**TEORIJA**  
  
**1. Što je formalna verifikacija (precizno odgovoriti, ne matematičko-logičkim simbolima)?**

To je postupak provjere da formalni model dijela izvedenog sustava (/), odgovara formalnoj specifikaciji (*S*) sa   
 matematičkom izvjesnošću ("odgovara" = logički zadovoljava).

**2. Navedi slijedom sve temeljne aktivnosti u modelu RUP.**  
 Početak (inception), Elaboracija (elaboration), Izgradnja produkta (construction),   
 Prijenos produkta korisnicima (transition)

**3. Objasni razliku između proširenog obrasca *extend* i uključenog obrasca *include*.**

*Vezom uključivanja* se povezuju dva obrasca uporabe na način da jedan obrazac u tijeku svog

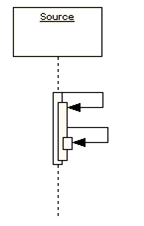
izvođenja u potpunosti izvede uključeni obrazac uporabe pri čemu trenutak izvođenja uključenog

obrasca nije specificiran dijagramom.

*Vezom proširenja* se povezuju dva obrasca uporabe pri čemu jedan proširuje funkcionalnost

drugog. Proširenje se ostvaruje ukoliko je zadovoljen određeni uvjet definiran u točki proširenja.

**4. UML sekvencijski dijagram je oblik dijagrama interakcije između objekata. Što na dijagramu   
 predstavljaju ovi primjeri petlji?**

Višestruko pozivanje internih procedura

**5. Oznaka na strelici poruke u UML sekvencijskom dijagramu je [x>2]2:m: = daj\_lokaciju(slike).   
 Objasni što znači pojedini dio oznake.** [x>2] – uvjet, 2:m: - parametri koji se vraćaju, daj\_lokaciju – naziv, (slike) – parametar

**6. Princip ugovornog oblikovanja (*design by contract*) postavlja neke zahtjeve na pozivanu proceduru.   
 Navedi te zahtjeve.**

*Preduvjeti* koje pozvana metoda traži da budu ispunjeni prije početka izvođenja (engl. *preconditions*).

*Uvjeti* koje pozvana metoda mora osigurati *nakon* završetka izvođenja (engl. *postconditions*).

*Invarijante* na koje pozvana metoda neće djelovati (mijenjati) pri izvođenju.

**7. Navedi generičke aktivnosti u procesima programskog inženjerstva.**  Specifikacija, razvoj i oblikovanje, validacija i verifikacija, evolucija

**8. Navedi barem 2 značajke kakvoće programskog produkta.**

Prihvatljivost, pouzdanost, održavanje

**9. Navedi 3 problema evolucijskog modela razvoja i oblikovanja PP.**

Proces razvoja nije jasno vidljiv, sustavi su loše strukturirani, često potrebne posebne vještine

**10. Oblikovanje PP obuhvaća akciju inženjerstva zahtjeva. Navedi sve načine na koji se mogu izraziti zahtjevi.**

Strukturiranim prirodnim jezikom, specifičnim jezikom za oblikovanje (SDL), grafičkom notacijom (ERA, UML) i

matematičkom specifikacijom (vremenska logika)

**11. Kako se u sekvencijskom dijagramu označava petlja?**

Na strelicu se napiše \*[uvjet]povr:=naziv(parametri) gdje zvjezdica označava petlju

**12. Ako su istinite formule?**

**(1) ¬P**

**(2) Q**

**(3) (P ili ¬Q ili R)**

**Općim postupkom dokazivanja teorema (opovrgavanje, obaranje) pokaži da li je R logička posljedica   
 navedenih formula.**

- R logička posljedica skupa {(1), (2), (3)} ako je svaki model tog skupa ujedno i model formule R

- model formule R je bilo koji u kojem je R=T

- iz (1) P=F, iz (2) Q=T, iz (3) (P v ¬Q v R) ako uvrstimo P i Q=>F v F v R=> da bi bilo točno, mora R biti T

-dakle, jedini model za skup {(1), (2), (3)} je: P=F, Q=T, R=T, što je i model za R, konačno, R je logička posljedica

skupa {(1), (2), (3)}

**13. Zaokruži neispravne formule CTL logike:**

a) A[pUEF¬r]

**b)** AEFr

**c)** FGr

**d)** AF[(rUg) i (pUr)]

e) A[¬pUA[qUr]]

**f)** A¬G¬p

**14. U okviru procesa ispitivanja PP postoje faze 1-funkcijski test, 2-test performansi, 3-test instalacije, 4-test   
 uporabe, 5-test komponenti, 6-test prihvatljivosti, 7-test integracije. Poredajte faze po vremenskom slijedu.**

5-7-1-2-6-3-4

**15. U arhitekturi protoka podataka pažnja je usredotočena na prijenos podataka između aktora. Upravljački tok je   
 implicitan. Navedi barem 3 mehanizma upravljanja u protoku podataka.**

Guranje (PUSH), povlačenje (PULL), guranje/povlačenje (PUSH/PULL), pasivni

**16. U arhitekturi PP koja se zasniva na repozitoriju podataka postoje 2 velike podskupine: baze podataka i oglasna   
 ploča. Navedi temeljnu razliku između ovih podskupina.**

Baza podataka – vanjski procesi iniciraju promjenu sadržaja  
 Oglasna ploča – promjena sadržaja inicira vanjske promjene

**17. Model programskog inženjerstva koji je najprikladniji kada zahtjevi u početku nisu potpuno definirani je:**

Evolucijski

**18. Obrasci uporabe koriste se pri modeliranju *funkcionalnih*/nefunkcionalnih zahtjeva (zaokruži).**

**19. Za razliku od sekvencijskog dijagrama kojem je u fokusu vremenska uređenost između događaja, u fokusu UML**

**kolaboracijsko/komunikacijskog je:**

Identifikacija sudionika komunikacije, tko s kime, sučelje

**20. Rezultat procesa IZ je:**

Specifikacija programskog produkta

**21. U iterativnom postupku razvoja određene arhitekture programske potpore postoje razredi: kocka...(još neki   
 geometrijski oblici) su spojeni u razred Oblik3D (predstavlja ih Oblik 3D s adaptacijom metoda). Princip   
 oblikovanja (ima ih 12) kojima se opisuju ovi postupci je:**

Povećanje kohezije

**22. U dijagramu** 1..3 2..5

**Crvena klasa instancira 3 objekta. Koliko maksimalno može objekata imati 2.klasa?**

Minimalno 2, maksimalno 15

**23. Namjena korištenja UML dijagrama?**

Ujedinjeni jezik za modeliranje (engl. Unified Modelling Language, kraće: UML) je normirani jezik opće namjene

koji se koristi za modeliranje računalnih sustava temeljenih na objektno-orijentiranoj paradigmi

**24. Program izgrađen u objektnoj arhitekturi je brži ako ima više/*manje* (zaokruži) dinamičkih povezivanja?**

**25. Dio nekog problema riješen je na sljedeći način: catpopis.txt|grupa z1|sort|analiza>rez.txt**

**Koji je to tip arhitekture?**

Protok podataka, cjevovodi i filtri

**26. Ako {Γ,L} kažemo da je konzistentan ako?**

Skup Γ je konzistentan akko (ako i samo ako) ne sadrži formule na temelju kojih bi ωi i ¬ωi istovremeno bili

Teoremi.

**27. Tko testira komponente prije integracije?**

Programer

**28. Korištenjem OCSF izgrađen je dio klijentske aplikacije, koji je nastao tako da je: naslijeđen razred   
 *AbstractClient* te je implementirana metoda: *handleMessageFromServer*. Pokaži fragmentom koda kako se   
 razred koristi ako bi se poslužitelju koji sluša na adresi 10.0.0.33:12345 poslala poruka sadržaja: „test“.**

**Na ispitu je bio popis razreda poslužitelja i klijenta.**

**ZADACI**

**1. Sljedeće rečenice napišite u predikatnoj logici:**

* 1. **Svaki student je mlađi od nekog nastavnika**
  2. **Svaki sin mog oca je moj brat.**

**2. Preslikaj u CTL:**

* + 1. **Iz svih dohvatljivih stanja gdje je p=istinito (dakle ima stanja u kojima je p neistinito) izvođenje programa će svakako završiti u stanju q=istinito.**

**b) Za svako stanje ako je u njemu p=istinito tada za svako stanje koje se iz njega može doseći u jednom   
 koraku dalje uvijek vrijedi da je q=neistino ili da je r=neistinito sve dok t ne postane posve istinit.**

**3. Napiši formalan (matematički-logičan) izraz za sljedeće izjave (standardno usvojen znak za „zadovoljava“ je|=)**

**a) Svako stanje S0 iz skupa Q0 u modelu M zadovoljava formulu vremenske logike φ.**

**b) Skup stanja za koja vrijedi Q(EX f) je skup stanja S takav da za svaki S postoji neko stanje t povezano s   
 relacijom R, te da je to stanje z ujedno u skupu stanja u kojima je formula vremenske logike f istinita.**

**4. Zadatak**

**a) Definiraj monotonost funkcije F: 2S** ⭢**2S nad skupom S (oznaka 2S označuje sve podskupove).**

**b) Za funkciju F: 2S** ⭢**2S definiranu izrazom F(X) = X {S1, S2} nađite njenu najmanju i najveću čvrstu točku   
 za S = {S0, S1, S2, S3}**

* 1. **Kako se izračunava najmanja, a kako najveća čvrsta točka monotonih funkcija nad skupovima sa n+1 članom.**

**5. Tijekom procesa strukturnog ispitivanja (white box) modula programske potpore generiran je graf tijeka**

**programa koji sadrži 9 čvorova i 14 lukova bez dodatnih povezanih komponenti. Koliko je potrebno testova da se**

**ispitaju svi temeljni putovi (staze) kroz program.**

**6. Skiciraj algoritam za izračunavanje skupa stanja Q(EG f) gdje je f bilo koja CTL formula.**

**7. Preslikaj 2 rečenice prirodnog jezika u ispravne formule logike predikata:**

**a) Ana voli svu Marijinu bradu.**

**b) Ana voli jednog Marijinog brata.**

**Koristi predikate voli(x,y), brat(x,y)**

**8. Napiši formulu CTL logike koja izražava: „Iz svakog stanja moguće je doći do početnog stanja.“**

**9. Tijekom procesa strukturnog ispitivanja (white box) modula PP generiran je graf tijeka programa koji sadrži 8**

**čvorova i 12 lukova bez dodatno povezanih komponenti. Koliko je najmanje potrebno testova da se ispitaju svi**

**temeljni putovi?**

**10. Zadana je funkcija double volumen\_kvadra(double a, double b, double c); Napiši 4 testna primjera.**

**11. Funkcija: int sort (int \*a, int \*b, int \*c);**

**Kod: if (!a||!b||!c)**

**return -1;**

**if (\*a>\*b)**

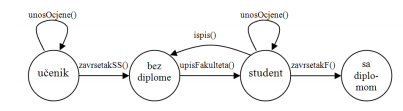
**zamijeni( , ) – tu je bilo nekoliko funkcija zamijeni, ali se ne sjećam što je bilo točno u pitanju...**

**return 0;**

**I sad je trebalo popuniti tablicu sa zaglavljem: broj|a|b|c i mislim da je trebalo popunjavati redoslijedom kako   
 se izvršavao program, označiti za pojedinu varijablu A ako se koristila njena adresa, ili V ako se koristila njena   
 vrijednost**

**12. Za razred *StatusOsobe* zadan je dijagram stanja prema slici, gdje su prijelazi uzrokovani pozivima metoda**

**razreda.**

****

1. **Kako oblikovati testni slučaj?**
2. **Navedite ga**
3. Tako da se prođu sva stanja i svi prijelazi
4. unosOcjene🡪zavrsetakSS🡪upisFakulteta🡪ispis🡪upisFakulteta🡪unosOcjene🡪zavrsetakF ili  
   unosOcjene🡪zavrsetakSS🡪upisFakulteta🡪unosOcjene🡪ispis🡪upisFakulteta🡪zavrsetakF

**13. Među životinjama nalazi se mačak koji jede sir. Zadano: životinja(x), mačak(x), jede\_sir(x)**

**14. Najveći miš se ne boji najmanje mačke. Zadano: miš(x), mačka(x), boji(x,y), najmanji(x), najveći(x)**

**15. Ako nema mačke, miševi mogu pojesti i cijeli kolut sira. Zadano: nema\_mačke, pojeden\_sir**

**16. Svjetlo može biti ili upaljeno ili ugašeno (ex ili). Zadano: upaljeno, ugašeno**

**17. Kripke struktura**

**a) AG(uči->AF(položio))**

**b) E(učiUpoložio)**

položio

polaže

uči