

אוניברסיטת בר-אילן, המחלקה למדעי המחשב

מבחן במערכות הפעלה, נוסח א'

מרצה: מר רז לין
מספר קורס: 20-231-89
מועד: סמסטר קיץ תשס"ג, מועד ב'
תוכנית עלית
זמן הבחינה: שעהיים.
תאריך: 18.1.04

המבחן בחומר סגור. אין להשתמש בספרים או חומר עזר כלשהו.
מותר השימוש במחשבון כיס בלבד.

הנחיות:

1. יש לציין בתחילת מחברת הבחינה את נוסח הבחינה.
2. יש לרשום את כל התשובות, כולל התשובות לשאלות האמריקאיות, במחברות הבחינה בלבד בציון מספר השאלה.
3. יש לרשום בכתב יד ברור בעט כחול או שחור בלבד.
4. בשאלות החישוב יש לרשום את כל הנוסחאות בהם השתמשתם והסבר. תשובה סופית בלבד לא תתקבל.
5. בשאלות האמריקאיות ייתכנו כמה תשובות נכונות או אף תשובה. יש לציין את כל התשובות הנכונות. אין צורך לנמק תשובות אמריקאיות.
6. בשאלות הפתוחות יש לנמק במידת הצורך. בכל מקרה יש לענות בתמציתיות ולעניין.
7. ניתן לקחת את טפסי הבחינה.
8. בהצלחה.

1. רוב מערכות ההפעלה המודרניות שיש להן יותר מ-CPU אחד (למשל, Unix) תומכות ב-SMP (Symmetric Multiprocessing). הסבירות בקצרה מה הכוונה.
2. אילו מהפעולות הבאות עשויה לגרום לפסיקה?
 - א. חלוקה ב-0.
 - ב. סיום פעולה של התקן I/O.
 - ג. כתיבה לקובץ.
 - ד. ביצוע פקודת system call.
 - ה. אף תשובה אינה נכונה.
3. כיצד מערכת ההפעלה מגינה על מרחב כתובות הזיכרון שלה מפני תהליכים במערכת? פרטו.
4. תנו לפחות דוגמה אחת לתפקיד של תהליך ה-init.
5. תנו דוגמה למנגנון אחד לתקשורת ישירה בין שני תהליכים שונים. הסבירו בקצרה כיצד המנגנון עובד?
6. מה ההבדל בין kernel-supported threads לבין user-level threads?
7. נכון או לא נכון: Deadlock יכול להיווצר גם אם יש רק 2 תהליכים.

8. במערכת הפעלה כלשהי קיים תזמון Round-Robin עם פלח זמן $q = 1$. בהנחה שכל פקודה לוקחת 1 מילי-שנייה, ובהנחה שתוכנית 1 מתוזמנת ראשונה, אילו מהפלטים הבאים אפשריים?

Program 1:

```
main() {
    int i = 10;
    while (true)
    {
        printf(i++);
    }
}
```

Program 2:

```
main() {
    int i = 0;
    while (i < 3)
    {
        printf(i++);
    }
}
```

- א. 10, 11, 12, 13, ... וכן הלאה, שכן תוכנית אחת מתבצעת כל הזמן בלולאה אינסופית ולא נגיע לתזמון תוכנית 2.
 ב. המתזמן יפענח שיש לולאה אינסופית בתוכנית 1 ולכן יתזמן קודם את תוכנית 2, לכן הפלט יהיה: 0, 1, 2, 10, 11, 12, ... וכן הלאה.
 ג. 10, 11, 0, 12, 1, 13, 14, 2, 15, 16, 17, ... וכן הלאה.
 ד. אף תשובה אינה נכונה.

9. נתונים 4 תהליכים P_1, P_2, P_3, P_4 . להלן פרצי העיבוד וזמני הגעה עבור כל תהליך:

תהליך	פרץ עיבוד	זמן הגעה
P_1	0.3	0.0
P_2	0.5	0.1
P_3	0.3	0.2
P_4	0.2	0.4

הראו טבלת גאנט עבור התזמון של התהליכים לפי FCFS עם לקיחה בכוח.

10. מוצע הפתרון הבא לבעיית הקטע הקריטי בין 2 תהליכים P_i ו- P_j . להלן הפתרון המוצע (עבור תהליך P_i):

Process P_i :

Shared variables:

```
var flag[NUM_OF_PROCESSES]: boolean; /* initialized to FALSE */
var turn: int; /* initialized to 0 */
```

Code:

```
flag[i] = TRUE;
turn = j;
while (flag[j] and turn == j) rest;
```

Critical Section

// end of critical section

האם הפתרון לעיל עונה על התנאים לפתרון בעיית הקטע הקריטי (יש לציין עבור כל תנאי האם הוא עונה או לא. במידה שכן - יש להסביר כיצד. במידה שלא - יש לציין דוגמה נגדית).
 11. מה זה Test&Set? מה זה בא לפתור? האם הפתרון הוא פתרון חומרה או תוכנה?

12. מה ההבדל בין פיצול חיצוני ופיצול פנימי (Internal vs. External fragmentation)? ניתן להסביר על-ידי דוגמה.

13. נתונים רצף הדפים הבא שיש למפות בזיכרון. הזיכרון כולל 3 מסגרות. סדר הגעת הדפים משמאל לימין. השתמשו באלגוריתם LFU (Least Frequently Used) למפות את הדפים ולציין כמה תקלות דף (page fault) נגרמו. יש להדגים את כל התהליך ולא לציין תשובה סופית בלבד:

1, 2, 4, 5, 3, 5, 5, 4, 2, 1

14. מהם ארבעת התנאים המחייבים בכדי שיווצר deadlock?

15. נתון רצף בקשות הדיסק הבאות (משמאל לימין). בדיסק יש 200 גלילים (0 עד 199). השתמשו בשיטת FCFS על-מנת למצוא את מרחק תזוזת הראש. הראש נמצא בתחילה בגליל 23 ונע כלפי מעלה (גליל 199).

24, 2, 124, 56, 99, 153, 34

16. בתוכנית 1 מצוי קוד עבור שני תהליכים שרצים במקביל. מה תהיה תוצאת ההדפסה (הערך של X) בתום הריצה של שני התהליכים? (אם קיימות כמה תוצאות אפשריות יש לציין את כולן).

17. בהנחה שתוכנית 2 בנספח התוכניות רצה ללא שגיאות, מה המספר המקסימלי של תהליכים שירוצו בו זמנית? יש לצייר עץ תהליכים.

18. (ארגון קבצים): במארוז דיסק עם ראשים נעים יש את הנתונים הבאים:

15,000	TRK	גודל מסילה בבתים
3,600	RPM	מס' סיבובים לדקה
4.33 ms	s	זמן חיפוש ממוצע
10	nbt	מס' גושים במסילה

בהתחשב בנתונים לעיל, מהו זמן גישה ממוצע בדיסק (a)?

נספח נוסחאות:

$$B \text{ (block size)} = nsb * S$$

$$IBG = nsb * ISG$$

$$TRK \text{ (track size)} \approx nst \cdot (S + ISG) \approx nbt \cdot (B + IBG)$$

$$r = \frac{1}{2} \cdot \frac{60 \cdot 1000}{RPM} \text{ (ms)}$$

$$a = s + r$$

$$s_l = s_c + \delta \cdot l$$

$$s_l = s_c + \delta \cdot l \Rightarrow s_l \approx s_c, \quad s_l \approx \frac{1}{3} \cdot 2r$$

$$s = s_c + \delta \cdot \frac{N}{3}$$

$$btt = \frac{B}{t} = \frac{\text{size of block}}{\text{transfer time}}$$

$$T_R = \frac{R+W}{t}$$

$$t = \frac{TRK}{2r}$$

$$T_B = \frac{B + IBG}{t} \approx btt$$

$$T_{TRK} = \frac{TRK}{t} = \frac{nbt \cdot (B + IBG)}{t} = 2r$$

$$T_{cyl} = \frac{ntc \cdot TRK}{t} = \frac{ntc \cdot nrt \cdot (R+W)}{t}$$

$$t'_{cyl} = \frac{ntc \cdot nrt \cdot R}{T_{cyl}} = t \cdot \frac{R}{R+W}$$

$$s' = \frac{2r}{nrt} = \frac{T_{TRK}}{nrt} = \frac{R+W}{t}$$

$$t' = \frac{R}{\frac{R+W}{t} + s'} = \frac{R}{2 \cdot \frac{R+W}{t}} = \frac{1}{2} \cdot t \cdot \frac{R}{R+W} = \frac{1}{2} \cdot t'_{cyl}$$

נספח תוכניות:

תוכנית 1:

global:

```
Semaphore S = 1  
X = 0  
A = 1  
B = 2  
C = 5
```

P1:

```
P(S)  
X = A  
X = X + B  
V(S)  
X = X + C
```

P2:

```
A = 9  
P(S)  
B = 21  
C = 43  
V(S)
```

write(X)

תוכנית 2:

main()

```
{  
    if (fork() >= 0) {  
        printf("1");  
        if (fork() >= 0) {  
            printf("2");  
            if (fork() >= 0)  
                printf("3");  
        }  
    }  
}
```