

## 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.

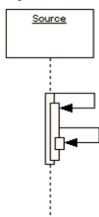
Zahtjevi, analiza, oblikovanje, implementacija, test

### 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 $[x > 2]2:m = \text{daj\_lokaciju}(\text{slike})$ .

Objasni što znači pojedini dio oznake.

$[x > 2]$  – uvjet, 2 – sekvencijski broj, m – povratna vrijednost, daj\_lokaciju – ime poruke, (slike) – argument poruke

### 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  $*[\text{uvjet}]\text{povr}:=\text{naziv}(\text{parametri})$  gdje zvjezdica označava petlju

**12. Ako su istinite formule?**

- (1)  $\neg P$
- (2)  $Q$
- (3)  $(P \text{ ili } \neg Q \text{ ili } R)$

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)  $(P \vee \neg Q \vee R)$  ako uvrstimo  $P$  i  $Q \Rightarrow F \vee F \vee R \Rightarrow$  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)  $A[pUEF\neg r]$
- b)  $AEFr$
- c)  $FGr$
- d)  $AF[(rUg) \text{ i } (pUr)]$
- e)  $A[\neg pUA[qUr]]$
- f)  $A\neg G\neg p$

**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

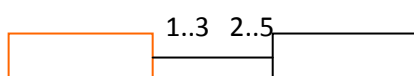
**20. Rezultat procesa IZ je:**

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**



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  $\{\Gamma, L\}$  kažemo da je konzistentan ako?

Skup  $\Gamma$  je konzistentan akko (ako i samo ako) ne sadrži formule na temelju kojih bi  $\omega_i$  i  $\neg\omega_i$  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.

29. Navedi 3 prednosti inkrementalnog nad spiralnim modelom

30. Navedi bar 2 razloga za smanjenje međuovisnosti komponenti

Zbog budućih promjena u produktu i zbog potencijalne buduće iskoristivosti samostalne komponente

31. Nasljeđivanje i agregacija, što se od toga treba programirati, a što ne?

## ZADACI

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

- a) Svaki student je mlađi od nekog nastavnika  $\exists v \leftrightarrow \Rightarrow \neg \forall \wedge$
- b) Svaki sin mog oca je moj brat.
- a)  $\forall x(\text{student}(x) \Rightarrow \exists y(\text{nastavnik}(y) \wedge \text{mlađi}(x,y)))$
- b)  $\forall x(\text{sin\_mog\_oca}(x) \Rightarrow \text{moj\_brat}(x))$

### 2. Preslikaj u CTL:

- a) 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  $q$ =neistinito ili da je  $r$ =neistinito sve dok  $t$  ne postane posve istinit.
- a)  $AF[p \Rightarrow EF(q)]$
- b)  $AG[p \Rightarrow AX(A((\neg q \vee \neg r) \cup t))]$

### 3. Napiši formalan (matematički-logičan) izraz za sljedeće izjave (standardno usvojen znak za „zadovoljava“ je $\models$ )

- a) Svako stanje  $S_0$  iz skupa  $Q_0$  u modelu  $M$  zadovoljava formulu vremenske logike  $\phi$ .
- 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.
- a)  $s \models M, S_0 \models \phi$
- b)  $s \models M, s \models$

### 4. Zadatak

- a) Definiraj monotonost funkcije  $F: 2^S \rightarrow 2^S$  nad skupom  $S$  (oznaka  $2^S$  označuje sve podskupove).
- b) Za funkciju  $F: 2^S \rightarrow 2^S$  definiranu izrazom  $F(X) = X \cup \{S_1, S_2\}$  nađite njenu najmanju i najveću čvrstu točku za  $S = \{S_0, S_1, S_2, S_3\}$
- c) 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.

$$CV(G) = \text{Lukovi} - \text{Čvorovi} + 2 * P = 14 - 9 + 2 * 1 = 7$$

### 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 braću.
- b) Ana voli jednog Marijinog brata.

Koristi predikate  $\text{voli}(x,y)$ ,  $\text{brat}(x,y)$

- a)  $\forall x[\text{brat}(x, \text{Marija}) \Rightarrow \text{voli}(\text{Ana}, x)]$
- b)  $\exists x[\text{brat}(x, \text{Marija}) \wedge \text{voli}(\text{Ana}, x)]$

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

$$AG(EF \text{ početak})$$

### 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?

$$CV(G) = \text{Lukovi} - \text{Čvorovi} + 2 * P = 12 - 8 + 2 * 1 = 6$$

### 10. Zadana je funkcija $\text{double volumen\_kvadra}(\text{double } a, \text{double } b, \text{double } c);$ Napiši 4 testna primjera.

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

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

`return -1;`

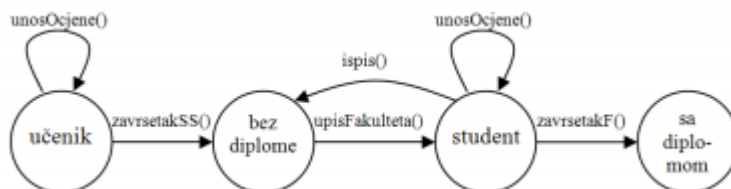
`if (*a>*b)`

`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.



a) Kako oblikovati testni slučaj?

b) Navedite ga

a) Tako da se prođu sva stanja i svi prijelazi

b) `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)`

$\exists x(\text{životinja}(x) \wedge \text{macak}(x) \wedge \text{jede\_sir}(x))$

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

$\forall x \forall y ((\text{miš}(x) \wedge \text{najveći}(x) \wedge \text{mačka}(y) \wedge \text{najmanji}(y)) \Rightarrow \neg \text{boji}(x,y))$

15. Ako nema mačke, miševi mogu pojesti i cijeli kolut sira. Zadano: `nema_mačke`, `pojedeni_sir`

$EF(\text{nema\_mačke} \wedge \text{pojedeni\_sir})$

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

$AG((\text{upaljeno} \wedge \neg \text{ugašeno}) \vee (\neg \text{upaljeno} \wedge \text{ugašeno}))$

17. Pretvori u predikatnu logiku: a) Samo mrtav vanzemaljac je dobar vanzemaljac

b) Za svaki brijeg u Hrvatskoj postoji viši brijeg u Hercegovini

c) Seviljski brijač brije sve ljude osim onih koji se sami briju

d) U Hrvatskoj postoje najmanje 2 nacionalna parka

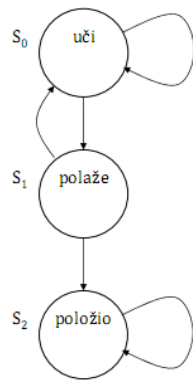
a)  $\forall x [(\text{vanzemaljac}(x) \wedge \text{dobar}(x)) \Rightarrow \text{mrtav}(x)]$

b)  $\forall x [(\text{brijeg}(x) \wedge \text{Hrv}(x)) \Rightarrow \exists y (\text{Herc}(y) \wedge \text{viši}(y,x))]$

c)  $\forall x \forall y [(\text{Sev}(x) \wedge \text{čovjek}(y) \wedge \text{ne\_brije}(y,y)) \Rightarrow \text{brije}(x,y)]$

d)  $\exists x \exists y [\text{Hrv}(x) \wedge \text{nac}(x) \wedge \text{Hrv}(y) \wedge \text{nac}(y) \wedge \text{raz}(x,y)]$

17. Kripke struktura zadana je grafom:



Za koja stanja vrijede formule:

a)  $AG(uči \Rightarrow AF(položio))$

b)  $E(uči \cup položio)$

a)  $S_2$

b)  $S_2$