1. Napisati klasu Osoba koja sadrži privatne attribute ime i prezime (stringovi) i konstruktor kojim se inicijalizuju ti atributi. Napisati samo getere za oba atributa. Napisati klasu Student koja nasleđuje klasu Osoba i poseduje String atribut brojIndeksa. U klasi Student napisati konstruktor kome se prosleđuju ime, prezime i broj indeksa i kojim se inicijalizuju odgovarajući atributi.
2. Napisati apstraktnu klasu Tim koja sadrži privatni atribut naziv i apstraktnu metodu int odigrajUtakmicu(Tim drugiTim). Napisati klasu KosarkaskiTim koja nasleđuje klasu Tim u kojoj treba implementirati metod odigrajUtakmicu tako da vraća -1, ukoliko prosleđeni drugiTim nije KosarkaskiTim, odnosno 1 ili 2 (bira se na slučajan način), ukoliko je pobedio jedan ili drugi tim.
3. Napisati apstraktnu klasu Auto koja sadrži jedan privatni atribut **naziv**, koji se prosleđuje kroz konstruktor (može postojati samo jedan konstruktor koji dobija String kao parametar). Napisati klasu Audi koja nasleđuje klasu Auto i koja u svom konstruktoru inicijalizuje atribut naziv iz natklase na „Audi“.
4. Napisati interfejs Radnik koji poseduje jedan metod boolean radi(). Napisati klasu Programer koja implementira interfejs Radnik i koja poseduje statički int atribut posao (inicijalno jednak 10). U toj klasi implementirati metod boolean radi() koji umanjuje vrednost atributa posao za 1, ukoliko je ta vrednost veća od 0 i tom prilikom vraća true, odnosno vraća false, ukoliko je vrednost tog atributa 0.
5. Napisati klasu Linija i u njoj dva atributa klase **Tacka**, koja se inicijalizuju kroz konstruktor, kome se prosleđuju 4 int parametra. U toj klasi napisati privatnu unutrašnju klasu Tacka koja poseduje dva javna numerička atributa x i y (koordinate), odgovarajući konstruktor, kao i metod double rastojanje(Tacka drugaTacka) koja vraća rastojanje do druge tačke.
6. Napisati klasu Student koja implementira Comparable interfejs i sadrži dva atributa – ime (String) i prosek (double). Napisati konstruktor kojim se inicijalizuju ti atributi. Implementirati compareTo metod tako da vrši poređenje po proseku.
7. Napisati klasu Student koja poseduje tri privatna atributa – ime (String), brIndeksa (int) i prosek (double). Napisati podrazmevani konstruktor, zatim konstruktor kojem se prosleđuju i inicijalizuju ime i brIndeksa, kao i konstruktor kome se prosleđuju ime, brIndeksa i prosek, pri čemu taj konstruktor prva dva parametra mora inicijalizovati pozivom drugog konstruktora.
8. Napisati klasu Matematika koja poseduje jednu javnu konstantu PI (double) = 3,14 i poseduje dve statičke metode – površinaKrug(double r) i obimKrug(double r) u kojima je potrebno izračunati površinu, odnosno obim kruga, za prosleđeni poluprečnik (r) i konstantu PI.
9. Napisati enum Komanda, sa mogućim vrednostima SABIRANJE i DELJENJE. Napisati klasu Matematika koja poseduje jednu statičku metodu double operacija(double a, double b, Komanda k), koja za prosleđenu komandu vrši odgovarajuću operaciju nad promenljivama a i b i vraća odgovarajući rezultat (voditi računa kod deljenja da ne dođe do izuzetka).
10. Napisati enum Dan sa vrednostima dana u nedelji. Napraviti klasu Trening koja sadrži privatne attribute sport (String) i listu dani (ArrayList<Dan>) u kojoj se čuvaju podaci o tome

- kojim danima se trenira taj sport. Napisati konstruktor kojim se inicijalizuje atribut sport i metod dodajDan(Dan dan) kojim se u listu dodaje novi dan, pod uslovom da taj dan nije dodat.
11. Napisati enum Mesec sa vrednostima meseci u godini. Napisati klasu Datum koja sadrži tri atributa – dan (int), mesec (Mesec) i godina (int). Napisati konstruktor kojim se inicijalizuju ova tri atributa.
 12. Napisati klasu Loto, koja sadrži jedan statički metod ArrayList<Integer> brojevi(), koji vraća 7 različitih slučajnih brojeva u intervalu od 1 do 33.
 13. Napisati klasu Auto, koja sadrži dva String atributa – marka i model. Napisati konstruktor pomoću kojeg se inicijalizuju ta dva atributa. Napisati main metod, pomoću kojeg se sa standardnog ulaza učitava jedan automobil.
 14. Napraviti klasu Brojac koja ima:
 - a. metodu add koja ne sadrži povratni tip, već samo uvećava vrednost brojača
 - b. metod count koji vraća tekuću vrednost brojača
 - c. metod toString
 - d. svi objekti klase Brojac imaju ISTU vrednost brojača
 15. Napraviti klasu Skola koja se ne može naslediti. Klasa sadrži polja direktor (String), dinamički niz studenata (pretpostaviti da postoji klasa Student) i dinamički niz profesora (pretpostaviti da postoji klasa Profesor). Polja inicijalizovati preko konstruktora (prosleđuju mu se cele liste i String koji označava profesora).
 16. Implementirati panel na kome je plavom bojom, na poziciji (100, 50), napisan tekst „RAF“. Zadatak obavezno rešiti upotrebom 2D grafike.
 17. Implementirati panel na kome je nacrtan kvadrat, stranice 20, čiji je gornji levi ugao na poziciji (30, 30).
 18. Implementirati panel na kome je nacrtan kvadrat, na poziciji (20, 20) (gornji levi ugao), stranice 30 piksela, čije su ivice plave boje. Potrebno je nacrtati dijagonalu kvadrata zelene boje, kao i krug, koji je upisan u kvadrat i koji je popunjen žutom bojom. Nije potrebno pisati main metod.
 19. Implementirati panel na kome je nacrtana elipsa, širine 50, a visine 40 piksela, sa centrom u tački (100, 100). Elipsa mora biti cela obojena u crvenu boju. Potom je potrebno u elipsi napisati plavim bojom tekst „RAF“. Nije potrebno pisati main metod.
 20. Implementirati panel, na kome je nacrtan plavom bojom pravougli trougao, proizvoljnih dimenzija stranica. Ispod trougla ispisati tekst „Pravougli trougao“ žutom bojom. Nije potrebno pisati main metod.
 21. Implementirati prozor, dimenzija 200x150 piksela, sa naslovom „Prozor“, čijim zatvaranjem se završava rad celog programa. Implementirati main metod u kojem se prikazuje kreirani prozor.

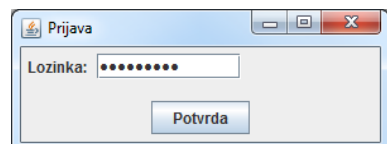
22. Implementirati prozor (bez main metoda) tako da ima izgled kao na slici (obratiti pažnju da dugmići nisu razvučeni po celoj širini prozora).



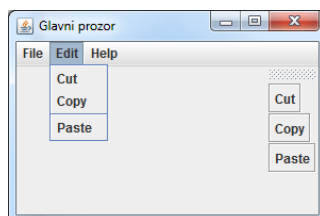
23. Implementirati prozor (bez main metoda) tako da ima izgled kao na slici.



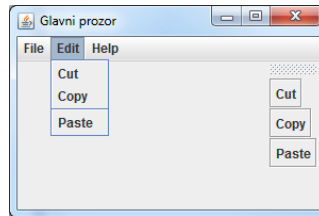
24. Implementirati prozor (bez main metoda) tako da ima izgled kao na slici.



25. Implementirati prozor (bez main metoda) tako da sadrži samo implementiran glavni meni (bez palete alata) kao na slici ispod.



26. Implementirati prozor (bez main metoda) tako da sadrži samo implementiranu paletu alata na istočnom delu prozora (bez glavnog menija) kao na slici ispod.



27. Implementirati prozor na kojem se nalazi jedno dugme, na koje kada se klikne, boja panela se boji u plavo.
28. Implementirati prozor na kome se nalaze dva radio dugmeta („Crveno“ i „Plavo“), koja su međusobno isključiva. Korisnik kada odabere prvo dugme, na standardnom izlazu se ispiše poruka „Crveno“, a kada odabere drugu opciju, na standardnom izlazu se ispiše tekst „Plavo“.
29. Implementirati prozor na kome se nalazi „combo-box“, sa opcijama „Crveno“ i „Plavo“. Izborom odgovarajuće opcije, ispisuje se odgovarajući tekst na standardni izlaz.
30. Implementirati prozor na kojem se nalazi jedna labela (JLabel) koja prikazuje trenutnu poziciju miša na panelu, tj. ta pozicija se ažurira kako korisnik pomera pokazivač miša.
31. Implementirati prozor na kojem se pritiskom na taster „D“ na tastaturi ispisuje na standardnom izlazu poruka „Pritisnuto je slovo D.“.
32. Implementirati prozor na kojem se nalazi jedno polje za unos teksta (Text Field) i jedno dugme – „Potvrda“. Klikom na to dugme na standardni izlaz treba ispisati uneti tekst iz polja.
33. Implementirati prozor na kojem se nalazi jedno polje za unos lozinke i jedno dugme – „Potvrda“. Klikom na dugme potvrda, ukoliko je korisnik uneo lozinku „Admin“, na standardni izlaz ispisati poruku „Uspešno ste se prijavili“, a ukoliko je uneo bilo koju drugu lozinku, na standardni izlaz ispisati tekst „Neuspešna prijava“.
34. Implementirati prozor na kojem se nalazi ispisan tekst „Računarski fakultet“. Ispod teksta nalazi se jedan „checkbox“ sa tekстом „Bold“ koji je inicijalno selektovan. Promenom stanja tog checkbox-a, tekst se bolduje, odnosno vraća u normalan izgled.
35. Implementirati prozor na kojem se nalazi glavni meni sa jednom opcijom „Boja“ koja ima podopcije „Plava“ i „Crvena“. Izborom odgovarajuće opcije, na standardnom izlazu se ispisuje odgovarajući tekst („Crveno“ ili „Plavo“).
36. U tekstualnoj datoteci studenti.txt se nalaze podaci o studentima – jedan red odgovara jednom studentu. Napisati main metod u kojem se iz te datoteke učitavaju svi redovi i ispisuju na standardni izlaz.
37. Napisati main metod u kojem je potrebno u tekstualnu datoteku „studenti.txt“ uneti sledeći tekst:

```
Petar|Petrovic|10
Nikola|Nikolic|9
```

38. Implementirati izuzetak `DeljenjeException`, koji nasleđuje `IOException`. Napisati klasu `Matematika` i u njoj jedan metod `double podeli(double br1, double br2)`. Metod bi trebalo da vraća količnik brojeva `br1` i `br2`, odnosno da „ispali“ instancu klase `DeljenjeException`, ukoliko je drugi broj jednak 0.
39. Klasa `Matematika` poseduje jedan statički metod `podeli(double br1, double br2)` koji može dovesti do `DeljenjeException`-a, koji nasleđuje `IOException`. Napisati main metod u kojem je potrebno samo pozvati taj metod sa proizvoljnim argumentima i uz pomoć `try catch` bloka hvatati `DeljenjeException`.