

# Elméleti zárthelyi dolgozat 2021.dec.1.

**Határidő** Nincs megadva határidő**Pont** 20**Kérdések** 20**Elérhető** 2021. dec 1, 16:00 - 2021. dec 1, 16:40 40 perc**Időkorlát** 20 perc

## Instrukciók

Kérem töltsék ki a feladatlapot!

Eredmények: 10-12 pont kettes, 13-14 pont hármás, 15-17 pont négyes, 18-tól ötös.

Az elméleti pótzh december 15-én lesz, 16 órakor. A pótzh során csak javítani lehet az első ZH eredményén.

Ezt a kvízt ekkor zárolták: 2021. dec 1, 16:40 .

## Próbálkozások naplója

	Próbálkozás	Idő	Eredmény
LEGUTOLSÓ	<a href="#">1. próbálkozás</a>	20 perc	12 az összesen elérhető 20 pontból

❗ A helyes válaszok el vannak rejtve.

Ezen kvíz eredménye: **12** az összesen elérhető 20 pontból

Beadva ekkor: 2021. dec 1, 16:20

Ez a próbálkozás ennyi időt vett igénybe: 20 perc

### 1. kérdés

**1 / 1 pont**

Mit biztosít, mit jelent, hogy a CPU üzemmódok bővültek, és megjelent például a "kernel" és "user" mód?



Ez a lehetőség azt jelenti, hogy az operációs rendszer tud a hatalmával élni és "teljhatalmú úr" a folyamatok fölött.



Olyan lehetőséget ad, amivel jobban igazítható egy-egy alkalmazástípushoz a rendszerünk.



Olyan lehetőséget ad, amivel mi is készíthetünk Operációs rendszert!



Ez a lehetőség azt jelenti, hogy az operációs rendszer nem tud a hatalmával élni és nem lehet "önkényúr" a folyamatok fölött.

## 2. kérdés

1 / 1 pont

Még ma is a legfontosabb háttértárunk a mágneses merevlemez. Mit jelent az, hogy a lemezen definiált sávok-szektorok metszésében elhelyezkedő blokk mérete 512 bájt?



Ez nem jelent semmit, a klaszter és a forgási sebesség számít.



Azt, hogy a legkisebb lemezfoglalási egység csak ennyi lehet.



Ez az operációs rendszertől függ, tehát más is lehet.



Meghatározza lemezfoglaltsági térkép szerepét, ha ez van, akkor nem kell ilyen.

## 3. kérdés

1 / 1 pont

Mi nem jellemzője a RAID 1 tömbnek?



2 lemez kell a tömbbe.



1 lemez meghibásodása esetén nyújt biztonságot.

☒ Dedikált paritásdiszk használat.

☐ Redundáns adattárolást biztosít.

Helytelen

#### 4. kérdés

0 / 1 pont

Mit jelent a titkosított fájlrendszer fogalma?

☐

A titkosítás azt biztosítja, hogy a fájlrendszer tartalmát kívülről egy független gépen ne lehessen olvasni.

☒

A titkosítás azt biztosítja egy multi-user rendszerben, hogy a tulajdonában lévő állományokat, könyvtárakat csak az adott felhasználó láthassa.

☐

A titkosítás azt biztosítja egy multi-user rendszerben, hogy a fájlrendszer tartalmát csak az adott felhasználó láthassa.

☐

A titkosítás azt jelenti, hogy csak az adminisztrátorok látják azt a területet.

#### 5. kérdés

1 / 1 pont

A preemptív operációs rendszer modell "mit előz meg, kitől kit véd meg"?

☐

A modellben bizonyos idő után megkéri az alkalmazást, hogy mondjon le a CPU-ról, így megelőzi a rendszer kisajátítását, összeomlását.



Nem előz meg semmit, hiszen a mai rendszerekben is találkozunk rendszerhibákkal.



A határidők megjelenése arra ösztönzi az alkalmazásokat, hogy gyorsan fejezzék be a válaszadásokat, így biztosítva a rendszert.



A modellben bizonyos idő után elveszi a rendszer a vezérlést az alkalmazástól, így megelőzi a rendszer kisajátítását, összeomlását.

## 6. kérdés

1 / 1 pont

A Peterson féle kölcsönös kizárás használatánál mi a probléma az alábbi kódrészlet piros sorai felcserélése során?

```
void belepes(int proc)
```

```
{int masik;
```

```
    masik=1-proc; //mivel N=2...
```

```
    akarja[proc]=1; //processz futni akar
```

```
    kovetkezo=proc;
```

```
    while( kovetkezo==proc &&
```

```
        akarja[masik]);
```

```
}
```



Ha rosszkor vált az ütemező "egymásnak eshetnek a filozófusok", többen is kritikus szakaszba kerülhetnek.



Ha a sorok után vált az ütemező, az okozhat problémát.



Ha a sorok előtt vált az ütemező az problémát okozhat.



Ha bármikor vált az ütemező "éhen nem halhatnak a filozófusok", nincs probléma.

## 7. kérdés

1 / 1 pont

Hogyan biztosítja a szemafor segítségével a jó kölcsönös kizárást az operációs rendszert?



Azzal biztosítja a jó kölcsönös kizárást, hogy ezen a "változón" végzett művelet oszthatatlan, nem lehet megszakítani.



Azzal biztosítja a jó kölcsönös kizárást, hogy ezen a "változó" nem igényel közös memóriát.



Posix ajánlás szerint a "volatile" jelzővel biztosítjuk a jó kölcsönös kizárást.



Azzal biztosítja a jó kölcsönös kizárást, hogy ezen a "változón" végzett művelet felhasználó szinten valósul meg.

Helytelen

## 8. kérdés

0 / 1 pont

Melyik állítás nem igaz a kölcsönös kizárást biztosító monitorra?



A monitor megvalósítás egy szemaforon alapuló biztonságosabb kölcsönös kizárást biztosító objektum.



A monitor eljárásban a condition "változókon" lévő wait, signal művelettel biztosítják, hogy ne blokkolódjanak a folyamatok.



A monitor egy operációs rendszer nyújtotta egyszerűbb, biztonságosabb kölcsönös kizárást biztosító eszköz.



A monitor a fejlesztő környezetek nyújtotta egyszerűbb, biztonságosabb kölcsönös kizárást biztosító eszköz.

### 9. kérdés

1 / 1 pont

Mi az üzenetküldéssel biztosított kölcsönös kizárás fő előnye?



Közös (rendszer) memória esetén nincs lehetőség a helyes kölcsönös kizárás biztosítására.



Nem szükséges közös (rendszer) memória a helyes kölcsönös kizárás biztosításához.



Még a monitornál is könnyebb használat.



Az, hogy a hálózati kommunikációhoz hasonló elvet követ.

Helytelen

### 10. kérdés

0 / 1 pont

Melyik állítás nem igaz az Író-Olvásó problémára?



Egyszerre egy író lehet az adatbázisban, amit a database szemafor biztosít.

Az alábbi részlet az író magányát biztosítja.

```
down(database); // kritikus
```

```
írunk_adatbázisba();
```

☒

```
up(database);
```

☐

Az olvasók számlálójának elérését és az író kölcsönös kizárását az alábbi részlet biztosítja.

```
down(mutex);
```

```
rc++;
```

```
if (rc==1) down(database);
```

```
up(mutex);
```

Az írók számlálójának kölcsönös kizárását az alábbi részlet biztosítja.

```
down(mutex);
```

```
rc++;
```

```
if (rc==1) down(database);
```

☐

```
up(mutex);
```

Helytelen

## 11. kérdés

0 / 1 pont

Jelölje meg mi nem a Sorsjáték ütemezés jellemzője?

☐

Az időintervallum lejártakor egy új sorsjegyet kap a folyamat és legközelebb rövidebb ideig futhat.

☐

Sorsjáték szerűen a kiosztott jegyek alapján választunk egy folyamatot.

☐

Hasonlít a Garantált ütemezéshez, csak a CPU idő számolás más elvek alapján történik.



Prioritások ütemezése, ami azt jelenti, hogy ennek megfelelően osztja szét a folyamatok között a sorsjegyeket.

**Helytelen****12. kérdés****0 / 1 pont**

A mai Windows rendszerekben használt prioritások ütemezője milyen módszert használ a kiéheztetés elkerülésére?



A Windows az Arányos ütemező elvét követve kerül el a kiéheztetést.



Az alacsony prioritású folyamatok ideiglenesen magas prioritási osztályba kerülnek, majd ott CPU időt kapnak, az idő letelte után vagy terminálnak vagy visszakerülnek az eredeti osztályba.



Az alacsony prioritású folyamatok egy idő után magas prioritási osztályba kerülnek, majd ott CPU időt kapnak, az idő letelte után vagy terminálnak vagy maradnak az eredeti osztályban.



A Windows a strucc algoritmust használva kerül el a kiéheztetést.

**Helytelen****13. kérdés****0 / 1 pont**

Jelölje meg mi nem jellemzője a CFS (Completely Fair Scheduler) ütemezőnek?



Prioritások ütemező Linux környezetben, a kiéheztetés veszélye fennáll.





Olyan prioritásos ütemező, aminél bináris fában (RedBlackTree) tároljuk az elhasznált CPU időket.



Alapértelmezett prioritásos ütemező Linux rendszerekben, Molnár Ingó volt ELTE hallgató nevéhez köthető az ütemezési elv.



A CPU idő "fair" kiosztási elve a Garantált ütemezéshez hasonlít.

#### 14. kérdés

1 / 1 pont

Melyik állítás nem jellemzője az indextábla alapú fájlrendszereknek?



A multiprogramozási környezetben ha szükséges a láncolás is megjelenik a fájlok elhelyezésénél.



Veszteséges tárolási módszer, helyette a folytonos tárolás a preferált.



Nem veszteséges tárolási mód.



Mai változataik támogatják a hozzáférési jogosítványok (pl ACL) használatát..

#### 15. kérdés

1 / 1 pont

Mi a 4. generációs számítógép-operációs rendszerek legfőbb jellemzője az alábbiak közül?



A mágneses háttértár kialakulása és fejlődése.

- ☒ A CPU (LSI áramkörök) megjelenése, fejlődése.
- ☐ Virtuális gépen végzett munka általánossá válása..
- ☐ Nincs ilyen, jelenleg is a harmadik generációs rendszereket használunk.

Helytelen

**16. kérdés****0 / 1 pont**

Mikor nem teljes folyamat mozgatása történik a memória és háttértár között?

- ☐ Multiprogramozás rögzített memória szeletek használata esetén.
- ☐ Soha nincs és nem is volt ilyen.
- ☒ Multiprogramozás memória csere használata esetén.
- ☐ Virtuális memória kezelés esetén, csak extrém esetben lehet ilyen.

**17. kérdés****1 / 1 pont**

Mi a TLB (Translation Lookaside Buffer) buffer?

- ☐ A TLB jelentése Total Load Balancer, a számítógépes klaszterek világában használjuk.
- ☐ Címfordításnál a memóriakezelő az elsődleges laptáblát így is szokta nevezni.
- ☐ Címfordításnál az utolsó directory page táblának a neve.



Címfordításnál a mai rendszerekben ebből a CPU-n belül lévő táblából nézi meg először az MMU, hogy a virtuális címnek mi a fizikai megfelelője.

**18. kérdés****1 / 1 pont**

A mai operációs rendszerekben milyen blokkos eszközhasználat(fájlrendszer) ütemező algoritmus a leginkább preferált?

- ☒ A CFQ(Completely Fair Queuing) algoritmus.
- ☐ A Bankár algoritmus.
- ☐ A FIFO algoritmus.
- ☐ Az Óra lapcsere (Round Robin)

**Helytelen****19. kérdés****0 / 1 pont**

Melyik a legnagyobb prioritású megszakítás kérés?

- ☐ Az NMI, a nem maszkolható interrupt.
- ☐ A szoftveres megszakítás.
- ☐ Az 1-es sorszámú megszakítás.
- ☒ A legnagyobb, az 5-ös sorszámú megszakítás.

**20. kérdés****1 / 1 pont**

### A bankár algoritmusban mit nevezünk biztonságos állapotnak?



Azt amiből még mi tudjuk befolyásolni a folyamatok befejezését a helyes erőforrás (hitel) kihelyezéssel.



Azt amiből még mi tudjuk befolyásolni a folyamatok befejezését a helyes erőforrás (hitel) visszakövetelésével.



Azt amiből már nem tudjuk befolyásolni a folyamatok befejezését a helyes erőforrás (hitel) kihelyezéssel.



Az amikor egy folyamat rendben terminál.

Kvízeredmény: **12** az összesen elérhető 20 pontból