

בחינה בעקרונות מערכות הפעלה

קרא בעיון לפני שתתחיל בפתרון הבחינה!

- א. בבחינה זו 18 שאלות סגורות (מבחן אמריקאי).
עליך לבחור בכל פעם בתשובה יחידה מבין התשובות המוצעות ולהקיף בעיגול את אות התשובה שבחרת, על גבי השאלון עצמו בלבד ליד טקסט השאלה
- ב. הציון נקבע על פי מספר השגיאות. כל שגיאה מורידה 6 נקודות. כל שאלה ללא תשובה מורידה רק 4 נקודות. הציון ההתחלתי הוא 105. הציון המרבי שניתן לקבל הוא 100. הציון הנמוך ביותר שניתן לקבל הוא 0.
- ג. כל חומר עזר אסור לשימוש בזמן הבחינה פרט למחשבון כיס פשוט. הבחינה נמשכת שלוש שעות.
- ד. במקרה של ספק בהבנת ניסוח השאלה ניתן לכתוב על גבי שאלון הבחינה הערה המבהירה את דרך הבנתך של השאלה. אין לכתוב נימוקים לתשובה!

בהצלחה!

שאלה 1.

hardlink במערכת הפעלה UNIX הנר:

- (א) קובץ מיחד (special file) במערכת הקבצים של מערכת ההפעלה.
- (ב) מושג המתקשר ל i-node.
- (ג) רכיב בטבלת FAT (file allocation table).
- (ד) אף תשובה קודמת אינה נכונה.

שאלה 2.

להלן טבלה עם זמני הגעת תהליכים והזמן הדרוש לעיבוד של כל אחד מהם. חשבו כמה תהליכים מסתיימים ביחידת זמן בממוצע (throughput) כאשר אלגוריתם התזמון הוא FIFO (עיבוד באצווה - batch). מס' יחידות הזמן נמדד החל מתחילת עיבוד התהליך הראשון ועד לסיום התהליך האחרון.

תהליך	זמן הגעה	זמן CPU
A	0	1
B	1	2
C	2	1
D	3	4

(א) $\frac{1}{2}$.

(ב) 1.

(ג) 2.

(ד) $\frac{3}{5}$.

שאלה 3.

מה התכונות של האלגוריתם הבא למניעה הדדית, עבור התהליכים $i = 0, 1$

אתחול (שני התהליכים משתפים את המשתנה):

turn = 1

אלגוריתם:

```
while (true){
    if (turn == i) {
        // ... Critical Section
    }
    else {
        while (turn <> i) ; // wait
        // ... Critical Section
    }

    turn = 1 - i;
}
```

(א) יש מניעה הדדית ויש הגינות.

(ב) יש מניעה הדדית ואין הגינות.

(ג) אין מניעה הדדית אבל יש הגינות.

(ד) אין מניעה הדדית ובנוסף יש סכנת deadlock.

רמז: היזכרו כיצד מטפל פתרון של Peterson בהגינות לשני תהליכים.

שאלה 4.

גרף המצבים של תהליך מכיל 3 מצבים בסיסיים: ready, blocked ו ready.

(א) המעבר מ running ל ready נובע תמיד מפעולה שבוצעה ע"י התהליך עצמו.

(ב) המעבר מ blocked ל ready נובע תמיד מפעולה שבוצעה ע"י התהליך עצמו.

(ג) המעבר מ ready ל running נובע לרוב מפעולה שבוצעה ע"י התהליך עצמו.

(ד) המעבר מ running ל blocked נובע לרוב מפעולה שבוצעה ע"י התהליך עצמו.

שאלה 5.

אלו מבין האלגוריתמים הבאים לתזמון תהליכים סובלים מבעיית ההרעבה (starvation)?

א) FCFS (first come first served)

ב) FCFS ו Round-Robin

ג) SJF (shortest job first) ותזמון לפי עדיפות קבועה (fixed priority scheduling)

ד) SJF (shortest job first) ותזמון לפי עדיפות משתנה (priority scheduling)

שאלה 6.

מערכת הפעלה נתונה משתמשת בשיטת ניהול הזיכרון באמצעות מחיצות בגודל משתנה (variable

partitioning).

להלן גדלים של המחיצות הפנויות המקושרות ברשימה בסדר משמאל לימין :

100 K, 500 K, 200 K, 300 K, 600 K

אלגוריתם המנהל את המחיצות מתבקש לטפל בתהליכים בגודל 212 K, 417 K, 112 K, 426 K

(סדר ההגעה הוא משמאל לימין).

איזו שיטה להקצאה במחיצות תהיה העדיפה מבין 2 השיטות הבאות: best-fit או first-fit בנתונים

הנ"ל?

א) first-fit עדיפה על best-fit.

ב) best-fit עדיפה על first-fit.

ג) אין הבדל ממשי בניהם.

ד) לא ניתן לספק תשובה עקב חוסר הנתונים.

שאלה 7.

מספר הסיביות (bits) של כתובת מדומה (virtual address) להיסט (offset) הנו 12. מילת זיכרון הנה באורך 32 סיביות. מהי גודלה של מסגרת (frame) בזיכרון הפיזי?

(א) 16K.

(ב) 8K.

(ג) 128K.

(ד) 4K.

שאלה 8.

להלן רשימת אלגוריתמים להחלפת דפים בזיכרון מדופדף: (1) LRU, (2) FIFO, (3) האלגוריתם האופטימלי, (4) אלגוריתם ההזדמנות השניה (second chance). אלו מן האלגוריתמים הללו סובלים מאנומלית בלאדי (Belady)?

(א) FIFO ו LRU.

(ב) FIFO.

(ג) LRU ואופטימלי.

(ד) FIFO ואלג' ההזדמנות השניה.

שאלה 9.

רשימות היכולות (capability lists) מוחזקות בד"כ במרחב הזיכרון של התהליך. כיצד מובטח שהמשתמש לא ישנה את תוכן הרשימה?

(א) ע"י שימוש בתכונת ה tagged architecture של החומרה.

(ב) ע"י הצפנת הרשימה במפתח סודי שלא ידוע למשתמש.

(ג) ע"י כך שכדי לשנות את רשימת היכולות נדרשות הרשאות מנהל מערכת.

(ד) 2 מן התשובות הקודמות נכונות.

(ה) שלושת התשובות הראשונות נכונות.

שאלה 10.

האם קיימת דרך סבירה להילחם בתופעת הדלפת מידע ע"י שימוש בערוצים חסויים (covert channels) בצורה מקיפה?

- (א) כן, ע"י הקצאה ושחרור אקראי של משאבים.
- (ב) כן, באמצעות חסימת האפשרות ליצירה ומחיקה של קבצים ע"י תהליכים החשודים להדלפה.
- (ג) לא.
- (ד) כן, ע"י צירוף של השיטות שהוזכרו בסעיפים א' ו ב'.

שאלה 11.

דיסקים מסוימים מחזיקים סוג של זיכרון מטמון שנקרא `track-at-a-time cache`. בזיכרון זה נשמרים כל הבלוקים שהראש הקורא עבר עליהם במסלול עד שהגיע לסקטור המבוקש. זיכרון זה שקוף ל `device independent software` (כלומר על קיום זיכרון זה ברמת התוכנה ידוע רק ל `device driver`). מה הם החסרונות של זיכרון מטמון מסוג זה?

- (א) סיבוכיות נוספת של `device driver`, חומרה מתוחכמת יותר, בזבוז CPU על העתקת בלוקים מן הזיכרון המטמון אל הזיכרון הראשי.
- (ב) סיבוכיות נוספת של `device driver`, חומרה מתוחכמת יותר, מספר רב יותר של תזוזות זרוע הדיסק.
- (ג) סיבוכיות נוספת של `device driver`, חומרה מתוחכמת יותר, זמן המתנה רב יותר לסיבוב הדיסק עד שראש הקורא נעמד על הסקטור הדרוש.
- (ד) אף תשובה קודמת אינה נכונה.

שאלה 12.

לשם מה מתעורר הצורך בנעילת (locking) דפים בזיכרון הפיזי?

- (א) למנוע מצב של הוצאת דף אליו מתבצע DMA transfer (וכתוצאה מכך למנוע שחלק המידע מהדיסק ייכתב על הדף החדש שנטען במקום זה שיוצא).
- (ב) כדי לממש מנגנון הבטחה (protection) מפני תהליכים שמנסים לשכתב קטע קוד (code segment) של עצמם.
- (ג) בדרך זו ניתן להבטיח כי כל הדפים מקבוצת העבודה (working set) של התהליך יישארו בזיכרון הפיזי.
- (ד) 2 מן התשובות הקודמות נכונות.

שאלה 13.

- ☐ במערכת מסוימת 4 משאבים מאותו סוג ושלושה תהליכים.
 - ☐ כל תהליך יכול לדרוש 2 משאבים לכל היותר.
 - ☐ 2 תהליכים אינם יכולים להשתמש במשאב בו זמנית.
 - ☐ תהליך אינו משחרר משאבים שהוקצו לו עד שיסתיים.
 - ☐ אין חילץ כפוי של משאבים.
 - ☐ תהליך שמקבל כל המשאבים הדרושים מסתיים בזמן סופי.
- האם יש צורך בטיפול בקיפאון במערכת?

- (א) אין צורך בכך מפני שאין חשש להיווצרות הקיפאון.
- (ב) יש צורך בכך מפני שעלולה להיווצר המתנה מעגלית.
- (ג) יש צורך בכך מפני שעלול להיווצר קיפאון, אומנם לא נוצרת המתנה מעגלית.
- (ד) אף תשובה קודמת אינה נכונה.

שאלה 14.

להלן מדיניות הקצאת משאבים במערכת מסוימת:

תהליך יכול לבקש/לשחרר משאבים בכל עת. לכל תהליך קיים וקטור של משאבים אשר להם ממתין תהליך. אם לא ניתן לספק בקשה למשאב כלשהו, אז המערכת עוברת על תהליכים אשר נמצאים במצב (blocked) עקב המתנה למשאבים. אם הם מחזיקים במשאב הדרוש, אז המשאב נחלץ מידיהם וניתן לתהליך הדורש. וקטור המשאבים אשר להם ממתין התהליך מתעדכן בהתאם. האם יכול להיווצר קיפאון במערכת כזאת?

דוגמא:

למשל במערכת עם 3 סוגי משאבים וקטור סך המשאבים במערכת הוא (4,2,2). תהליך P0 מבקש (2,2,1), P1 מבקש (1,0,1) ושניהם מקבלים את בקשתם. אם לאחר מכן P0 מבקש (0,0,1) הוא עובר למצב blocked (אין משאב 3 פנוי). כעת אם P2 מבקש (2,0,0) הוא מקבל משאב אחד שפנוי ואחד נלקח מ P0 (כי P0 נמצא במצב blocked). וקטור המשאבים שהוקצו ל P0 מתעדכן ל (1,2,1) ווקטור המשאבים המבוקשים מתעדכן ל (1,0,1).

- א) כן, כי עלולה להיווצר המתנה מעגלית בין התהליכים.
- ב) כן, כי יש חילוף משאבים.
- ג) לא, כי לא מתקיים תנאי החזק והמתן (hold and wait).
- ד) לא, אך עלול לקרות שתהליכים ישהו זמן לא מוגבל במצב blocked.

שאלה 15.

ניתן לייעל את קריאת המערכת fork במערכת UNIX :

- א) ע"י שיתוף קטעי זיכרון ושכפולם לפי הצורך.
- ב) ע"י ביצוע קריאה לקריאת מערכת exec מייד לאחר חזרה מ fork.
- ג) ע"י השהיית תהליך הבן לזמן מה כדי להבטיח שתהליך האב יתוזמן קודם.
- ד) כל התשובות נכונות.

שאלה 16.

קריאת המערכת readlink במערכת UNIX נועדה ל:

- א) קריאת התוכן של קובץ הספרייה.
- ב) קריאת התוכן של קובץ ה soft link.
- ג) קריאת התוכן של קובץ המוצבע ע"י soft link.
- ד) שליפת מידע על קובץ ה softlink.
- ה) שליפת מידע על קובץ המוצבע ע"י softlink.

שאלה 17.

האם קיימים במערכת הפעלה NT מנגנונים להרצת קבצי הרצה (בינאריים) של MS-DOS ?

- א) כן, עבור גרסאות NT למעבדים אשר אינם תומכים ב instruction set של מעבדי Intel, קיים מנגנון האמוליציה (emulation) של פקודות ה- instruction set.
- ב) כן, קיים מנגנון הטיפול בקריאות המערכת MS-DOS דרך ה environmental subsystem.
- ג) כן, אך לא מובטחת תאימות בינארית (binary compatibility) מלאה.
- ד) כל התשובות הקודמות נכונות.
- ה) 2 תשובות קודמות בלבד נכונות.

הערה: אמוליצית פקודות ה- instruction set פירושה תרגום בין פקודות ה- instruction set של מעבד אחד לפקודות ה- instruction set של מעבד אחר.

שאלה 18.

מה הוא ה idle thread במערכת הפעלה NT ?

- א) ה thread עם עדיפות הכי גבוהה במערכת.
- ב) ה thread אשר נמצא במצב waiting.
- ג) ה thread אשר רץ כל עוד אין threads במצב ready.
- ד) ה thread אשר הסתיים.

בהצלחה