



Pouzdanost računalnih sustava

Završni ispit – Primjeri zadataka



1. /1 bod/ Pouzdanost sustava takva da se očekivanja mogu opravdati njegovim obavljanjem zadaća definira pojam iz područja pouzdanosti:
- neosjetljivost na pogreške (engl. *fault tolerance*),
 - oslonljivost (engl. *dependability*),
 - raspoloživost (engl. *availability*),
 - sigurnost (engl. *safety*),
 - ništa od navedenog.
2. /1 boda/ Geometrijska distribucija slučajne varijable je diskretna distribucija dobivena iz eksponencijalne distribucije.
3. /1,5 bod/ Navedite 4 modela kvara prema njegovoj manifestaciji u pogrešku počevši od najniže razine u hijerarhiji kvarova:
- (1) model kvara na nivou sklopa
 - (2) model kvara na nivou logike
 - (3) model kvara na nivou sklopke
 - (4) model kvara na nivou sustava
4. /1 bod/ Sklopovske tehnike za izbjegavanje kvara su: modifikacija radne okoline, korištenje komponenti veće kvalitete i povećanje stupnja integracije sklopa.
5. /1 bod/ Ulazne i izlazne grane kombinacijskog sklopa definiraju skup:
- dominantnih kvarova,
 - ekvivalentnih kvarova,
 - mogućih mjesta kvara,
 - ispitnih točaka.
6. /1 bod/ Neka od pravila *ad-hoc* metoda oblikovanja za ispitivanje (engl. *Design for Test*) su:
- Izbjegavanje asinkrone logike.
 - Jednostavna inicijalizacija sklopova.
 - Korištenje redundancije.
 - Onemogućavanje prekidanja povratnih veza.
 - Podjela velikih brojača na manje.
7. /1 bod/ Maksimalni broj dodatnih pinova koje definira *Boundary Scan* standard je:
- 3,
 - 4,
 - 5,
 - 6,
 - nijedno od navedenog.
8. /1 bod/ Svojstva samo-dualnih logičkih funkcija su pogodna za izvedbu tehnike vremenske redundancije:
- alternirajuće logike,
 - REDWC,
 - RESO,
 - RESWO.

9. /1 bod/ Korištenjem polinoma $1+x+x^3$ kao generatora pseudoslučajnih brojeva LFSR sklopom dobije se niz od:
- 5 različitih pseudoslučajnih brojeva,
 - 7 različitih pseudoslučajnih brojeva,
 - 8 različitih pseudoslučajnih brojeva,
 - 12 različitih pseudoslučajnih brojeva,
 - nijedno od navedenog.
10. /1 bod/ Kod $3N$ kodiranja, proces kodiranja informacije odgovara:
- Množenju konstantom 3.
 - Posmaku informacije za 3 mjesta.
 - Zbrajanju s konstantom 3.
11. /1 bod/ Paritetni kôd gdje svaki bit informacije sudjeluje u stvaranju pariteta više grupa naziva se:
- bit po oktetu,
 - bit po riječi,
 - isprepleteni paritet,
 - preklapajući paritet.
12. /1 bod/ Vodopadni, iterativni, komponentni i dugovječni su:
- modeli karakteristične krivulje programske pouzdanosti,
 - modeli procesa programskog inženjerstva,
 - vrste faktora koji utječu na kvarove.
13. /1 bod/ Pretpostavka logaritamskog pada intenziteta kvarova nakon ispravljanja karakteristika je jednog od modela po:
- Halstead-u,
 - Jelinski-Moranda,
 - McCabe-u,
 - Mills-u,
 - Musa-Okumoto.
14. /2 boda/ Ako je kvorum pisanja Q_w , a kvorum čitanja Q_r , kod protokola konsenzusa kvorumom u raspodijeljenom sustavu s N poslužitelja moraju biti ispunjeni sljedeći uvjeti da bi se osigurao konzistentan rad sustava: $2Q_w > N$ i $Q_w + Q_r > N$.
15. /1 bod/ Zatajenje komponente aktivira:
- hladne pričuve,
 - tople pričuve,
 - vruće pričuve.
16. /1 bod/ Izravne i neizravne troškove posjedovanja sustava koji su smanjeni povećanom pouzdanošću sustava obuhvaćamo pojmom:
- Ukupna cijena posjedovanja (engl. TCO – Total Cost of Ownership)
17. /1 bod/ Weibullova distribucija kvarova u sustavu se svodi na eksponencijalnu kad za faktor oblika a vrijedi:
- $a = 0$,
 - $a \neq 0$
 - $a < 1$
 - $a = 1$
 - $a = 2$
18. /1 bod/ Na kojoj matematičkoj funkciji za opis raspodjele kvarova se temelji evaluacija modela kod Monte Carlo simulacije?

- a. funkciji distribucije slučajne varijable (engl. *cumulative distribution function*)
- b. funkciji gustoće funkcije vjerojatnosti (engl. *probability density function*),
- c. funkciji mjere rizika (engl. *hazard*),
- d. funkciji vremena očekivanja (engl. *MTTF*),
- e. ništa od navedenog.

19. /1 bod/ Za metodu predviđanja pouzdanosti prebrojavanja komponenti (engl. *parts count method*) vrijedi:

- a. bolja je od M-od-N model,
- b. bolja je od neserijsko-neparalelnog modela,
- c. ima lošije predviđanje od paralelnog sustava,
- d. primjenjiva je za serijski sustav.
- e. to je konzervativna metoda

20. /1 bod/ Prema empiričkoj formuli MIL-HDBK-217E modela učestalosti pogrešaka utjecaj povećanja stupnja integracije integriranih sklopova (faktori C_1 i C_2) na cjelokupnu pouzdanost sustava je:

- a. Nema značajnijeg utjecaja na pouzdanost sustava.
- b. Povećava pouzdanost sustava.
- c. Smanjuje pouzdanost sustava.

21. /1 bod/ U primjeru NI sklopa s pretpostavkom pojavljivanja *pričvršće-na-0* i *pričvršćen-na-1* kvarova analizom dominantnosti moguće je postići sažimanje broja kvarova na:

- a. 0.667,
- b. 0.5,
- c. 0.375,
- d. 0.333.

22. /1 bod/ Način rada u kojem se vrijednosti na paralelnom ulazu sklopa pohranjuju u bistabile njegovog *Boundary Scan* lanca naziva se:

- a. EXTEST.
- b. INTEST.
- c. PRELOAD.
- d. SAMPLE.
- e. Nijedno od navedenog.

23. /1 bod/ Sklop koji u sebi obuhvaća funkcionalnosti bistabila, generatora uzoraka, analize odziva i lanca ispitivanja ima skraćani naziv:

- a. BILBO,
- b. BIST,
- c. LFSR,
- d. MISR,
- e. TAP.

24. /1 bod/ Vrsta pariteta koji dijeli izvorni podatak u grupe jednake veličine i svakoj grupi pridružuje jedan paritetni bit tako da su susjedni bitovi u različitim grupama pariteta naziva se:

- a. Bit po riječi,
- b. Bit po čipu,
- c. Isprepleteni paritet,
- d. Preklapajući paritet.
- e. Ništa od navedenog.

25. /1 bod/ Vrsta zaštitnog informacijskog koda koji stvara zaštitne bitove na način da prebroji broj jedinica u izvornoj informaciji, komplementira tu vrijednost i doda je u binarnom obliku na kraj izvorne informacije zove se:

- a. Bergerov kôd,
- b. Duplikacijski kôd,
- c. *m-od-n* kôd,
- d. rezidualni kôd,
- e. nijedno od navedenog.

26. /1 bod/ Informacijska zaštita s *Hammingovom* udaljenošću $H_d=3$ ima sposobnost otkrivanja:

- a. Jednostrukih pogrešaka.
- b. Dvostrukih pogrešaka.
- c. Trostrukih pogrešaka.

27. /1 bod/ Predstavnik modela pouzdanosti programske potpore temeljenog na ubacivanju pogrešaka je:

- a. *Halsteadov* model,
- b. *Jelinski-Moranda* model,
- c. *McCabeov* model,
- d. *Millsov* model.

28. /1 bod/ Prema *Lamportovom* algoritmu minimalan broj lojalnih generala u slučaju M izdajica koji treba biti da bi komunikacija bila pouzdana i postignut ispravan dogovor je:

- a. $M+1$,
- b. $2M+1$,
- c. $3M+1$,
- d. $4M+1$.

29. /1 bod/ Tehnika postizanja visoke raspoloživosti kod koje su aktivne i redundantne komponente grupirane i logički nerazdvojive korisnicima, a vrijeme oporavka trenutačno naziva se:

- a. hladna pričuve,
- b. topla pričuve,
- c. vruća pričuve.

30. /1 bod/ Dokumentacija i prioriteta, odabir položaja sustava, ključnih ljudi i zamjena, uvježbavanje, razvoj i provedba testova i stalno nadopunjavanje su ključni elementi

plana oporavka od katastrofalnog događaja (engl. *disaster recovery*).

31. Navedite tehnike za procjenu vrijednosti parametara matematičkih distribucija iz eksperimentalnih podataka. U tablici je, za neki sustav, prikazan broj uočenih pogrešaka u vremenskim intervalima.

Vremenski interval	Broj pogrešaka
0-10	98
10-20	56
20-30	33
30-40	14
40-65	9

Opisuje li eksponencijalna razdioba dovoljno dobro eksperimentalno dobivene podatke o pogreškama uz pretpostavljenu razinu prihvatljivosti:

- $\alpha = 0.05$?
- Ako da (u odnosu na a.), vrijedi li isto za razinu prihvatljivosti $\alpha = 0.10$?
- Ako ne (u odnosu na a.), odredite je li eksponencijalna razdioba prihvatljiva za razinu prihvatljivosti $\alpha = 0.01$?

Napomena. Za procjenu parametra λ koristite metodu maksimalne sličnosti pri čemu pretpostavite uniformnu razdiobu pogrešaka unutar mjerenog.

Rješenje:

Tehnike za procjenu parametara: procjena točke, procjena maksimalne sličnosti, metoda momenata, analiza linearne regresije, intervali povjerenja.

Vremenski interval P - K	Sredina intervala	Broj pogrešaka O_i	E_i	$(O_i - E_i)^2 / E_i$
0-10	5,0	98	102,93	0,24
10-20	15,0	56	52,48	0,24
20-30	25,0	33	26,76	1,46
30-40	35,0	14	13,64	0,01
40-65	52,5	9	11,56	0,56

Ukupno: 210

$$\lambda = 0,0674$$

$$\chi^2 = 2,5049$$

Broj stupnjeva slobode: $5 - 1 - 1 = 3$

Pearove tablice: $\alpha = 0.05$: $\chi^2_{0.05} = 7,815$, $\chi^2 < \chi^2_{0.05} \rightarrow$ zadovoljava

$\alpha = 0.10$: $\chi^2_{0.10} = 6,251$, $\chi^2 < \chi^2_{0.10} \rightarrow$ zadovoljava

32. Navedite tehnike za procjenu vrijednosti parametara matematičkih raspodjela iz eksperimentalnih podataka. U *Tablici* je, za neki sustav, prikazan broj uočenih pogrešaka u vremenskim intervalima.

Vremenski interval	Broj pogrešaka
0-10	108
10-20	78
20-30	30
30-40	13
40-80	3

Odgovorite na pitanja:

- Opisuje li eksponencijalna razdioba dovoljno dobro eksperimentalno dobivene podatke o pogreškama uz pretpostavljenu razinu prihvatljivosti $\alpha = 0.05$?
- Ako da (u odnosu na a.), vrijedi li isto za razinu prihvatljivosti $\alpha = 0.10$?
- Ako ne (u odnosu na a.), odredite je li eksponencijalna razdioba prihvatljiva za razinu prihvatljivosti $\alpha = 0.01$?

Rješenje:

- Tehnike za procjenu vrijednosti parametara matematičkih raspodjela iz eksperimentalnih podataka su:

- Procjena točke (engl. *point estimation*)
- Procjena maksimalne sličnosti (engl. *maximum likelihood estimators*)
- Metoda momenata (engl. *method of moments*)
- Analiza linearne regresije (engl. *linear regression analysis*)
- Intervali povjerenja (engl. *confidence interval*)

- Metoda maksimalne sličnosti:

	Oi		Ei		chi-square	
			0		0	
0-10	5,0	108	540	0,5274	122,37	1,69
10-20	15,0	78	1170	0,2492	57,83	7,04
20-30	25,0	30	750	0,1178	27,33	0,26
30-40	35,0	13	455	0,0557	12,91	0,00
40-80	60,0	3	180	0,0474	10,99	5,81
		232	3095		13,34052	0,07496
					14,80	

- Iz *Pearovih* tablica:

- $df = 5 - 1 - 1 = 3,$
 - $\alpha = .1 \rightarrow 6.251 < 14.80$
 - $\alpha = .05 \rightarrow 7.815 < 14.80$
 - $\alpha = .01 \rightarrow 11.345 < 14.80$

- → ne zadovoljava ni za jednu razinu prihvatljivosti

33. Za sustav prikazan slikom 1 koji čine 2-od-4 sustav procesora s glasačem spojen sa 2-od-3 sustavom memorija s glasačem, odredite izraz za pouzdanost. Neka je pouzdanost pojedinih procesora R_p , memorija R_m te pojedinih glasača R_v

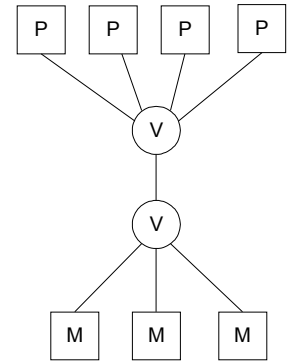
Rješenje:

Za M-od-N sustave vrijedi: $R_{M\text{-od-}N} = \sum_{i=0}^{N-M} \binom{N}{i} R^{N-i} (1-R)^i$

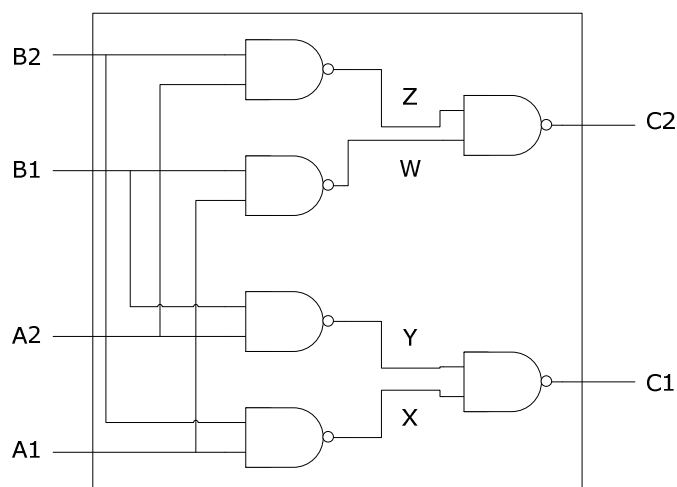
$$R_{\text{procesori}} = R_v * R_{2\text{-od-}4} = R_v * (6R_p^2 - 8R_p^3 + 3R_p^4)$$

$$R_{\text{memorije}} = R_v * R_{2\text{-od-}3} = R_v * (3R_m^2 - 2R_m^3)$$

$$\begin{aligned} R_{\text{sustav}} &= R_{\text{procesori}} * R_{\text{memorije}} = \\ &= [R_v * (6R_p^2 - 8R_p^3 + 3R_p^4)] * [R_v * (3R_m^2 - 2R_m^3)] \end{aligned}$$



34. Objasnite svojstva dvotračnog ispitnog sklopa (na *Slici 1*). Za slučajeve kvarova „A2 pričvršćen-na-0“ i „W pričvršćen-na-1“ prikažite postupak otkrivanja pogreške.



Slika 1. Dvotračni ispitni sklop

Rješenje:

Otkriva greške *pričvršćen-na-0* i *pričvršćen-na-1* pod pretpostavkom dovođenja alternirajućih parova ulaza (= *valjanih kodnih riječi*). Ako se na izlazu ne pojavi valjana kodna riječ greška je otkrivena.

$$X = \overline{A1 \cdot B2}$$

$$W = \overline{A1 \cdot B1}$$

$$Y = \overline{A2 \cdot B1}$$

$$Z = \overline{A2 \cdot B2}$$

$$C1 = \overline{X \cdot Y}$$

$$C2 = \overline{A2 \cdot B2}$$

B2	B1	A2	A1	C2	C1
0	1	0	1	1	0
0	1	1	0	0	1
1	0	0	1	0	1
1	0	1	0	1	0

Za kvar *A2 pričvršćen-na-0* greška se očituje u dva kombinacije ulaza:

B2	B1	A2	A1	C2	C1
0	1	0	1	1	0
0	1	0	0	0	0
1	0	0	1	0	1
1	0	0	0	0	0

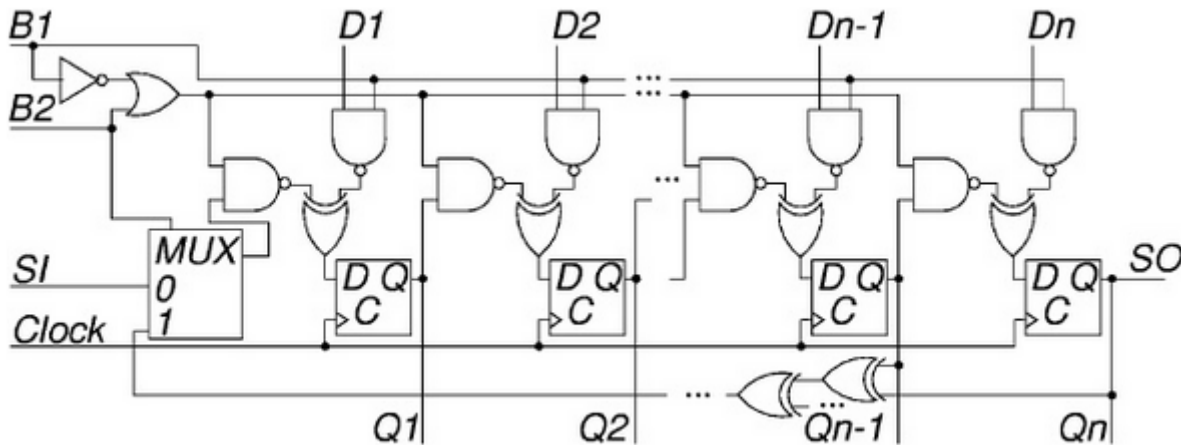
Za kvar *W pričvršćen-na-1* greška se očituje u dva kombinacije ulaza:

B2	B1	A2	A1	C2	C1
0	1	0	1	0	0
0	1	1	0	0	1
1	0	0	1	0	1
1	0	1	0	1	0

35. Prikažite strukturu i navedite sve moguće načine rada BILBO sklopa s ugrađenom logikom za promatranje.

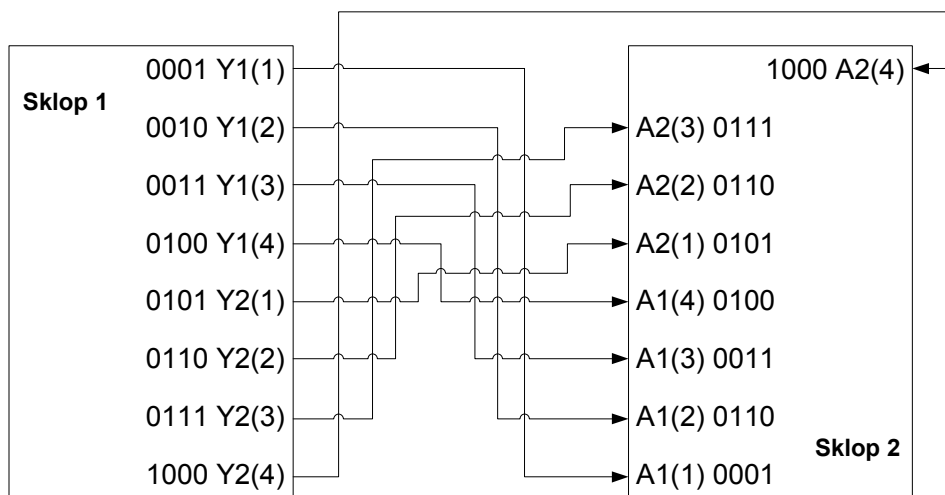
Rješenje:

- Obuhvaća funkcionalnosti bistabila, generatora uzoraka, analize odziva i lanca ispitivanja.
- Struktura BILBO registra:



- Načini rada:
 - $B1 = 0, B2 = 0$: Scan Mode, linear shift register. **Ispitni rad.** Upis podataka u bistabile/čitanje.
 - $B1 = 0, B2 = 1$: **Generiranje uzoraka.**
 - $B1 = 1, B2 = 0$: Normal Mode. **Normalan način rada.**
 - $B1 = 1, B2 = 1$: MISR Mode. **Testiranje signature višestrukih ulaza.**

36. Ako je dana shema tiskane pločice kao na *Slici*, objasnite kako se *boundary scan* metodom određuje do kojeg je kvara došlo te provedite analizu kvara.



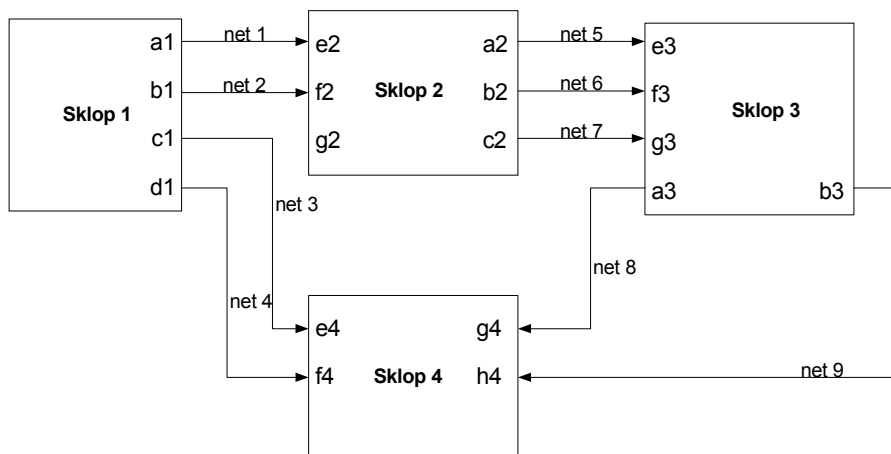
Rješenje:

- *EXTEST* način rada.
- Prema algoritmu brojanja se određuje
 - broj ispitnih vektora: $\lceil \log_2(n+2) \rceil = \lceil 3.322 \rceil = 4$ ($n=8$ priključnica)
 - i način na koji se definiraju (kao na slici, uzlazno brojeći od 1)
- Analiza:

Sklop 1		→	Sklop 2	
Y1(1)	0001		A1(1)	0001
Y1(2)	0010		A1(2)	0110
Y1(3)	0011		A1(3)	0011
Y1(4)	0100		A1(4)	0100
Y2(1)	0101		A2(1)	0101
Y2(2)	0110		A2(2)	0110
Y2(3)	0111		A2(3)	0111
Y2(4)	1000		A2(4)	1000

- Zaključak: → Y1(2) i Y2(2) → wired OR

37. Za tiskanu pločicu prikazanu na *Slici 2*, načinite analizu kvara prospojen mreže *boundary scan* metodom. Pretpostavite da su linije *net 1* i *net 4* kratko spojene te da se kvar manifestira kao *wired AND*; te da je linija *net 7* prekinuta i kvar se odražava kao „pričvršćen na 1“. Odredite minimalno jedan skup ispitnih vektora za svaki od kvarova.



Slika 2. Tiskana pločica za *boundary scan* testiranje

Rješenje:

Za kratko spojene (*wired AND*) *net 1* i *net 4* na Sklopu 1:

SAMPLE/PRELOAD naredbama postavljamo izlaze *a1*, *b1*, *c1* i *d1* Sklopa 1, a naredbom EXTEST ih propuštamo na paralelne ulaze Sklopova 2 i 4.

Broj ispitnih vektora: $\#(ispita) = \lceil \log_2(n+2) \rceil$, gdje je $n = 4$

$$\rightarrow \#(ispita) = \lceil \log_2 6 \rceil = \lceil 2.585 \rceil = 3$$

Postavljamo ih na *a1*, *b1*, *c1* i *d1* po algoritmu brojanja.

vektori 12

(Sklop 1) ↓↓

a1: 01

b1: 01

c1: 01

d1: 01

(Sklop 2 i 4)

e2: 000

f2: 010

e4: 011

f4: 000

→ Na *e2* i *f4* su očitane neispravne vrijednosti.

Za prekinutu *net 7* liniju (*pričvršćen-na-1*):

Alterniramo vrijednosti na linijama. Dovodimo dva ispitna vektora na Sklop 1 naredbom SAMPLE/PRELOAD, npr. 0000 i 1111, pa ih propuštamo na ulaze Sklopa 2 naredbom EXTEST. Naredbom PRELOAD ih posmičemo na izlaze Sklopa 2 i naredbom EXTEST propuštamo na ulaze Sklopa 3.

vektori 123

(Sklop 1) ↓↓↓

a1: 001

b1: 010

c1: 011

d1: 100

(Sklop 2)

e2: 01

f2: 01

(Sklop 3)

e3: 01

f3: 01

g3: **11**

→ Na g3 je očitana neispravna vrijednost.

38. Navedite klasifikaciju aritmetičkih kodova i objasnite način kodiranja za svaki od njih. Aritmetičkim rezidualnim *modulo-7* kôdom zaštitite podatke 101000110 i 1001111010. (Detaljno prikazati postupak kodiranja.)

Rješenje:

Klasifikacija:

- AN kod
 - Množenje podatka s konstantom
 - Iskorištava svojstvo: $A(N+M) = AN + AM$
- Rezidualni kodovi
 - Dijeljenje podatka i zapis ostatka dijeljenja (dodatnih bitova) na kraj izvornog podatka
 - Vrijedi svojstvo: $(M+N) \bmod k = (M \bmod k + N \bmod k) \bmod k$
 - Za neke vrijednosti djelitelja dijeljenje je moguće zamijeniti modulo zbrajanjem ($N = 3, 7, 15$)
- Ispitne sume
 - Izračun sume i slanje zajedno s izvornim podatkom
 - Vrste:
 - Jednostruke preciznosti
 - Dvostruke preciznosti
 - Honeywell
 - S ostatkom

Rezidualni *modulo-7* kôd – *low cost* kôd → moguće je dijeljenje zamijeniti s modulo-7 zbrajanjem (podjelom u grupe od po tri bita).

- Podatak 101000110 (= 326)

101	000	110
↓		↓ (modulo-7 zbrajanje)
101		110
↓		
100 (= 4) → provjera: $326 \bmod 7 = 4$		

- Podatak 1001111010 (= 634)

001	001	111	010
↓		↓ (modulo-7 zbrajanje)	
010		010	
↓			
100 (= 4) → provjera: $634 \bmod 7 = 4$			

39. Proširite *Hamming*-ov (11,7) kôd u svrhu otkrivanja dvostrukih pogrešaka te napišite odgovarajuću matricu provjere pariteta. Pokažite način rada takvog kôda za ulazni podatak 1110001 te uz pretpostavku jednostruke pogreške u podatkovnim bitovima i dodatne jednostruke pogreške u zaštitnim bitovima usporediti ga s početnim (11,7) kôdom.

Rješenje:

Izvorni podatak (7 bita): $k_7 k_6 k_5 k_4 k_3 k_2 k_1$

Bitovi zaštite (4 bita): $C_3 C_2 C_1 C_0$

Kodirani podatak (11 bita): $C_0 C_1 k_1 C_2 k_2 k_3 k_4 C_3 k_5 k_6 k_7$

Izračun zaštitnih bitova:

$$C_0 = k_1 \text{ XOR } k_2 \text{ XOR } k_4 \text{ XOR } k_5 \text{ XOR } k_7$$

$$C_1 = k_1 \text{ XOR } k_3 \text{ XOR } k_4 \text{ XOR } k_6 \text{ XOR } k_7$$

$$C_2 = k_2 \text{ XOR } k_3 \text{ XOR } k_4$$

$$C_3 = k_5 \text{ XOR } k_6 \text{ XOR } k_7$$

Dodatni zaštitni bit:

$$C_4 = C_0 \text{ XOR } C_1 \text{ XOR } k_1 \text{ XOR } C_2 \text{ XOR } k_2 \text{ XOR } k_3 \text{ XOR } k_4 \text{ XOR } C_3 \text{ XOR } k_5 \text{ XOR } k_6 \text{ XOR } k_7$$

Matrica provjere pariteta (izračun *sindroma*):

$$\begin{bmatrix} S_0 \\ S_1 \\ S_2 \\ S_3 \\ S_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \oplus \begin{bmatrix} C_0 \\ C_1 \\ k_1 \\ C_2 \\ k_2 \\ k_3 \\ k_4 \\ C_3 \\ k_5 \\ k_6 \\ k_7 \\ C_4 \end{bmatrix}$$

Za izvorni podatak 1110001 kodirani podatak je: 11101101001 gdje su podvučeni bitovi zaštite.

Ako se npr. dogodi dvostruka greška na bitovima C_0 i k_2 izračun sindroma daje 0100 što bi značilo da je greška na četvrtom bitu što nije točno.

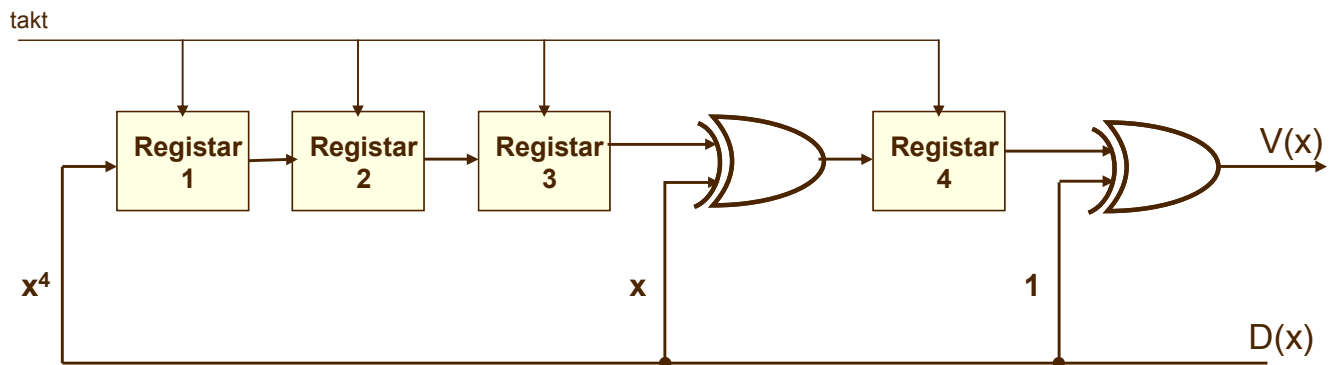
Uvođenjem dodatnog bita pariteta izračunati sindrom je 00100 što je proturječno. Dodatni bit kaže da je sve u redu (0), a sindrom pokazuje na pogrešku četvrtog bita (0100).

40. Nacrtajte sklop za kodiranje cikličkog kôda generatorom polinoma $G(x) = 1 + x + x^4$. Prikažite korake kodiranja podatka 01110.

Dobivena kodna riječ: _____.

Rješenje:

Sklop za kodiranje:



$$V(x) = G(x) \cdot D(x)$$

$$V(x) = (x^4 + x + 1) \cdot (x^3 + x^2 + x) = x^7 + x^6 + x^5 + x^4 + x \rightarrow 11110010 \text{ (gledano od najznačajnijeg bita)}$$

Kodiranje po koracima:

takt	R1	R2	R3	R4	D(x)	V(x)
0	0	0	0	0	0	0
1	0	0	0	0	1	1
2	1	0	0	1	1	0
3	1	1	0	1	1	0
4	1	1	1	1	0	1
5	0	1	1	1	0	1
6	0	0	1	1	0	1
7	0	0	0	1	0	1
8	0	0	0	0		

→ Dobivena kodna riječ: $V(x) = 01001111$ (gledano od najmanje značajnog bita).

41. Navedite svojstva m -od- n koda u pogledu načina kodiranja i dekodiranja te sposobnosti otkrivanja jednostrukih i dvostrukih pogrešaka. Raspišite kôd 3-od-6.

Rješenje:

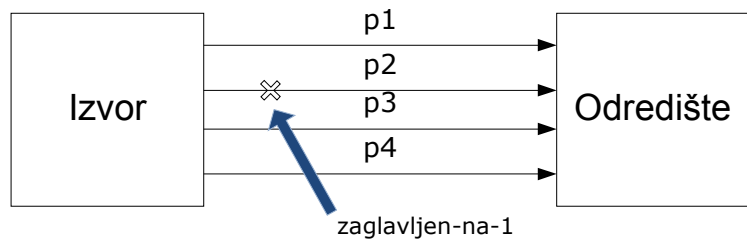
- n bita
- točno m jedinica u svakoj kodnoj riječi
- jednostruka greška $\rightarrow m-1$ ili $m+1$ jedinica
- separabilni kôd:
 - jednostavno kodiranje i dekodiranje
 - $H_d = 2$
 - Otkriva jednostruke greške, ali ne mora otkriti sve dvostruke greške.

Najjednostavnija implementacija – dodavanje i bitova na i bitova izvorne informacije:

Informacija	Kodirana informacija 3-od-6
000	000 111
001	001 110
010	010 101
011	011 100
100	100 011
101	101 010
110	110 001
111	111 000

42. Blok podataka 0111011001100101 se u paketima po 4 bita šalje preko komunikacijskog kanala. Pretpostavite da komunikacijski kanal unosi pogrešku te se ona manifestira kao *pričvršćen-na-1* kvar na liniji p2. Ispravnost primljenih podataka se na odredištu provjerava na principu ispitnih suma. Usporediti načine otkrivanja pogreške primjenom:

- ispitne sume jednostruke preciznosti,
- ispitne suma dvostruke preciznosti,
- Honeywell ispitne sume,
- ispitne sume s ostatkom.



Rješenje:

a. Ispitna suma jednostruke preciznosti ignorira preljev.

Na izvoru: Na odredištu (p2 pričvršćen-na-1):

0111	0111
0110	0110
0110	0110
<u>0101 +</u>	<u>0101 +</u>
1000	1000

Budući da je p2 zaglavljen na 1, prijenosom ispitne sume na odredištu dobivamo 1100 != 1000, odnosno detektira se pogreška.

b. Ispitna suma dvostruke preciznosti uzima u obzir preljev.

Na izvoru: Na odredištu (p2 pričvršćen-na-1):

0111	0111
0110	0110
0110	0110
<u>0101 +</u>	<u>0101 +</u>
0001 1000	0001 1000

Prijenosom ispitne sume 0001 1000 na odredištu dobivamo 0101 1100 != 0001 1000 → detektira se pogreška!

c. Honeywell ispitna suma

Na izvoru: Na odredištu (p2 pričvršćen-na-1):

0110 0111	0110 0111
<u>0101 0110 +</u>	<u>0101 0110 +</u>
1011 1101	1011 1101

Prijenosom ispitne sume 1011 1101 na odredištu dobivamo 1111 1111 != 1011 1101 → detektira se pogreška.

d. Ispitna suma s ostatkom.

Na izvoru: Na odredištu (p2 pričvršćen-na-1):

0111	0111
0110	0110

$$\begin{array}{r}
 0110 \\
 \underline{0101 +} \\
 \quad 1 + \\
 \hline
 1001
 \end{array}$$

$$\begin{array}{r}
 0110 \\
 \underline{0101 +} \\
 \quad 1 + \\
 \hline
 1001
 \end{array}$$

Prijenosom ispitne sume 1001 na odredištu dobivamo 1101 == 1001 → ne detektira se pogreška!

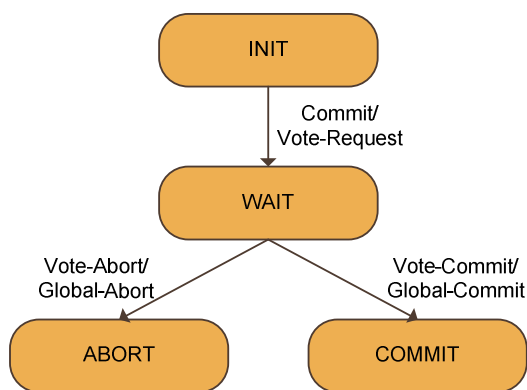
43. Skicirajte karakteristične sekvence rada protokola potvrđivanja u dvije faze (*Two-Phase Commit*) kod raspodijeljenih sustava za scenarije uspješne i neuspješne transakcije. Navesti koji se problemi javljaju i objasniti potrebu protokola potvrđivanja u tri faze (*Three-Phase Commit*).

Rješenje:

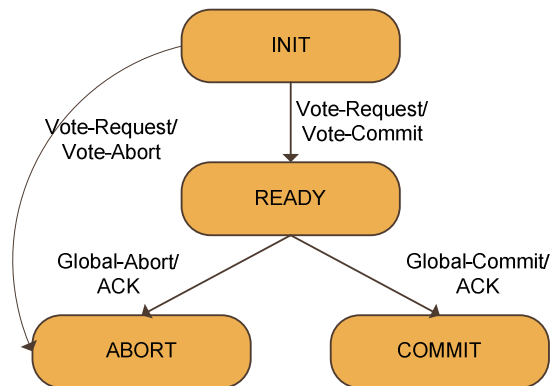
Protokol potvrđivanja u 2 faze (2PC):

Faza 1. Glasanje

Faza 2. Odluka o potvrđivanju

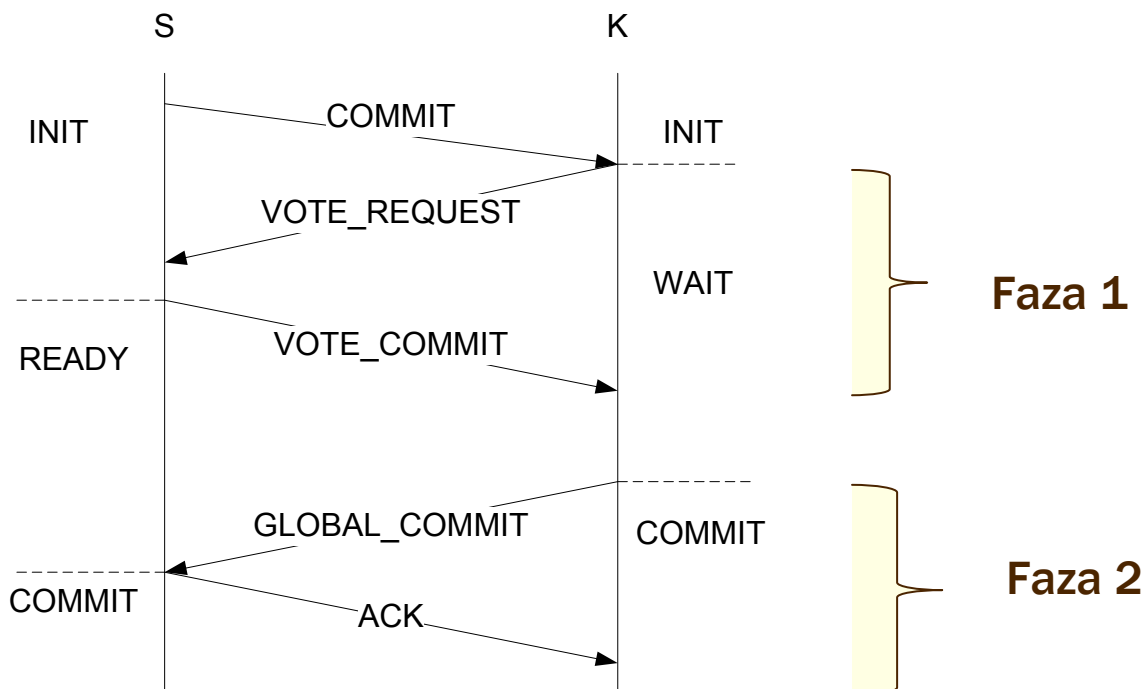


Na strani koordinatora



Na strani sudionika

Po fazama:



Kod ispada koordinatora sudionici pitaju ostale sudionike o stanju:

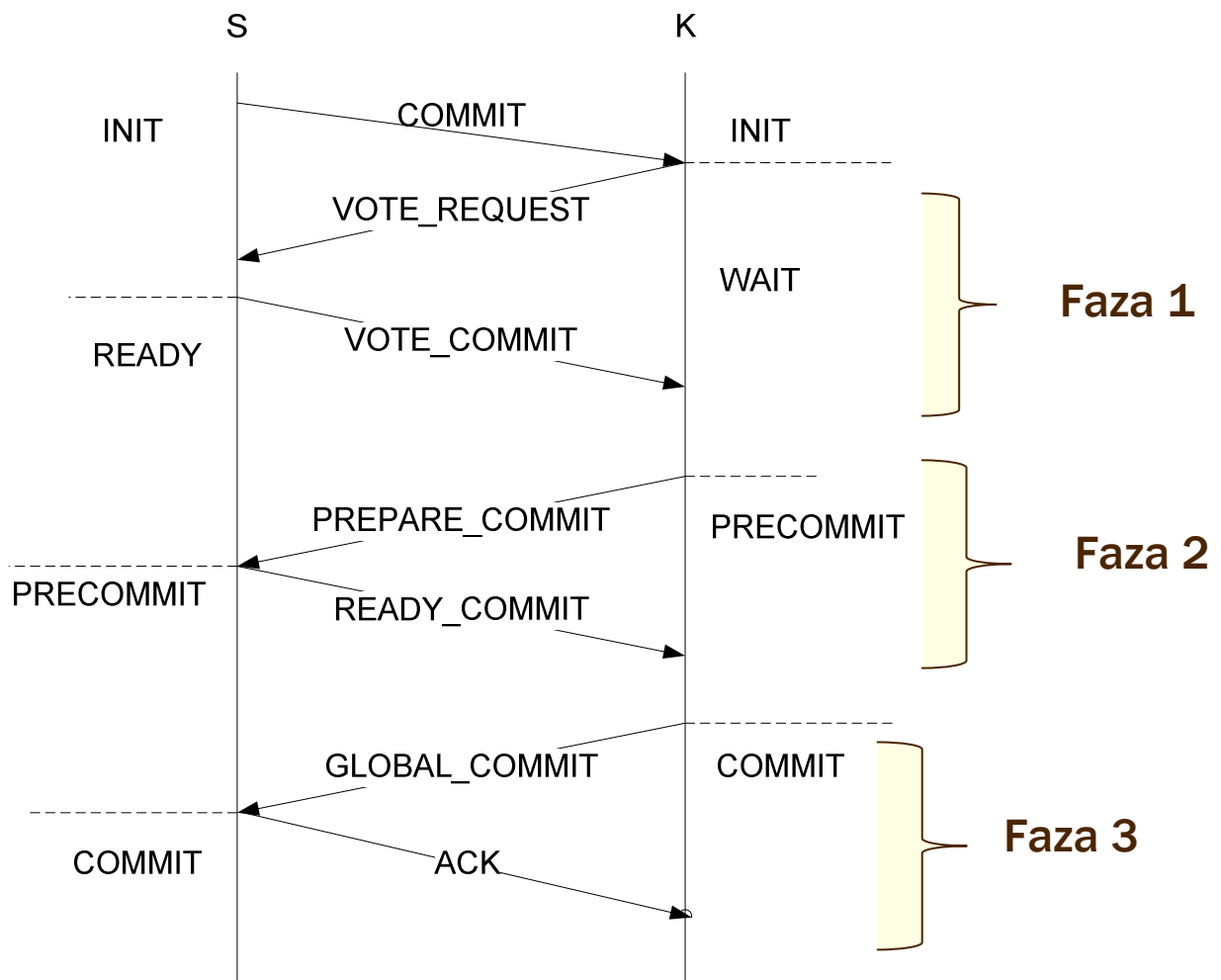
- Ako je neki čvor još u stanju INIT → ide se u ABORT.
- Ako je neki čvor u stanju ABORT → ide se u ABORT.
- Ako je neki čvor u stanju COMMIT → ide se u COMMIT.
- Ako su svi čvorovi u stanju READY → čeka se oporavak koordinatora.

➔ Problem je ako se neki čvor sruši da se može oporaviti u stanje COMMIT ili ako koordinators dugo nije u funkciji i ne može se donijeti odluka. Čvorovi ne mogu potvrditi transakciju iako su svi spremni za potvrđivanje jer je neki od sudionika možda ispao i ne zna se kako bi on glasao.

Rješenje: protokol potvrđivanja u tri faze (3PC). Zahtjevi:

- Ne postoji stanje iz kojeg se direktno može u COMMIT i ABORT
- Ne postoji stanje iz kojeg se ne može donijeti konačna odluka, a da se iz njega može napraviti prijelaz u COMMIT

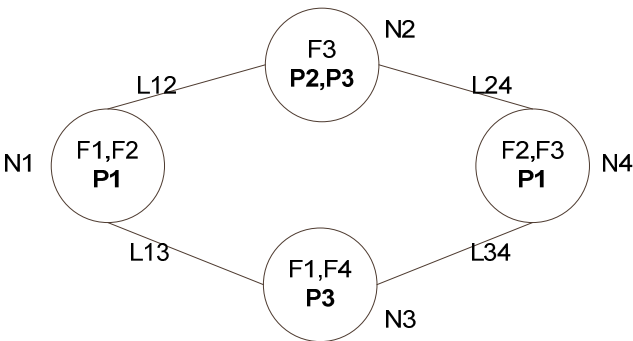
Dodana je potvrda odluke (stanje PREPARE_COMMIT).



Ispad koordinatora - kao i za 2PC sudionik može kontaktirati ostale sudionike i:

- iz stanja READY može otići u stanje ABORT:
 - ako su ostali u stanju INIT ili READY
- i u COMMIT ako su svi sudionici u stanju PRECOMMIT β razlika u odnosu na 2PC

44. Neka je dan raspodijeljeni sustav kojeg sačinjava 4 čvora N1 – N4 te se na njima izvode tri programa P1 - P3, slika 2. Za uspješno izvođenje pojedinih programa moraju biti dostupne datoteke F1 – F4 prema sljedećim zavisnostima: $P1 \rightarrow \{F1, F2, F3\}$, $P2 \rightarrow \{F1, F2, F4\}$, $P3 \rightarrow \{F1, F2, F3, F4\}$. Primjenom Kumarova algoritma za određivanje pouzdanosti raspodijeljene programske podrške, odredite skup minimalnih stabala pokrivanja datoteka za program P2.



Rješenje:

Zadano je:
 $P2 \rightarrow \{F1, F2, F4\}$; $P2 \in N2 \rightarrow N2$ uvijek mora biti u FST
 $F1 \in \{N1, N3\}$,
 $F2 \in \{N1, N4\}$,
 $F4 \in N3 \rightarrow N3$ uvijek mora biti u FST

$F1 \in N1, F2 \in N1$	$FST1 = \{N2, N1, N3, L12, L13\}$
$F1 \in N1, F2 \in N4$	$FST2 = \{N2, N1, N3, N4, L12, L13, L34\}$ $FST3 = \{N2, N4, N3, N1, L24, L34, L13\}$
$F1 \in N3, F2 \in N1$	$\{N2, N1, N3, L12, L13\} = FST1$
$F1 \in N3, F2 \in N4$	$FST4 = \{N2, N4, N3, L24, L34\}$

Minimalno stablo pokrivanja datoteka je FST za koje ne postoji niti jedno drugo stablo FST_j t.d.
 $FST \subseteq FST_j$

$FST1 \subseteq FST2 \rightarrow FST2$ nije minimalno
 $FST4 \subseteq FST3 \rightarrow FST3$ nije minimalno
Rješenje: MFST = $FST1 = \{N2, N1, N3, L12, L13\}$, $FST4 = \{N2, N4, N3, L24, L34\}$

45. Objasnite što znači mjera ciklomatske složenosti i unutar kojeg modela pouzdanosti je definirana.

Rješenje:

Mjera strukturne složenosti. To je gornja granica broja ispita koji mogu pokriti sve moguće put.eve kod izvođenja programa. Koristi se u *McCabeovom* modelu te broji lukove, čvorove i povezane dijelove programa.

$CV(G) = \text{Lukovi} + \text{Čvorovi} + 2P$, P = broj povezanih komponenti (potprograma, prekidnih rutina)

46. Za odsječak kôda:

```
if (i < 10)
{
    if (j > 100)
    {
        k = i + j;
    }
}
```

definirajte i izračunajte vrijednosti parametara te procjenu broja pogrešaka programa prema metrici *Halsteadovog* modela pouzdanosti.

Rješenje:

Broj različitih operanada $n_2 = 5$ (i, 10, j, 100, k)

Ukupni broj operanada $N_2 = 7$ (i, 10, j, 100, k, i, j)

Broj različitih operatora $n_1 = 10$ (if, (, <,), {, }, >, =, +, ;)

Ukupni broj operatora $N_1 = 15$ (if, if, (, (, <,),), {, {, }, }, >, =, +, ;)

→ Volumen programa $V = (N_1 + N_2) \log(n_1 + n_2) = 22 \cdot \log 15 = 25.87$

47. Za odsječak kôda:

```
if (i > 0)
{
    for (j = 0; j < 8; j++)
    {
        for (k = 0; k < 8; k++)
        {
            sum = sum + a[j] * b[k];
        }
    }
}
```

definirajte i izračunajte vrijednosti parametara te procjenu broja pogrešaka programa prema metrici *Halstead*-ovog modela pouzdanosti.

Rješenje:

Broj različitih operanada $n_2 = 8$ (i, 0, 8, j, k, sum, a, b)

Ukupni broj operanada $N_2 = 18$ (i, 0, j, 0, j, 8, j, k, 0, k, 8, k, sum, sum, a, j, b, k)

Broj različitih operatora $n_1 = 13$ (if, (, >,), {, for, =, <, ++, +, * }, ;)

Ukupni broj operatora $N_1 = 26$ (if, (, (, (, >,),),), {, {, {, for, for, =, =, =, <, <, ++, ++, +, * }, }, }, ;)

→ Volumen programa $V = (N_1 + N_2) \log(n_1 + n_2) = 44 \cdot \log 21 = 58.18$

→ E = broj pogrešaka = $V/3000$

48. Navedite mogućnosti spremanja stanja i oporavka programske potpore metodom kontrolnih točki po pitanju:

- a. informacija koje se spremaju,
- b. opsega i učestalosti spremanja stanja,
- c. sinkronizma kod oporavka.

Rješenje:

Uspostavljanje prethodno sačuvanog stanja u slučaju zatajenja

Informacije koje se spremaju:

- Varijable procesa
- Okolina
- Vrijednosti registara

Mogućnosti:

- Potpuno - spremanje čitavog stanja
- Inkrementalno - samo dijelovi koji su promijenjeni od posljednje kontrolne točke

Učestalost:

- Slučajan odabir
- Fiksni intervali
- Broj uspješno obavljenih transakcija

Asinkroni oporavak – nekoordinirano među čvorovima

- Veća učestalost kontrolnih točki
- Jednostavnije – koristi se kada je:
 - Vjerojatnost zatajenja manja
 - Ograničena komunikacija među čvorovima

Sinkroni oporavak

- Koordinirano postavljanje kontrolnih točaka
- Potreban manji broj kontrolnih točaka

49. Objasnite SPE klasifikaciju razvoja programske potpore.

Rješenje:

(1985) Lehman, Belady – promatranje promjena u programima i varijacije

SPE klasifikacija:

- S-tip
 - Rješenja specifičnih problema.
 - Moguće formalno definirati i automatski generirati.
- P-tip
 - Rješenja problema iz stvarnog života.
 - Složeni problemi, samo djelomična rješenja.
 - „*never-ending change*“
- E-tip
 - Interakcija s korisnikom – ugrađeni u radnu okolinu.
 - Pomoć za rješavanje zadataka.

50. Navedite ciljeve i ključne elemente plana oporavka od katastrofalnog događaja (engl. *disaster recovery plan*).

Rješenje:

Ciljevi: preživjeti, rasti, imati budućnost nakon izvanrednih okolnosti.

Ključno je planiranje → engl. *disaster recovery plan*, a ključni elementi:

- Dokumentacija i prioriteta – HW, SW, mreža.
- Odabir položaja (s obzirom na vjerojatnosti nepogodnih okolnosti).
- Kvalitetan tim ljudi – ključni ljudi i zamjene.
- Uvježbavanje – stalne provjere.
- Razvoj i provedba ispitivanja (testova).
- Stalno planiranje i nadopunjavanje plana.