

בחינה

פקולטה : מדעי הטבע
מחלקה : מדעי המחשב
שם הקורס : תכנות מתקדם
קוד הקורס : 7031010
תאריך הבחינה : 12/1/2022
סמסטר : א, מועד : 1
משך הבחינה : שעה
שם המרצה : ד"ר פנחס ויסברג

לפניכם 10 שאלות, משקל כל שאלה 10 נקודות
יש לענות על כל השאלות
ללא חומר עזר, אפשר להשתמש במחשבון

בהצלחה

1. (10) מצב מיוחס

- א. מה גורם למעבד לעבור ממצב משתמש (user) למצב מיוחס (kernel) ומה גורם לו לחזור למצב משתמש.
ב. מה מונע מתהליך במצב משתמש (user) לגשת לזיכרון של מערכת ההפעלה.

תשובה :

- א. פסיקה מכל סוג גורמת למעבד לעבור למצב מיוחס, בחזרה מקוד הפסיקה הקרנל מבצע פקודה (iret) שמחזירה למצב רגיל.
ב. דפי הקרנל נמצאים בטבלת הדפים של כל תהליך.
מה שמונע מתהליך לגשת אליהם כל עוד המעבד במצב user הוא סימון הדפים supervisor בטבלת הדפים.

2. open (10)

- א. כשפותחים קובץ במצב של כתיבה והוספה כותבים:
`fd = open("myfile", O_WRONLY | O_APPEND);`

הסבירו איך התוו "ו" (pipe) גורם לצרוף שני המצבים?

ב. מה המשמעות של `O_CREAT | O_EXCL`?

תשובה:

- א. כל מצב מיוצג על ידי ביט אחד, התוו "ו" מבצע OR של הביטים.
- ב. צור את הקובץ אם לא קיים, אם קיים החזר שגיאה.

3. `lseek (10)`

מה יכיל הקובץ "file.txt" לאחר כל אחת מארבע פקודות `write()` שבתכנית.

```
int main(void) {
    char    buf1[] = "123456789012";
    char    buf2[] = "AA";
    int     fd, pos;
    fd = open("file.txt", O_WRONLY | O_TRUNC | O_CREAT
              , S_IRUSR | S_IWUSR);
    write(fd, buf1, 12);
    lseek(fd, -5, SEEK_CUR);
    write(fd, buf2, 2);
    lseek(fd, 3, SEEK_SET);
    write(fd, buf2, 2);
    lseek(fd, -2, SEEK_END);
    write(fd, buf2, 2);
}
```

תשובה:

```
123456789012
1234567AA012
123AA67AA012
123AA67AA0AA
```

4. (10) דיסק

לדיסק יש את הנתונים הבאים:

זמן חיפוש (seek): 10ms
מהירות הסיבוב: 6000RPM (סיבובים לדקה)
מספר הסקטורים במסילה: 100

מה זמן הקריאה של סקטור אחד (במקום אקראי)?

תשובה:

זמן סיבוב: $1000 / 6000 \times 60 = 10\text{ms}$
זמן חצי סיבוב: 5ms
זמן העברה: $10\text{ms} / 100\text{sectors} = 0.1\text{ms}$

זמן הקריאה: $10\text{ms} + 5\text{ms} + 0.1\text{ms} = 15.1\text{ms}$

5. (10) מקומיות

בקוד הבא:

```
int v[2];
int num = 10;
v[0] = num;
v[1] = v[0];
```

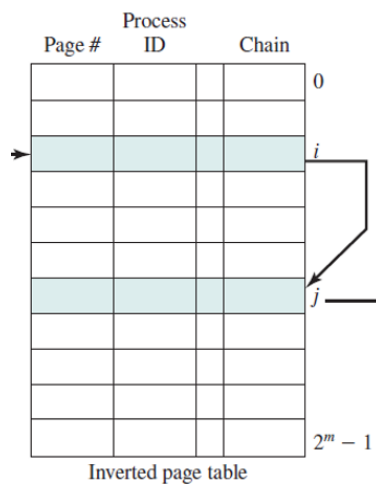
- א. (5) איזה מקרה יש של מקומיות בזמן (בנתונים)?
 ב. (5) אילו שני מקרים יש של מקומיות במקום (אחד בקוד ואחד בנתונים)?

תשובה :

- א.** בנתונים: פעמיים גישה ל- num או ל- v[0].
ב. בקוד: ביצוע שורה אחרי שורה (4 אחרי 3), בנתונים: גישה ל- v[1] אחרי v[0].
הערה: מקומיות בזמן בקוד אפשרית רק כאשר חוזרים שוב על אותה פקודה באותו מקום בקוד, לדוגמה בלולאה.

6. (10) טבלת דפים הפוכה

בטבלת דפים הפוכה משתמשים בטבלת גיבוב (hash table):



כל איבר בטבלה מכיל מספר דף ולוגי ומספר תהליך.

לשם מה מכיל כל איבר ברשימה את מספר הדף ולוגי ואת מספר התהליך?

תשובה :

כדי לדעת לאיזה דף לוגי הדף הפיזי שייך, צריך לכתוב בטבלה את מספר הדף הלוגי.

לכמה תהליכים יכול להיות אותו מספר דף לוגי וכדי שלא ימופו לאותו דף פיזי, פונקציית הגיבוב צריכה לקבל כפרמטר גם את מספר התהליך ולשמור בטבלה.

7. (10) TLB

תכנית ניגשה לכתובת בזיכרון, כדי לתרגם את הכתובת הלוגית לפיזית ה-MMU מחפש ב-TLB.

הסבירו מתי יקרה כל אחד מארבעת המקרים הבאים:

- א. לא נמצא ב-TLB ולא נגרמה פסיקה.
- ב. לא נמצא ב-TLB ונגרמה פסיקה.
- ג. נמצא ב-TLB ולא נגרמה פסיקה.
- ד. נמצא ב-TLB ונגרמה פסיקה.

תשובה:

- א. נמצא בטבלת הדפים, ביט valid בטבלת הדפים דלוק, החומרה מביאה מטבלת הדפים.
- ב. ביט valid בטבלת הדפים כבוי - לא טעון לזיכרון או לא באזורי הזיכרון של התהליך.
- ג. TLB hit.
- ד. אין הרשאת גישה - ניסיון כתיבה ל-read only, ניסיון גישה לדפי supervisor.

8. (10) דפדוף

- א. נניח שגודל הדף הוא 1KB, מה מספר הדף הלוגי שבו נמצאת הכתובת הדצימלית 215201 ?
- ב. מה המרחק (offset) מתחילת הדף ?

תשובה:

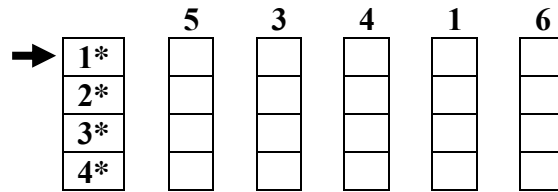
- א. $215201 / 1024 = 210$
- ב. $215201 \% 1024 = 161$

9. (10) אלגוריתם השעון

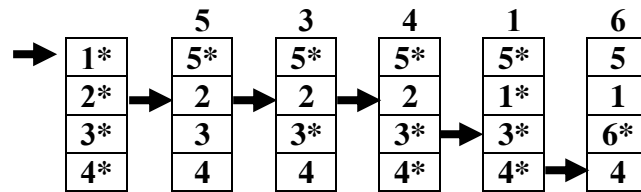
נניח שישנם 4 דפים פיסיים שמכילים את ארבעת הדפים הלוגיים 1,2,3,4. כוכבית מימין למספר הדף מסמנת שהייתה גישה לדף (הביט used דלוק). חץ משמאל למספר הדף מסמן שהמחוג מצביע עליו. (המחוג נע כלפי מטה, כשמגיע למטה חוזר בקפיצה למעלה)

התכנית המשיכה לגשת לסדרת הדפים (משמאל לימין):
5,3,4,1,6

ציירו והשלימו איך תראה כל עמודה לאחר הבאת הדף החסר:



תשובה:



10. Mark and Sweep (10)

בשתי השורות המסומנות בקוד הבא:

```
void mark(ptr p) {
    if ((b = isPtr(p)) == NULL)
        return;
    if (blockMarked(b))
        return;
    markBlock(b);
    len = length(b);
    for (i=0; i < len; i++)
        mark(b[i]);
    return;
}
```

- מה בודקת הפונקציה `isPtr(p)`?
- מה הצורך בלולאה `for (i=0; i < len; i++)`?

תשובה:

- א. האם p מצביע לבלוק תפוס בערימה.
- ב. בשפת $C++/C$ יתכן שכל מילה בבלוק היא מצביע.