



Név:

Neptun:

Aláírás:

Rendelkezésre álló idő: **80 perc**. Használható segédeszközök: –  
Összpontszám: **40 pont**. Szükséges kiskérdésekből: **7 pont**, összesen: **16 pont**  
Feladattípusok:

T *Teszt*: a helyes állítások bejelölendők (akár több is). Tökéletes válasz: 1 pont, van bejelölt rossz válasz: -1 pont, nincs bejelölt rossz válasz, de nincs minden jó bejelölve: 0 pont

Ö *Összekötős*: A bal oldali felsorolás elemeinek összekötése a jobb oldali felsorolás elemeivel  
Tökéletes válasz: 1 pont, nem tökéletes válasz, vagy a válasz hiánya: 0 pont

K *Kiegészítő*: A kérdésre adott válasz beírása a keretbe, illetve a pontozott vonalra, tömören  
Tökéletes válasz: 2 pont, nem tökéletes válasz, vagy a válasz hiánya: 0 pont

Ha a "T" típusú kérdésekből negatív pontszám jönne ki, a többi pontszámot **nem** csökkentjük.

T:	Ö:
K:	Sz1:
Sz2:	Sz3:
	$\Sigma$

1. Az Arduino PWM kimenetein különböző PWM értékek mellett

T

- Az impulzusok szélessége állandó
- Az időegységenkénti impulzusok száma állandó
- Az impulzusok amplitúdója állandó

2. Mely állítások igazak az SSD alapú adattárolókra?

T

- Az MLC adattárolás esetén az SSD gyorsabban öregszik, mint SLC esetén
- A seek idő és a forgási késleltetés teszi ki az olvasási kérések kiszolgálásának nagy részét
- A lebegő gate-es tranzisztorok öregedése visszafordítható
- A kopás-kiegyenlítés célja a meghajtó adatleolvasási sebességének növelése

3. Mikor hajtható végre egy művelet a vezérlésáramlásos információfeldolgozási modellben?

T

- Amikor odaér a vezérlő token
- Amikor szükség van a művelet eredményére
- Amikor minden operandusa rendelkezésre áll
- Egyik állítás sem igaz

4. Miért előnyös, ha az SSD vezérlő adattömörítést és de-duplikációt is végez?

T

- Mert így kevesebb írás műveletet kell végeznie, ami kevésbé koptatja a meghajtót
- Mert így kevesebb írás műveletet kell végeznie, ami rövidebbé teszi a válaszütemet
- Mert így a névleges kapacitásnál több adatot tud a felhasználó tárolni
- Egyik állítás sem igaz

5. Mely állítások igazak a ZBR adattárolást alkalmazó merevlemezekre?

T

- Minden sávban azonos az adatsűrűség
- A külső sávok felé haladva lassítani kell a lemez forgását
- A lemez pereme felé haladva a sávonkénti szektorok száma nem csökken
- Egyik állítás sem igaz

6. Mely állítások igazak a NAND flash alapú SSD tárolókra? Egy lapra csak akkor lehet írni, ha...

T

- ha az a lap vagy üres, vagy már nem használt adatokat tartalmaz
- ha a blokk törlése óta még senki sem írt arra a lapra
- közvetlenül előtte letöröljük azt az egy lapot, amire írni szeretnénk
- Egyik állítás sem igaz

7. Hogyan lehet egy merevlemez adathordozó rétegeinek egy szakaszán a mágneszettség irányát a kívántnak megfelelően beállítani?

T

- A kérdéses szakaszt rövid időre állandó külső mágneses mezőnek tesszük ki
- A kérdéses szakaszt rövid időre változó külső mágneses mezőnek tesszük ki
- Egyik sem, a mágneses adathordozókon nem lehet a gyártáskor kialakított mágneszettséget megváltoztatni

8. Mely információ-feldolgozási modellek támogatják a feladatban rejlő párhuzamosság automatikus felderítését? T

- A vezérlésáramlásos modell
- Az igényvezérelt modell
- Az adatáramlásos modell

9. Mely állítások igazak a Daisy chain alapú és a PIC alapú interrupt kezelésre? Ö

	Az eszközök prioritása nem változtatható
Daisy chain alapú interrupt kezelés	Nem működik, ha egyidejűleg több eszköz is jelez interruptot
PIC alapú interrupt kezelés	Körbenforgó interrupt kiszolgálás megvalósítható Tetszőlegesen bővíthető

10. A merevlemezek várakozási sorának ütemezői mely késleltetéseket veszik figyelembe a döntéshozás során? Ö

SATF (Shortest Access Time First)	A seek időt
SSTF (Shortest Seek Time First)	Az adatok interfészen való átvitelének idejét A forgási késleltetést

11. Mely állítások igazak a Daisy chain alapú és a párhuzamos arbitrációra? Ö

	Az eszközök sorrendje egyben meghatározza a prioritásukat is
Daisy chain alapú arbitráció	Képes figyelembe venni, hogy egyes perifériák késleltetésérzékenyek lehetnek
Párhuzamos arbitráció	Nem működik, ha egyidejűleg több eszköz is bejelenti igényét a buszra Ha elromlik az arbiter, leáll a rendszer

12. Forgalomszabályozás hiányában mikor és milyen hibák fordulhatnak elő? Ö

Adat egymásrafutás	Küldő gyorsabb, mint a fogadó
Adathiány	Fogadó gyorsabb, mint a küldő

13. Mi az oka annak, hogy nagyon nagy órajel frekvencia mellett a párhuzamos busz kevésbé alkalmas adatátvitelre, mint a soros? K

14. Rajzolja le, milyen részekből áll a merevlemezen a szervó információ! K

15. Mi az elosztott arbitráció előnye a centralizált arbitrációhoz képest?

K

16. Hány tranzisztor kell 12 bit tárolásához

K

SLC flash memória esetén: .....

2 bites MLC flash memória esetén: .....

3 bites TLC flash memória esetén: .....

17. Rajzolja fel az  $X=(A+B+C)*D$  kifejezéshez tartozó precedenciagráfot olyan elemi műveletekből építkezve, melyek csak két változó között végzett alpműveleteket engednek meg!

K

**1. számpélda** Egy merevlemez 3 db kétoldalas lemezt tartalmaz, melyek mindegyikén 100000 sáv található, minden sávban 500 szektorral. A szektorok mérete 500 bájt. ZBR nincs, az adatátviteli interfész sebessége pedig  $100 \cdot 10^6$  bájt/s. A parancsfeldolgozási késleltetés olyan kicsi, hogy nullának tekintjük. Az átlagos seek idő 5 ms, a lemez forgási sebessége 12000 fordulat/perc.

(a) Adjuk meg CHS koordináta rendszerben a merevlemez kapacitását! (1 pont)

C= \_\_\_\_\_ H= \_\_\_\_\_ S= \_\_\_\_\_

(b) A merevlemez kapacitása bájtokban mérve (1 pont):

(b) \_\_\_\_\_

(c) Mennyivel lassabb egy 100 egymásutáni szektorra vonatkozó olvasási kérés teljes kiszolgálási ideje, mint egy 1 szektorra vonatkozó kérésé? (3 pont)

(c) \_\_\_\_\_

(d) Mekkora seek idővel érhetjük el, hogy egy szektor teljes kiszolgálási ideje ugyanakkora maradjon, ha a lemezt fele olyan gyorsan forgatjuk? (ms-ban kifejezve) (3 pont)

(d) \_\_\_\_\_

2. **számpélda** Processzorunk órajel frekvenciája 200MHz. A számítógéphez egy egeret kötünk, amely 40 ms-onként frissíti az elmozdulásvektorát. A processzor részéről az elmozdulásvektor lekérdezése 800 órajelet igényel.

(a) Hányszor kell másodpercenként lekérdezni az egértől az elmozdulásvektort, hogy ne maradjunk le semmiről?

(a) \_\_\_\_\_

(b) Mekkora terhelést jelent a processzor számára, ha az egér kezelésére polling-ot használ (%-ban kifejezve)?

(b) \_\_\_\_\_

(c) Mekkora terhelést jelent a processzor számára, ha az egér kezelésére interrupt-ot használ (%-ban kifejezve)? Feltesszük, hogy az egér csak akkor jelez interruptot, ha elmozdulásvektor megváltozott, ami átlagos használat mellett méréseink szerint másodpercenként 10-szor következik be. A processzor interrupt feldolgozási ideje (az adatátviteli időn felül) 200 órajel.

(c) \_\_\_\_\_

(6 pont)

3. **számpélda** Egy 8 blokkból álló SSD pillanatnyi állapota az alábbi ábrán látható.

0.	18	1.	2	2.	17	3.	8	4.	7	5.	18	6.	3	7.	9
T		H	1	T		H	13	É	5	H	11	T		É	1
T		É	11	T		H	7	H	6	H	8	T		H	5
T		É	9	T		É	8	É	6	H	4	T		H	10
T		H	3	T		É	3	H	9	T		T		É	3

A nagy téglalapok a blokkokat, a bennük lévő kis téglalapok pedig a lapokat reprezentálják. A blokkok bal felső sarkában a sorszám, a jobb felsőben pedig az eddigi törlések száma látható. Minden lap állapotát nyilvántartjuk ("H"=használatban, "É"=érvénytelen, "T"=törölt), valamint, ha volt már rájuk írás, akkor az, hogy melyik LBA cím vonatkozik rájuk (most tekintünk el attól, hogy a szektorméret és a lapméret nem egyenlő).

A kiinduló állapotban a 0-ás, 2-es és 6-os blokkok törölt állapotban vannak, az írási front pedig az 5-ös blokk.

Kérdések:

(a) Hogyan változik az SSD állapota, ha a kiindulási állapotban bekapcsol a szemétygyűjtő algoritmus, és meg sem áll, amíg a törölt blokkok listája eggyel nem nő (vagyis négy törölt blokk nem lesz)? A szemétygyűjtő válassza mindig a legtöbb érvénytelen lapot tartalmazó blokkot, ha több ilyen is van, akkor azon közül a legkevésbé kopottat. Ha új írási frontra van szükség, az SSD válassza azt, amelyik a lehető legegyszerűsebb kopáshoz vezet!

0.		1.		2.		3.		4.		5.		6.		7.	

(4 pont)