

מס' נבחן

ת"ז

בחינת סמסטר: ב'
השנה: תשע"ח
מועד: ב

תאריך הבחינה: 26/7/18
שעת הבחינה: 17:00
משך הבחינה: 3 שעות

השאלון לא ייבדק בתום הבחינה
ע"י המרצה

מרצה: ד"ר ברק שנהב

שם הקורס: מערכות הפעלה
קוד הקורס: 10303

הוראות לנבחן:

- חומר עזר שימושי לבחינה:
- כל חומר כתוב או מודפס, אך לא יותר מ-40 עמודים.
- מחשבון מדעי "פשוט"

- אין לכתוב בעפרון / עט מחיק
- אין להשתמש בטלפון סלולארי
- אין להשתמש במחשב אישי או נייד
- אין להשתמש בדיסק און קי ו/או מכשיר מדיה אחר
- אין להפריד את דפי שאלון הבחינה

מבנה הבחינה והנחיות לפתרון:

יש לענות במחברת הבחינה על כל השאלות.

*** שאלון הבחינה לא ייבדק ע"י המרצה, לא ייסרק ולא יישמר ***
*** לא יינתן ציון על תשובות אשר תיכתבנה בשאלון זה ***

בהצלחה!

כל הזכויות שמורות © לד"ר ברק שנהב
מבלי לפגוע באמור לעיל, אין להעתיק, לצלם, להקליט, לשדר, לאחסן מאגר מידע, בכל דרך
שהיא, בין מכאנית ובין אלקטרונית או בכל דרך אחרת כל חלק שהוא מטופס הבחינה



שאלה 1 (20 נקודות)

למערכת מחשב פשוטה יש 500KB זיכרון. כתובות פיזיות 0-200KB משמשות את מערכת ההפעלה. בזמן T_0 נמצאים במערכת שלושה תהליכים:

תהליך	זמן סיום	בסיס	גודל
P1	T_0+10	260KB	10KB
P2	T_0+20	400KB	60KB
P3	T_0+15	220KB	30KB

להלן נתונים על חמישה תהליכים שנוספו למערכת לאחר זמן T_0 :

תהליך	זמן התחלה	זמן סיום	גודל
P4	T_0+1	T_0+50	10KB
P5	T_0+2	T_0+24	40KB
P6	T_0+3	T_0+24	10KB
P7	T_0+12	T_0+24	10KB
P8	T_0+18	T_0+50	28KB

- כיצד תיראה מפת הזיכרון בזמן T_0+25 אם נעשה שימוש בשיטת First Fit? הסבירו
- כיצד תיראה מפת הזיכרון בזמן T_0+25 אם נעשה שימוש בשיטת Best Fit? הסבירו
- כיצד תיראה מפת הזיכרון בזמן T_0+25 אם נעשה שימוש בשיטת Worst Fit? הסבירו
- כיצד תיראה מפת הזיכרון בזמן T_0+25 אם נעשה שימוש ב-Paging כאשר גודל הדף / המסגרת הינו 4 KB? הניחו שלכל תהליך מוקצות מסגרות בכמות הנדרשת להכיל את כל התהליך בזיכרון, כאשר בכל פעם מקצים את המסגרת הפנויה הראשונה שנמצא (המסגרות של תהליכים P1-P3 הינן רציפות בזיכרון).

במידה וחסר נתון עשו שימוש בהנחה סבירה.

שאלה 2 (12 נקודות)

- (4 נק') הסבירו, בקצרה, מהם user mode ו-kernel mode?
- (6 נק') לכל אחת מהפעולות הבאות, ציינו (והסבירו, בקצרה, את בחירתכם) האם היא תבוצע ב-user mode, ב-kernel mode או בשניהם:
 - גישה לרכיב קלט/פלט, למשל כרטיס תקשורת
 - גישה לזיכרון הראשי (RAM)
 - בקשת שירות ממערכת ההפעלה (קריאה System call)
- (2 נק') ישנם מעבדים בהם יש יותר משני מצבים (modes). הסבירו, בקצרה, לאיזה צורך יכולים לשמש מצבים נוספים אלו?

שאלה 3 (20 נקודות)

נתונה התוכנית הבאה :

```
#include <sys/types.h>
#include <stdio.h>
#include <unistd.h>

int main()
{
    pid_t pid;
    int i;

    printf("START\n");
    for (i=0 ; i < 4 ; i++ /* notice i++ below! */) {
        printf("A%d\n", i);
        if ((pid = fork()) < 0) {fprintf(stderr, "Fork failed\n"); return 1;}
        if (pid == 0) {i++; printf("B%d\n", i);}
        if (pid > 0) {wait(NULL); printf("C%d\n", i);}
    }
    printf("END\n");
    return(0);
}
```

בהנחה שכל קריאות המערכת (system calls) הצליחו :

- א. (2 נק') מה משמעות קריאת המערכת wait?
- ב. (6 נק') כמה פעמים תקרא קריאת המערכת fork? הסבירו
- ג. (12 נק') מה יהיה פלט התוכנית (בהנחה שכל פקודות ההדפסה מתבצעות מידית)?
הסבירו

שאלה 4 (12 נקודות)

- א. (4 נק') לכל אחד מהאלגוריתמים הבאים לתזמון תהליכים (scheduling) ציינו (והסבירו, בקצרה, את בחירתכם) האם הוא preemptive, nonpreemptive או שלא ניתן לדעת :
 - i. FCFS
 - ii. SJF
 - iii. Priority
 - iv. Round robin
- ב. (4 נק') הסבר בקצרה מהי מערכת זמן-אמת? מה ההבדל בין מערכת Hard real-time (למשל מערכת לניהול רמזורים) לבין מערכת Soft real-time (למשל חלונות וגרסאות מסוימות של יוניקס)?
- ג. (4 נק') במערכות מרובות מעבדים מקובלות שתי גישות שונות לתזמון תהליכים :
 - i. תור משותף לכל המעבדים
 - ii. תור נפרד לכל מעבדמה היתרון של כל אחת משיטות אלו ביחס לאחרת? הסבירו, בקצרה

שאלה 5 (10 נקודות)

- א. (3 נק') מהו סמפור (Semaphore) ומהו מיוטקס (Mutex)?
- ב. (4 נק') ציינו והסבירו, בקצרה, בעיות מרכזיות שעלולות להיווצר כתוצאה משימוש "רשלני" במנעולים. אין צורך לפרט את הדרכים להתמודד עם בעיות אלו.
- ג. (3 נק') מהו spinlock? מדוע מומלץ, בדרך כלל, לא לעשות שימוש במנעול העושה שימוש ב-spinlock? ציינו מקרה שבו spinlock הינו פתרון עדיף, הסבירו, בקצרה.

שאלה 6 (10 נקודות)

- א. (4 נק') הסבירו, בקצרה, מהו זיכרון ווירטואלי? לאיזה צורך הוא משמש?
- ב. (6 נק') מהו page fault? מהם השלבים העיקריים בטיפול ב-page fault? הסבירו, בקצרה.

שאלה 7 (10 נקודות)

"Sometimes, when a processor accesses memory, it spends a significant amount of time waiting for the main memory. This situation is known as a **memory stall**. A memory stall may take tens and even few hundreds of CPU cycles."

- א. (4 נק') הסבירו, בקצרה, מהו memory stall? ומה יכול לגרום לתופעה זו?
- ב. (4 נק') הסבירו, בקצרה, מדוע קשה (או לא ניתן) להתמודד עם תופעה זו באמצעות מערכת ההפעלה?
- ג. (2 נק') מהו פתרון אפשרי, בחומרה, לתופעה? ציינו, בקצרה, יתרונות וחסרונות לפתרון זה.

שאלה 8 (6 נקודות)

- א. (4 pt.) A thread can periodically check its interruption status by invoking either the interrupted() method or the isInterrupted() method, both of which return true if the interruption status of the target thread is set. How do these methods differ?
- ב. (2 pt.) Explain why both methods are provided?

פתרון

פתרון מועד
10303

שאלה 1

בזיכרון תיוותר מערכת ההפעלה בכתובות 0-200 ושני תהליכים

א. P4 – 200-210

P8 – 310-338

ב. P4 – 250-260

P8 – 310-338

ג. P4 – 270-280

P8 – 330-358

ד. P4 – 200-212

P8 – 240-260, 272-280

שאלה 2

א. מצבי פעולה של המעבד. מצב משתמש משמש להרצת קוד של המשתמש ומצב ליבה משמש בד"כ להרצת הקוד של מערכת ההפעלה. כאשר המעבד עובר ממצב למצב משתנה האופן בו מבוצעות חלק מפקודות האסמבלי. למשל, במצב משתמש פקודת STORE תיכשל אם ניגשים לכתובת לא חוקית בעוד שבמצב ליבה הבדיקה לחוקיות הכתובת בד"כ מושבתת.

ב. i – kernel

ii – both

iii – user

ג. Virtual machines

שאלה 3

א. העברת התהליך למצב wait עד שתהליך בן מסיים את ריצתו

ב. 7

ג. A0, B1, A2, B3, END, C2, A3, B4, END, C3, END, C0, A1, B2, A3, B4, END, C3, END, C1, A2, B3, END, C2, A3, B4, END, C3, END

שאלה 4

- א. i – non preemptive
ii – both
iii – both
iv – preemptive
- ב. מערכת בה כל בקשה מתבצעת תוך זמן סופי, קבוע מראש (או שניתנת הודעת אי הצלחה מיד עם קבלת הבקשה). מערכת זמן אמת קשה מקפידה על ההגדרה הנ"ל ובודקת שאכן כל בקשה תסתיים במועד. מערכת זמן אמת רכה "משתדלת" לעמוד בבקשות ע"י מתן עדיפות "עליונה" לבקשות (והגבלת המימוש לבקשות לזמן קצוב).
- ג. תור משותף – איזון עומסים
תור נפרד – affinity – המשפר בד"כ את ביצועי זיכרון המטמון.

שאלה 5

- א. מנעול המקבל ערך שלם
מנעול בעל ערך בוליאני
- ב. Deadlock, Priority inversion
- ג. מנעול הממתין "המתנה חמה". כאשר עלות ההמתנה החמה (מבזבז מעבד) נמוכה מעלות החלפת ההקשר, למשל כשיש עודף מעבדים או שזמן ההמתנה יהיה קצר מאוד.

שאלה 6

- א. מעבר למערכת ההפעלה (שמירת התהליך "בצד", מעבר ל-kernel mode)
- ב. קריאה ל-interrupt handler שמטפל ב-PF.
- ג. איתור מסגרת פנויה (במידה ואין יש לבחור "קורבן" ולפנות אותו לזיכרון המשני)
- ד. קריאת הדף מהדיסק למסגרת הפנויה (מבוצע ע"י ה-DMA, בזמן זה ה-CPU יטפל בתהליך אחר)
- ה. כאשר קריאת הדף תסתיים יתרחש interrupt חומרתי. עדכון טבלת הדפים של התהליך.
- ו. העברת התהליך חזרה לתור ה-Ready.

שאלה 7

- א. השהיה קצרה בריצת המעבד. בד"כ כתוצאה מהחטאות מטמון (cache miss) או סיעוף (branching).
- ב. המעבר למערכת ההפעלה ובחזרה לתהליך יקח, בד"כ, יותר זמן ממשך ה-memory stall.
- ג. Hardware threads – מאפשר הרצת מספר תהליכים בו"ז על המעבד, כך שאם אחד (או יותר) מהתהליכים נקלע ל-memory stall שאר התהליכים ממשיכים לרוץ ומנצלים את המעבד.

שאלה 8

נושא ללימוד עצמי