

# OS – Parcijalni Test I

In no particular order:

## **1. Promena (switching) procesa ne nastaje prilikom:**

- Kreiranja novog procesa, čiji je prioritet manji od procesa koji se trenutno izvršava.

## **2. Bolje performanse višenitnog sistema posledica su:**

- Mnogo manje vremena potrebnog za promenu konteksta niti.

## **3. Sistemskim pozivom prelazi se u:**

- Kernel mode

## **4. Sistemski poziv predstavlja:**

- Poziv funkcija operativnog sistema od strane aplikativnog programa

## **5. Jedna od sledećih stanja nije stanje niti:**

- SUSPENDOVANA (nije stanje niti)

## **6. Promene konteksta niti implementiranih na nivou kernela (KLT) su:**

- Sporije u odnosu na implementacije niti na nivou korisnika (ULT)

## **7. Sledeći događaj dovodi do kreiranja procesa:**

- Korisnik za terminalom se prijavljuje (login) na sistem.

## **8. Kontekst niti predstavlja:**

- Stanje registara procesora u trenutku prekida/blokiranja niti.

## **9. Promena (switching) procesa je pojam koji se odnosi na:**

- Prebacivanje procesa iz stanja Izvršava Se u neko drugo stanje i prebacivanje jednog procesa iz stanja Spreman u stanje Izvršava Se.

## **10. Kod hibridne implementacije niti:**

- Jedna ili više niti na nivou korisnika preslikava se u manji ili jednak broj niti na nivou kernela

## **11. Proces koji je blokiran na uslovnoj promenljivoj S monitora deblokira se:**

- Izvođenjem operacije semSignal(S)

## **12. Detekcija uzajamnog blokiranja ima za cilj:**

- Oporavak od uzajamnog blokiranja ukoliko postoji.

## **13. Vreme zadržavanja (turnaround time) je:**

- Vreme koje protekne od aktiviranja do kompletiranja procesa

**14. Niti se uvode radi:**

- Konkurentnog izvršavanja delova posla dodeljenog procesoru

**15. Proces prelazi iz stanja Izvršava se u stanje Završen:**

- Kada je proces završio izvršavanje ili je prekinut

**16. Dok je jedan proces u monitoru, svi drugi procesi koji pozivaju monitor se:**

- Blokiraju

**17. Bankarov algoritam se koristi za:**

- Proverava da li dodela resursa vodi u bezbedno stanje

**18. Kod problema proizvođač/potrošač postoji ograničenje:**

- U datom trenutku samo jedan agent (Producer / Consumer) može pristupiti buffer-u

**19. Ako postoji kružno čekanje procesa, tada:**

- Postoji uzajamno blokiranje ako su ispunjena 3 potrebna uslova za uzajamno blokiranje

**20. Monitor je:**

- Konstrukcija programskog jezika.

**21. Semafori se mogu koristiti za :**

- Uzajamno isključivanje i sinhronizaciju procesa

**22. Proces u uzajamnom blokiranju:**

- Ne može da oslobodi resurse

**23. Prilikom blokiranja niti vrednost programskog brojača čuva se u okviru:**

- Upravljačkog bloka niti

**24. Operativni sistem mora da ispreplete izvršavanje više procesa iz sledećih razloga:**

- Da vi ostvario maksimalno iskorišćenje procesora, a da istovremeno ima odgovarajuće vreme odziva

**25. Kod implementacije niti na nivou korisnika (ULT):**

- Prebacivanje konteksta niti ne zahteva prelaza u kernel mode.

**26. Proces kome je dodeljen procesor, nalazi se u stanju:**

- Izvršava se

**27. Svaka nit može:**

- Pristupiti globalnim promenljivima procesa

28. **Nit je:**

- Entite koji se raspoređuje i izvršava na procesoru.

29. **Promena (switching) mode-a je pojam koji se odnosi na :**

- Prebacivanje procesora iz mode-a kernel u mode korisnik

30. **Svaka nit ima?**

- Sopstveni magacin

31. **Multiprogramiranje je:**

- Kada u memoriji računara može istovremeno biti više procesa koji se vrlo brzo smenjuju u izvršavanju

32. **Proces je u stanju Suspendovan kada:**

- Prebaci se na disk, da bi se oslobodila glavna memorija

33. **U savremenim operativnim sistemima poput Windows 7, svaki proces ima:**

- Više niti izvršenja

34. **Algoritam planiranja u UNIX-u ne koristi**

- FCFS princip

35. **Srednjoročni planer određuje:**

- Koje suspendovane procese treba prebaciti sa diska u memoriju

36. **Proces ne sadrži:**

- Tabelu procesa

37. **Kreiranje niti je brže od kreiranja procesa, jer se:**

- Za niti ne vezuju resursi

38. **Svaka nit procesa sadrži:**

- Upravljački blok niti

39. **Koji operativni sistem nije klon UNIX-a?**

- Symbian (System V, BSD, Solaris JESU KLONOVI)

40. **Koje informacije ne sadrži PCB procesa ?**

- Informacije o sistemskim pozivima

41. **Proces ne sadrži :**

- Tabelu procesa

42. **Proces prelazi iz stanja Blokiran u stanje Spreman:**

- Kada se pojavi događaj na koji je čekao

**43. Kreiranjem procesa kreira se:**

- Osnovna nit procesa

**44. Na jednoprocesorskom računaru konkurentni procesi:**

- Mogu se preplitati u vremenu

**45. Dok je jedna proces u monitoru, svi drugi procesi koji pozivaju monitor se:**

- BLOKIRAJU

**46. Koji se problemi javljaju kod rešenja uzajamnog isključivanja korišćenjem zabrane prekida (HW rešenje)**

- Ne radi na multiprocesorima

**47. U indirektne metode za prevenciju uzajamnog blokiranja ne spada**

- Sprečavanje pojave kružnog čekanja

**48. Proce prelazi iz stanja Novi u stanje Spreman:**

- Kada ima dovoljno resursa za njegovo aktiviranje

**49. Proces je u stanju Suspendovan kada?**

- Kada se prebaci na disk da bi se oslobodila glavna memorija

**50. Proces prelazi iz stanja Spreman u stanje Izvršava se:**

- Kada je izabran od strane OS-a za izvršavanje na CPU.

**51. Binarni semafor može uzimati vrednosti:**

- 0 i 1

**52. Prevencija uzajamnog blokiranja:**

- Vodi neefikasnom korišćenju resursa i izvršavanju procesa

**53. Deadlock je izraz za:**

- Uzajamno blokiranje procesa (samrtni zagrljaj)

**54. U prenosu poruke nije moguće**

- Blokirajući send, neblokirajući receive

**55. Problem proizvođač/potrošač sa ograničenim baferom se može rešiti upotrebom**

- Dva brojačka i jednim binarnim semaforom (I THINK)

**56. Operacija cWait(C) na uslovnoj promenljivoj C monitora:**

- Blokira proces koji je pozvao ovu operaciju i proces smešta u red procesa koji čekaju da ponovo uđu u monitor kada se uslovi promene

**57. U datom trenutku u monitoru se može izvršavati**

- Samo jedan proces

**58. Ako postoji više instanci resursa u sistemu, ciklus u grafu dodele resursa označava da**

- Može, ali ne mora da postoji uzajamno blokiranje

**59. Prilikom kreiranja procesa, operativni sistem:**

- Kreira PCB procesa i unosi ga u tabelu procesa

**60. Kako procesor zna u kom modu radi**

- Na osnovu vrednosti odgovarajućeg bita u registru statusne reči programa PSW

**61. Kratkoročni planer određuje**

- Koji od spremnih procesa će se sledeći izvršiti

**62. Algoritam planiranja u UNIX-u ne koristi**

- FCFS

**63. Kod planiranja više redova sa povratnom spregom (Feedback) proces kome istekne vremenski kvant prebacuje se:**

- Na kraj reda nižeg prioriteta

**64. Kod implementacije niti na nivou korisnika, moguće je :**

- Da se višenitni program izvršava na bilo kom OS na kome postoji biblioteka niti

**65. Promena moda obavezno povlači promenu procesa**

- NE.

**66. Prevencija uzajamnog blokiranja:**

- Vodi neefikasnom korišćenju resursa i izvršenju procesa

**67. Procesi koji se nalaze u redu koji je pridružen semaforu su:**

- Blokirani

**68. U indirektne metode za prevenciju uzajamno blokiranja ne spada:**

- Sprečavanje pojave kružnog čekanja

**69. Rad u podeli vremena (timesharing):**

- Omogućava rad više korisnika čiji procesi se obavljaju multiprogramski.

**70. Dispečer je deo operativnog sistema koji ima zadatak da:**

- Izvrši promenu (switching) procesa.

**71. Upotrebom mašinske instrukcije za testiranje i postavljanje (testset) može se rešiti uzajamno isključivanje procesa:**

- Da.

**72. Graf dodele resursa ne sadrži:**

- Stanja procesa.

**73. Stanje sistema je bezbedno:**

- Ako operativni sistem može dodeliti resurse svakom procesu po određenom redosledu bez nastanka uzajamnog blokiranja.