

Operációs rendszerek vizsga kérdések
válaszokkal
(ELTE-IK Prog.Terv.Mat 2005)

Témakörök :

1. Alapfogalmak
2. Folyamatok
3. Párhuzamosság
4. Memóriakezelés
5. Állományrendszerek

1. Alapfogalmak

Mi válthat ki a kivételt?

Válasz: Valamely utasítás végrehajtása.

Milyen követelményt támasztunk a nagygépek operációs rendszereivel szemben?

Válasz: Teljes körű hibakezelés.

Mi válthat ki a megszakítást?

Válasz: Külső eszköz be- és kiviteli tevékenysége.

Mi a firmware?

Válasz: Hardveregységbe épített szoftver.

Hol folytatódik a végrehajtás vetelés típusú kivételből való visszatéréskor?

Válasz: Nem folytatható a végrehajtás.

Melyik rendszer az, amelyik nem fut mobil eszközökön?

Válasz: IBM z/OS, IBM VSE/ESA.

Mi nem tartozik a tágabb értelemben vett biztonság (safety) fogalmához?

Válasz: Rugalmasság, Hatékonyság (performance, efficiency), Hordozhatóság (portability), Könnyű kezelhetőség (user friendliness)

Mi az az overhead?

Válasz: A rendszerszoftver erőforrásfelhasználása.

Mi a middleware?

Válasz : Az operációs szolgáltatásait kibővítő rutinyűjtemény.

Hol folytatódik a végrehajtás csapda típusú kivételből való visszatéréskor?

Válasz: A kivételt kiváltó utasítást követő utasítástól

Hol folytatódik a végrehajtás hiba típusú kivételből való visszatéréskor?

Válasz: A kivételt kiváltó utasítás újratekintésével

Mi tartozik a tágabb értelemben vett biztonság (safety) fogalmához?

Válasz: Biztonság (security), hibatűrés (fault tolerance)

Mit definiál a POSIX szabvány?

Válasz: Nyílt rendszerek alkalmazásprogramozói felületét.

Melyik operációs rendszer objektum-orientált felépítésű?

Válasz: IBM OS/400.

Melyik rendszer az, amelyik fut mobil eszközökön?

Válasz: Microsoft Windows CE ,Palm OS , Symbian OS

2.Folyamatok

Öt programot indítunk el egyszerre. Becsült futásidejük: 20, 60, 30, 10, 50 s. Az indulástól számított hányadik másodpercben ér véget a 2. folyamat futása, ha az előző sorrendet tekintve Round-Robin ütemezéssel futtatjuk őket?

Legyen egy időszak hossza 1 s, továbbá eltekinthetünk a folyamatok közti átkapcsolási időtől.

Válasz: 87.

Öt köteget program érkezik a számítóközpontba. Becsült futásidejük: 20, 60, 30, 10, 50 perc. Mennyi az átlagos áthaladási idő, ha a legrövidebbet először algoritmussal futtatjuk őket?

Válasz: 76 perc.

Melyik algoritmus nem szakítja meg futás közben a folyamatot?

Válasz: Legrövidebbet először ütemezés.

Mely folyamatok prioritását változtatja dinamikusan az OpenVMS ütemezője?

Válasz : Interaktív folyamatokét.

Mikor emeli az OS/400 egy folyamat prioritását?

Válasz: Ha olyan erőforrást használ, amelyre egy magasabb prioritású folyamat vár.

Melyik folyamattípust támogatja az OpenVMS?

Válasz: Interaktív, köteget, valós idejű.

Melyik algoritmus futtatja a folyamatokat érkezési sorrendben?

Válasz: FIFO ütemezés.

Melyik ütemezésben nem játszanak szerepet prioritások?

Válasz: Round Robin.

Melyik állítás igaz a Round Robin ütemezésre?

Válasz: Minden folyamatnak egyforma időszelet jut.

Öt programot indítunk el egyszerre. Becsült futásidejük: 20, 60, 30, 10, 50 s. Mennyi az átlagos áthaladási idő, ha az előző sorrendet tekintve Round-Robin ütemezéssel futtadjuk őket? Legyen egy időszelet hossza 1 s, továbbá eltekinthetünk a folyamatok közti átkapcsolási időtől.

Válasz: 117s.

Ha a Round Robin ütemezésben az időszeletet végtelenre választjuk, akkor...

Válasz: ...akkor a FIFO ütemezést kapjuk.

Futó állapotba kerül egy futó folyamat, ha...

Válasz: ...központi egységre kerül.

Öt kötegelt program érkezik a számítóközpontba. Becsült futásidejük (érkezési sorrendben): 20, 60, 30, 10, 50 perc. Mennyi az átlagos áthaladási idő, ha érkezési sorrendben futtatjuk őket?

Válasz: 100 perc.

Öt kötegelt program érkezik a számítóközpontba. Becsült futásidejük: 20, 60, 30, 10, 50 perc. A prioritásuk: 1, 3, 2, 5, 4. A legnagyobb prioritás az 5. Mennyi az átlagos áthaladási idő, ha prioritásos ütemezéssel futtadjuk őket?

Válasz: 102 perc.

Mi a zombi folyamat Linuxban?

Válasz: Befejeződött folyamat, ami a folyamatáblázatban ragadt.

Mit tesz a Windows azzal a folyamattal, amely nagyon régóta vár a processzorra?

Válasz: Megemeli a prioritását két időszelet erejéig.

Melyik ütemezés az, amely a folyamatokat egy sor adatszerkezetben tárolja, mindig a sor első elemét futtatja egy meghatározott ideig, majd a sor végére pakolja?

Válasz : Round Robin ütemezés.

Az $a=1/2$ paraméteres öregedési algoritmust használjuk a futási idők megbecsülésére. Az előző négy futás, a legrégebbivel kezdve 40, 24, 40, 20 ms volt. Mennyi a becslés a következő futásra?

Válasz: 28ms.

Melyik ütemezést használjuk valós idejű rendszerekben?

Válasz: Legkorábbi határidőt először, legkisebb lazaság.

Blokkolt állapotba kerül egy futó folyamat, ha...

Válasz: ...központi egységtől különböző erőforrásra vár.

Milyen ütemezési stratégiát alkalmaz valós idejű folyamatokra a Linux?

Válasz: Round Robin vagy FIFO választható

Milyen ütemezési stratégiát alkalmaz interaktív folyamatokra a Linux?

Válasz: Round Robin .

3.Párhuzamosság

A folyamatváltások letiltása...

Válasz: ...látszat párhuzamosság esetén hatásos módszer a kritikus szekciók kezelésére.

A kölcsönös kizárás feltétele előírja, hogy...

Válasz: ...egyszerre maximum egy folyamat legyen kritikus szekcióban.

A haladás feltétele előírja, hogy...

Válasz: a következő alkalommal a kritikus szekcióba lépő folyamat azok közül kerüljön kiválasztásra, akik az adott pillanatban belépésre várnak.

Igaz állítások :

- Az operációs rendszer maga is gyakran használ szemaforokat a benne felmerülő szinkronizációs problémák megoldására.
- A szemafor implementációjától függ, hogy a down művelet (ha szükséges) tevékenyen várakozik-e vagy sem.
- A szemafor megengedi, hogy egyszerre több folyamat várakozzék a down művelet végrehajtása közben.
- Az erőforrás-szükségletek előzetes ismeretének birtokában ügyes szervezéssel elkerülhető a holtponatok kialakulása.
- Egynél több processzor ugyanazon számítógépbe tétele nem növeli a rendszer megbízhatóságát.
- A többprocesszoros rendszerek egyik ismérve, hogy a processzorok ugyanazt az órajelet használják.
- A fűrtözés (klaszterezés) segítségével növelhetjük a rendszerünk megbízhatóságát.
- A spinlockok tevékenyen várakoznak.
- Az olvasók-író problémában a védett objektum nem olvasható miközben az író folyamat változtatja.

Ha egy folyamattól az előzőleg engedélyezett erőforrás nem vehető el semmilyen módon,...

Válasz: ...akkor holtponat alakulhat ki.

A holtponatok létejöttéhez nem szükséges, hogy...

Válasz: ...egyszerre legfeljebb egy folyamat legyen kritikus szekcióban.

Az üzenetküldés...

Válasz: ...különböző számítógépeken futó folyamatok kommunikációjának természetes módja.

Peterson megoldása a kritikus szekció problémájára...

Válasz: ...nem működik három folyamat esetén.

A rendszer állapota holtponat-elkerülés tekintetében biztonságos, ha...

Válasz: ...a folyamatokat valamilyen sorrendben be lehet úgy fejezni, hogy eközben nem áll elő holtponat.

A kritikus szekciók problémáját megoldó módszerektől elvárjuk az alábbi feltételek teljesülését:

Válasz: Kölcsönös kizárás, haladás, korlátozott várakozás.

A tevékeny várakozás...

Válasz: csak rövid ideig megengedhető, mivel feleslegesen pazarolja a processzort.

A feladat egy objektum konzisztenciájának védelme a félkész módosítások kiolvasása előtt. Melyik klasszikus problémáról van szó?

Válasz: Író-olvasó probléma.

A szigorú váltogatás módszere...

Válasz: megsérti a haladás feltételét, ezért nem jó megoldás a kritikus szekciók problémájára.

A versenyhelyzet...

Válasz: felderítéséhez, megértéséhez előfordul, hogy behatóan, alacsony szinten is ismernünk kell a rendszerünket.

A sorszámhúzásos szinkronizáció...

Válasz: belépési sorszámok kiosztásával biztosítja a korlátozott várakozás feltételének teljesülését.

A szemafor...

Válasz: up művelete várakozás nélkül eggyel növeli a szemafor értékét.

A többprocesszoros rendszerek...

Válasz: általában nem tudnak zavartalanul tovább működni, ha az egyik processzor váratlan meghibásodásodik.

Melyik a legkevésbé műveletigényes holtpontkezelő eljárás?

Válasz: Strucc algoritmus.

Melyik a leginkább műveletigényes holtpontkezelő eljárás?

Válasz: Bankár algoritmus.

A sorszámhúzásos szinkronizáció...

Válasz : előfordul, hogy tevékenyen várakozik.

Adatcsere történik egy közös tárolón keresztül. Melyik klasszikus problémáról van szó?

Válasz: Termelő-fogyasztó probléma .

Azt mondjuk, hogy egy folyamathalmaz holtpontban van, ha...

Válasz: minden eleme a halmaz valamelyik másik eleme által kiváltható eseményre várakozik.

A korlátozott várakozás feltétele...

Válasz: ...felső korlátot ad arra, hogy a kritikus szekcióba történő belépés előtt hány más folyamatot kell előreengednünk.

A monitorok...

Válasz:

- ...műveleteit egyszerre csak egy folyamat hajthatja végre.
- ...kényelmesen alkalmazhatók osztott változók versenyhelyzet-mentes kezelésére.
- ...osztott memórián alapuló rendszerek nyelvi szintű szinkronizációs eszköze.

Kritikus szakaszban fut egy folyamat, ha...

Válasz : a folyamat egy más folyamatokkal együtt használt, közös erőforrást használ.

A test and set lock (TSL) gépi utasítás...

Válasz: a zár értékét igazra állítja, és visszaadja a zár korábbi értékét.

A monitorállapotok...

Válasz: signal művelete előfordul, hogy lényegében nem csinál semmit.

4. Memóriakezelés

Egy dinamikus partíciókat használó rendszer a kezdőcím szerint rendezve a következő lyukakat tartalmazza: I:12KB, II:6KB, III:5KB, IV:10KB, V:9KB, VI:1KB. Melyik lyukat választja egy 5KB-os foglalásra a "best fit"?

Válasz: III.

Ha a virtuális címtér 64KB és a fizikai Memóriakezelés 32KB akkor melyik eset fordulhat elő az alábbiak közül?

Válasz: 16 virtuális lap és 8 lapkeret van.

Hogyan lehet egy programot könnyen és hatékonyan betölteni különböző kezdőcímekre, virtuális Memóriakezelés nélkül?

Válasz: Bázisregiszter használatával.

Egy dinamikus partíciókat használó rendszer a kezdőcím szerint rendezve a következő lyukakat tartalmazza: I:12KB, II:6KB, III:5KB, IV:10KB, V:9KB, VI:1KB. Melyik lyukat választja egy 5KB-os foglalásra a "next fit"?

Válasz : Bármelyik előfordulhat a megadott válaszok közül.

Mit nevezünk szegmensnek?

Válasz: Az egymástól független lineáris címtereket.

Egy fizikai memóriában négy lapkeret (L) van. A lapok betöltésének (B) és utolsó elérésének (U) idejét valamint az A és D bitek értéke a következő L;B;U;A;D sorrendben: (0;140;220;0;0), (1;210;280;1;0), (2;100;270;1;1), (3;180;200;1;1). Melyik lapot cseréli a "második lehetőség" algoritmus?

Válasz: 0.

Mikor történik laphiba?

Válasz: Akkor amikor olyan lapra hivatkozunk, amelyhez nem tartozik lapkeret a fizikai memóriában.

Milyen típusú kivétel a laphiba?

Válasz: Hiba.

Mi történik, ha az igényelt munkahalmaz mérete nagyobb, mint az aktuálisé?

Válasz: A folyamat vergődik a sok laphibától.

Mit nevezünk Bélády-anomáliának?

Válasz: Amikor a FIFO algoritmus használata során több lapkerettel több laphiba keletkezik.

Igaz állítás(ok):

A virtuális Memóriakezelés mérete legalább a fizikai Memóriakezelés méretével egyezik meg, vagy annál nagyobb.

Sorszámozzuk a Memóriakezelés lapokat 0,...,9 számokkal. Tegyük fel, hogy a memóriában három lap fér el. Jelenleg a 0,4,7 sorszámú lapok vannak a memóriában, és a következő sorrendben hivatkozunk a lapokra:

1,2,3,4,1,2,5,1,2,3,4,5. Hány lapcsere történik a FIFO lapcserélési algoritmus használata esetén?

Válasz: 9.

Sorszámozzuk a Memóriakezelés lapokat 0,...,9 számokkal. Tegyük fel, hogy a memóriában három lap fér el. Jelenleg a 0,4,7 sorszámú lapok vannak a memóriában 1-es R bittel, és a következő sorrendben hivatkozunk a lapokra: 1,2,3,4,2,5,2,3,1. Hány lapcsere történik a második lehetőség lapcserélési algoritmus használata esetén?

Válasz: 7.

Egy dinamikus partíciókat használó rendszer a kezdőcím szerint rendezve a következő lyukakat tartalmazza: I:12KB, II:6KB, III:5KB, IV:10KB, V:9KB, VI:1KB. Melyik lyukat választja egy 5KB-os foglalásra a "first fit"?

Válasz: I.

Egy dinamikus partíciókat használó rendszer a kezdőcím szerint rendezve a következő lyukakat tartalmazza: I:12KB, II:6KB, III:5KB, IV:10KB, V:9KB, VI:1KB. Melyik lyukat választja egy 5KB-os foglalásra a "worst fit"?

Válasz: I.

Egy fizikai memóriában négy lapkeret (L) van. A lapok betöltésének (B) és utolsó elérésének (U) idejét valamint az A és D bitek értéke a következő L;B;U;A;D sorrendben: (0;140;220;0;0), (1;210;280;1;0), (2;100;270;1;1), (3;180;200;1;1). Melyik lapot cseréli az LRU algoritmus?

Válasz: 3.

Egy fizikai memóriában négy lapkeret (L) van. A lapok betöltésének (B) és utolsó elérésének (U) idejét valamint az A és D bitek értéke a következő L;B;U;A;D sorrendben: (0;140;220;0;0), (1;210;280;1;0), (2;100;270;1;1), (3;180;200;1;1). Melyik lapot cseréli a FIFO algoritmus?

Válasz : 2.

Sorszámozzuk a Memóriakezelés lapokat 0,...,9 számokkal. Tegyük fel, hogy a memóriában négy lap fér el. Jelenleg a 0,4,7,9 sorszámú lapok vannak a memóriában, és a következő sorrendben hivatkozunk a lapokra: 1,2,3,4,1,2,5,1,2,3,4,5. Hány lapcsere történik a FIFO lapcserélési algoritmus használata esetén?

Válasz: 10.

Sorszámozzuk a Memóriakezelés lapokat 0,...,9 számokkal. Tegyük fel, hogy a memóriában három lap fér el. Jelenleg a 0,4,7 sorszámú lapok vannak a memóriában, és a következő sorrendben hivatkozunk a lapokra: 1,2,3,2,4,1,3,2,1,4. Hány lapcsere történik a LRU lapcserélési algoritmus használata esetén?

Válasz: 8.

Mi történik, ha az igényelt munkahalmaz mérete kisebb, mint az aktuálisé?

Válasz: Pazarlunk, mert olyan lapok is benn vannak a folyamat aktuális munkahalmazában, amit már nem használ.

5.Állományrendszerek

Igaz állítások:

- A naplózott fájlrendszer feleslegessé teszi a váratlan leállások utáni idő- és erőforrásigényes konzisztencia-ellenőrzést.
- A napló írásakor a tanult I/O ütemező algoritmusok általában nem használhatók.
- A naplózott fájlrendszerek az esetek többségében kevésbé hatékonyak, mint a naplózatlanok.
- A fájlrendszerek általában szigorú faszerkezetű könyvtárrendszert használnak.
- A UNIX operációs rendszer a fájlok nevét a könyvtár részének tekinti, és megkülönbözteti a fájl egyéb metaadataitól.
- Az NTFS fájlrendszerben a fájlok a tulajdonképpeni adattartalom mellett tetszőleges számú "extra" bájt sorozatot is tartalmazhatnak.
- Előfordul, hogy a kombinált RAID megoldások (RAID 1+0, 0+1) akár egyszerre több diszk kiesését is elviselik.
- A RAID 6 szint a paritásblokkok mellett hibajavító blokkokat is eltárol.
- A RAID 2 szintet gyakorlatilag sohasem használják.
- A lift ütemezést különösen nagy I/O terhelésű rendszereken célszerű használni.
- A lusta ütemezés hajlamos egyes kéréseket kiéheztetni.
- Ha csak egyetlen kérés várakozik teljesítésre, akkor a sorrendi és a lusta I/O ütemezés ugyanazt az eredményt adja.
- Adattömörítéssel a processzor kárára javíthatunk a lassú bevitel/kivitelű rendszerünk teljesítményén.

A bevitel/kivitel ütemező legfontosabb célja...

Válasz: ...az író/olvasó parancsok sorrendjének megfelelő megválasztásával csökkenteni a fejmozgások által okozott várakozási időt.

Legyen a blokkméret 1 KB. Egy 400 MB kapacitású lemez bittérképe hány blokkot foglal el?

Válasz: 50.

Az ext2 fájlrendszer...

Válasz:

- megpróbálja csökkenteni a fájl-töredezettségből eredő káros hatásokat.
- a fájlok adatblokkjait igyekszik a meta-adatok közelében tárolni.

Mely RAID szintek ismeretét várhatjuk el leginkább egy fejlett adattároló megoldástól?

Válasz: RAID 0, RAID 1, RAID 5.

A bittérképeket a fájlrendszerek implementációjában...

Válasz: sok esetben használják a szabad blokkok nyilvántartására.

Miért káros a fájlok töredezettsége?

Válasz: Mert a töredékesen, a lemezen szétszórva elhelyezkedő fájlok csökkentik a rendszer teljesítményét.

Az indexelt blokkfoglalás...

Válasz: nem ad megoldást a belső töredezettség problémájára.

Miért káros a belső töredezettség?

Válasz: Mert a fájlok belsejében (tipikusan a fájl végén) töredékesen feltöltött blokkok kihasználatlan tárterületet, így kapacitásvesztést eredményeznek.

Miért káros a külső töredezettség?

Válasz: Mert a szabad területek elaprózódása ellehetetlenítheti az új összefüggő fájlok létrehozását.

A lemezegységen az adatokat fizikailag...

Válasz: ... cilinderekre, azon belül sávokra, azon belül azonos méretű szektorokra osztva helyezük el.

A folytonos blokkfoglalás...

Válasz: esetén a fájlok beolvasása és a már lefoglalt, fájlban belüli adatblokkok felülírása különösen hatékonyan elvégezhető.

Melyik módszer esetén fordulhat elő, hogy a háttértárolón összeségében lenne elegendő hely, mégis helyhiány miatt nem tudjuk az állományt elhelyezni?

Válasz: folytonos helyfoglalás.

A láncolt listás blokkfoglalás...

Válasz: ...csak szekvenciális fájlkezelésre alkalmas, ha a következő blokk számát az adatblokkokban tároljuk.

Legyen a blokkméret 1 KB, továbbá tegyük fel, hogy a blokkok címe elfér 32 biten. Egy 100 MB kapacitású lemez szabad blokkjainak felsorolása legfeljebb hány blokkot foglal el, ha a blokkok utolsó négy bájtján a lista következő blokkjának sorszámát tároljuk

Válasz:402.

Melyik logikai fájl szerkezettel dolgozik a népszerű operációs rendszerek többsége?

Válasz: A fájl bájtok sorozata.

Egy merevlemez a következő paraméterekkel rendelkezik: egy sáv kapacitása 4096 Byte, a körülfordulási idő 20 ms, az átlagos pozicionálási idő 30 ms.

Mennyi 512 Byte méretű szektor beolvasásának az átlagos ideje? (A pozicionálási idő az I/O parancs kiadásától a legelső bájtt beolvasásáig eltelt idő.)

Válasz: 32,5 ms.

Ha egy diszk meghibásodik, a RAID 5 tömb...

Válasz:

- sebessége az adatok teljes helyreállításáig jelentősen csökken.
- a hibás diszk kicserélése után egy darabig még nem lesz képes egy másik diszk kiesését elviselni.

Egy merevlemez a következő paraméterekkel rendelkezik: egy sáv kapacitása 4096 Byte, a körülfordulási idő 20 ms, az átlagos pozicionálási idő 30 ms.

Mennyi 512 Byte méretű szektor beolvasásának az átlagos ideje? (A pozicionálási idő az I/O parancs kiadásától a legelső bájtt beolvasásáig eltelt idő.)

Válasz: 30 ms.

A logikai formázás közben...

Válasz:...a lemezen felépítjük a fájlrendszer működéséhez szükséges adatszerkezeteket.

