

מס' שאלון - 496

1

באפריל 2015

סמסטר 2015א

מס' מועד 92

20594 / 4

שאלון בחינת גמר

20594 - מערכות הפעלה

משך בחינה: 3 שעות

בשאלון זה 10 עמודים

מבנה הבחינה:

קראו בעיון לפני שתתחילו בפתרון הבחינה!

א. המבחן מורכב משלושה חלקים.

ב. בחלקים א ו - ב מופיעות שאלות פתוחות. ענו תשובות מלאות, בכתב קריא ובקיצור נמרץ. אין חובה להשתמש בכל השורות המוקצות לצורך התשובות, אך אין לחרוג מהמקום המוקצה.

ג. בחלק ג ( שאלות אמריקאיות ) עליכם לבחור בכל פעם בתשובה יחידה מבין התשובות המוצעות ולהקיף בעיגול את אות התשובה שבחרתם.

את התשובות לכל השאלות יש לכתוב בשאלון הבחינה.

חומר עזר:

כל חומר עזר אסור בשימוש, פרט למחשבון, שאינו אוצר מידע.

החזירו

למשגיח את השאלון

וכל עזר אחר שקיבלתם בתוך מחברת התשובות

בהצלחה !!!









חלק א (55 נקודות)

ענו על שלוש השאלות הבאות.

שאלה 1 (24 נקודות)

למדנו על אלגוריתם פטרסון המהווה פתרון סביר לבעיית הקטע הקריטי בין שני תהליכונים. הפתרון שלמדנו עובד על ארכיטקטורה עם זיכרון מטמון. כעת הניחו שבמחשב עברו אתם כותבים קוד אין זיכרונות מטמון (אין cache). כלומר, כתיבה על-ידי תהליכון אחד לזיכרון תראה באופן מיידי על ידי תהליכון אחר. להלן פסודו-קוד שמתאר את האלגוריתם למניעה הדדית אשר חסרים בו מספר פרטים:

סימונים:

Q1 מציין שתהליכון T1 מעוניין להיכנס לקטע הקוד הקריטי.  
Q2 מציין שתהליכון T2 מעוניין להיכנס לקטע הקוד הקריטי.  
קו תחתון (\_\_\_\_) מציין מקום שבו עליכם להשלים את קטע הקוד

```
Q1 := false;  
Q2 := false;  
TURN = 1; // valid values for TURN are 1 and 2
```

protocol of T1	protocol of T2
<pre>Q1 = true; TURN = <u>1</u>; while( Q2 and <u>TURN==1</u> ) /*do nothing*/ ; /* here is the critical section */ Q1 := false;</pre>	<pre>Q2 = true; TURN = <u>2</u>; while( Q1 and <u>TURN==2</u> ) /*do nothing*/ ; /* here is the critical section */ Q2 := false</pre>

(המשך השאלה בעמוד הבא)







8 נקי' + א. נא השלימו את הפרטים (בטבלה לעיל) כך שמובטח שיש מניעה הדדית ואין הרעבה.

(8 נקי') + ב. הסבירו מדוע האלגוריתם שהשלמתם מבטיח מניעה הדדית.

## הסבר:

תהלים 119: אהבים לקח קל' א<sub>2</sub>=false וכו

[illegible]

(8 נק') + ג.

## הסבר:

תהליך לא יכול לקרות בלעדיו

האליסות לזכר נחמן

מקרים

 ~~$\chi^2_{(p)} \sim \chi^2_{(n)} T_1 \sim \chi^2_{(n)} \frac{1}{n}$~~ 

(1) תהליך  $T_2$  מחקה קלוסיה אלוסיות, תהליך  $T_1$  מסיר קלס

ק"ל' וּמִמֶּשֶׁלֶךְ הַלְשֵׁמוֹת דְּתִיבְרֵי' מִלֵּל' לְמִינֵי ק"ל' :

המקרא שם זרעו ה'צ"ח ש, מהקצ"ח הקר"א

היום יום שבת 10/12/2019  
המשך הבחינה בעמוד הבא

**המשך הבחינה בעמוד הבא**

ה-CPU הוא 'כוכב' המרכזי בעמדה הראשית.

(2) כמו המקרא הראשון רק הפעם  $T_1$  לאמר "אם קטע"

מחנה מליכנס יחד ע"ע

$T_1$  יורד,  $\alpha_1 = \text{true}$  ו  $\text{turn} = 1$  ו'תקף ציפיה

ה'ש"ס י"ח

turn=1 - עיבוד CPU 1ms לקיחה  $T_2$  זמן

[illegible]

ד"ר גמלקר, 54 יולי 1982

20594 / 92 —2015N

\* כל מקום שכתוב תה"ל<sup>3</sup> הכוונה לתה"ל 594 / 92 - 2015א







שאלה 2 (18 נקודות)

ביל וסטיב החליטו לחבל בקוד הפתוח של מערכות Linux באמצעות הכנסת זמן שהייה גרוע ככל האפשר למערכת. בשאלה זו נעזור להם לתכנן זמן גרוע ככל האפשר.

ביל הציע לממש את מדיניות הזימון LJF – Longest Job First. מדיניות זו הפוכה בהתנהגותה ממדיניות SJF שלמדנו. כלומר, הניחו שכל התהליכים מגיעים יחד וזמני הביצוע ידועים מראש, ונריץ את התהליכים לפי זמן הביצוע בסדר יורד, כשכל תהליך רץ מתחילתו ועד סופו.

(6 נק') א.

האם מדיניות LJF היא הגרועה ביותר מבין כל המדיניות בלי הפקעה, לפי מדד זמן השהיה הממוצע? הוכיחו או תנו דוגמה נגדית.

בין המקרים  
הנוריים  
הם המקרים  
הנוריים  
הם המקרים  
הנוריים

ספק כן, לא ניתן להוכיח כי היא הגרועה ביותר מבין כל המדיניות בלי הפקעה, לפי מדד זמן השהיה הממוצע.

אם כל התהליכים יהיו בעלי אותו זמן קצב, אזי זמן השהיה הממוצע של LJF יהיה זהה לזה של SJF.

(4 נק') ב. סטיב טוען שקיימות מדיניות עם הפקעה גרועות יותר מ-LJF. האם סטיב צודק? הסבירו. אם כן, תנו דוגמה.

ההוכחה היא שיש מדיניות עם הפקעה גרועה יותר מ-LJF, וזוהי מדיניות SJF.

תהליך	זמן קצב	זמן השהיה
A	30	30
B	20	20
C	10	10

(5 נק') ג. נניח שהגיעו למערכת מ תהליכים. נסמן את זמני הריצה שלהם ב-  $t_i$  ואת זמני

השהיה שלהם תחת SJF ב-  $S_i$ . (שימו לב, זמן השהיה כולל את זמן הריצה). מהו זמן ההשהיה של התהליך ה-  $i$  תחת LJF? הניחו שזמני הריצה של כל התהליכים שונים.

נניח, דוגמאות:  $t_1 < t_2$ , אזי זמן השהיה של התהליך הראשון יהיה  $S_1 = t_1$  וזמן השהיה של התהליך השני יהיה  $S_2 = t_1 + t_2$ .

לפיכך, זמן השהיה של התהליך ה-  $i$  תחת LJF יהיה  $S_i = t_1 + t_2 + \dots + t_i$ .

נסמן  $L_i$  את זמן השהיה של התהליך ה-  $i$  תחת SJF.

לפיכך, זמן השהיה של התהליך ה-  $i$  תחת SJF יהיה  $L_i = t_n + t_{n-1} + \dots + t_i$ .







שאלה 3 (13 נקודות)

496 recently

6 נק' + א. הסבירו מהי הבעייתיות של האלגוריתם LRU להחלפת דפים כאשר מממשים אותו באמצעות רשימה משורשרת.

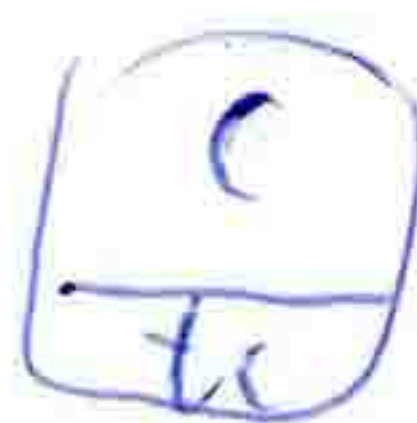
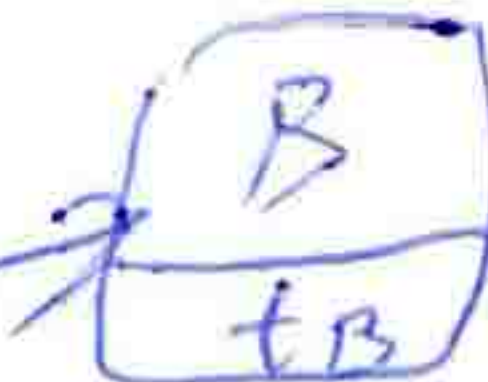
הסבר:

זהל בעם שניש'ם לעם נדרש לעדור על כל הא'ה'ם  
 של הרש'מה המקושרת ונקראו הצורך (המקרא של פ'ה'ה)  
 נדרש לעדכן את הרש'מה בשלם שפ'מות ה'ש'ות לעד'ם'ה'א  
 מאוק לעד'ה וכל'ות הא'ק'ל'ם הרש'מה המקושרת פ'ה' כן  
 יכולה לעלות ג'לה, פ'ת'ון לה ה'א' לא פ'ק'ה' (פ'ק'ה')

7 נק' + ב. תארו את אלגוריתם WSClock (working set clock) להחלפת דפים ברמה של פסיאודו-קוד.

תיאור האלגוריתם:

הק'ה' (על מ' מצ'י'ם ה'ת'ם) ~~ה'ת'ם~~ while(1) ✓  
 { process = getNextProcess(); // ה'ת'ם ו'ת'ם ו'ת'ם  
 pTime = process.time; // ה'ת'ם א'ת'ם א'ת'ם  
 if ((CURRENT\_TIME - pTime) > T) // ה'ת'ם א'ת'ם א'ת'ם  
 { swapPages(process, new); // ה'ת'ם א'ת'ם א'ת'ם  
 new.time = CURRENT\_TIME; // ה'ת'ם א'ת'ם א'ת'ם  
 nextProcess(); // ה'ת'ם א'ת'ם א'ת'ם  
 break; // ה'ת'ם א'ת'ם א'ת'ם  
 }  
 use {  
 process.time = CURRENT\_TIME; // ה'ת'ם א'ת'ם א'ת'ם  
 nextProcess(); // ה'ת'ם א'ת'ם א'ת'ם  
 }  
 }  
 המשך הבחינה בעמוד הבא









## חלק ב (25 נקודות)

ענו על חמש השאלות הבאות. משקל כל שאלה 5 נקודות.

+ שאלה 4

מה הם external fragmentation ו- internal fragmentation?

external fragmentation - ריסוק חיצוני - כאשר נוצרים מנוחים  
 בין הקטורים המוקצים בפ'כרין, אים מנוחים אלו קטנים  
 יש לא ניתן להקצות להם זיכרון נוסף וזה מקום המזלזל.  
 internal fragmentation - ריסוק פנימי - כאשר מוקצה יותר זיכרון מאשר  
 הדרוש, בקוק לא, תהליך יחסיק הזיכרון זה וגוא יהיה המזלזל.

+ שאלה 5

מהו hard-link? מה ההבדל בינו לבין soft-link?

hard-link - זיכרון יש המבנה לקובץ מסוים (ל-*inode* של הקובץ)  
 ה-*inode* סופר כמה זמית מופנים לקובץ זה.  
 soft-link - קיצור דרך לקובץ (הפניה)  
 ההקצאה מחקת קובץ שיש מופנה האחיד-*inode* תכרום למחלקת  
 הקובץ עצמו והאחיד-*inode* יהיה שונה, לעומת זאת מחלקת  
 קובץ האחיד-*inode* לא תמחק את הקובץ כל עוד יש קצבים המפנים  
 אליו.

+ שאלה 6

הסבירו כיצד ניתן לזהות סחרור (thrashing) במערכת.

thrashing - סחרור - התאמה מרובה של דאוקים  
 של זיכרון המערכת.  
 ניתן לזהות סחרור כאשר התהליך מקבל זמן CPU  
 אולם מתקשים מסוים (התהליך לא מוכרז יסא גולסיה  
 אינסופי אבל קצב של החיפוש הוא מסוים אולי).  
 כ' 337  
 -2

המשך הבחינה בעמוד הבא



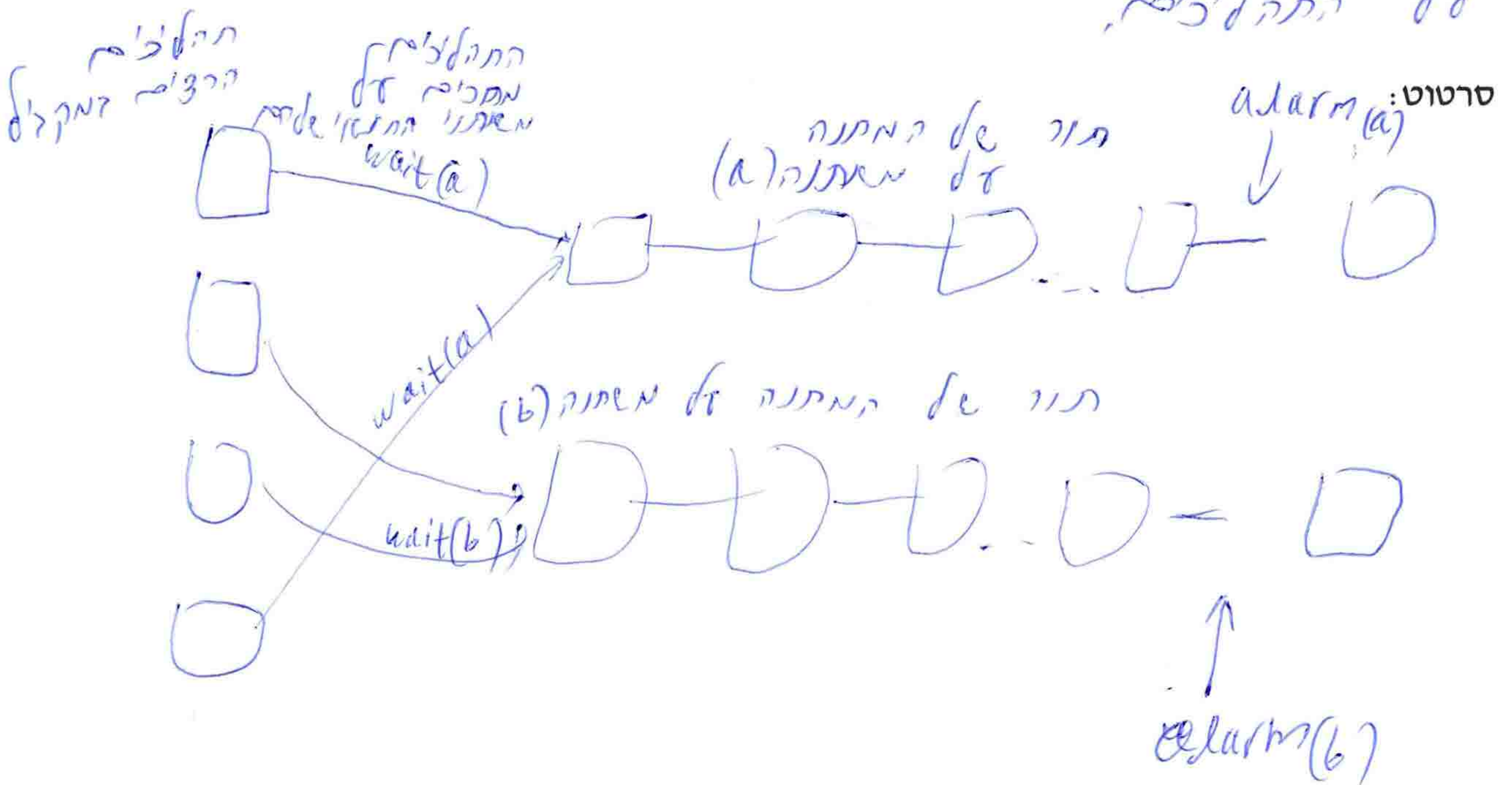




מהו monitor? סרטטו כיצד הוא פועל (יש לסרטט את המימוש של ה Monitor: תור/תורים של

תהליכים, תור/תורים של המתנה וכו').

סרטטו  
 monitor הוא מנגנון של שפה המציד מתודות  
 לניהול משאבים קריטיים ולקבוצה של קצרים קריטיים  
 למעט כך הוא מסתמך על condition variable  
 שסליהם ניתן לבצע פעולות wait/alarm כדי  
 על התהליכים.



ה'זמן בקופת הממשל  
 לקבוצה של תהליכים  
 הממתנה על משתנה

המשך הבחינה בעמוד הבא

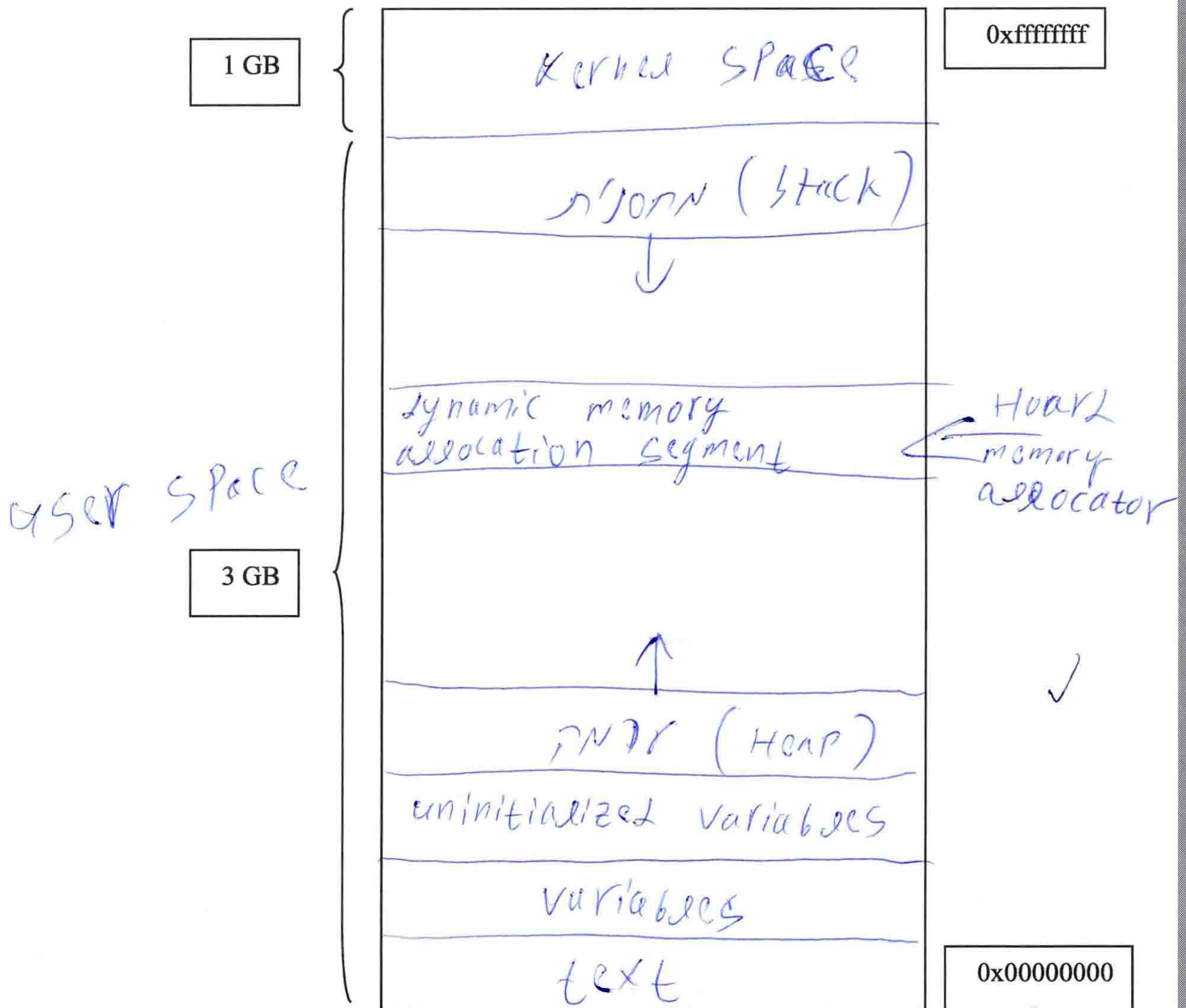






+ שאלה 8

היכן מקצה Hoard memory allocator את שטחי הזיכרון שגודלם יותר מ-S/2, כאשר S הוא גודל הסופר-בלוק? השלימו את הציור של מפת הזיכרון של תהליך בלינוקס והצביעו על מקום ההקצאות של Hoard memory allocator על גבי המפה. ציינו מהו תפקידו של כל סגמנט במפה.



Stack

המשך הבחינה בעמוד הבא







## חלק ג (20 נקודות)

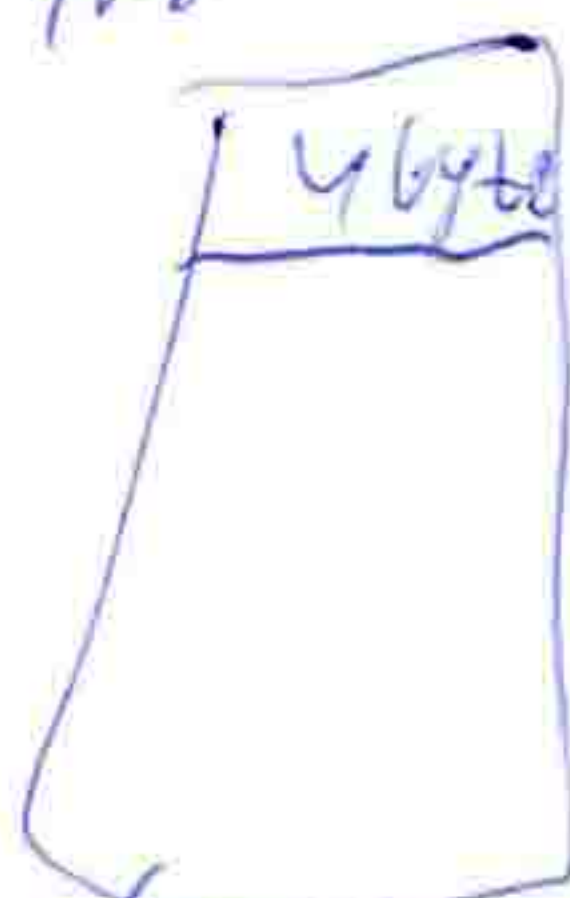
ענו על ארבע שאלות רב-ברירה (אמריקאיות) הבאות. משקל כל שאלה 5 נקודות.  
בכל שאלה יש לבחור את התשובה הנכונה ולהקיף בעיגול את אות התשובה שבחרתם.

+ שאלה 9

מערכת הקבצים של מערכת הפעלה מסוימת משתמשת בשיטת ה-I-node.

זכור:

ק"ב



אם כתיבות קבוצה קטנה היא  
 $2^8 = 256$   
 $2^{10} = 1024$   
 $2^2 = 4$

• גודל הבלוק במערכת הקבצים הוא 1 Kbyte

• כתובת הבלוק בדיסק היא 4 בתים (bytes)

• 12 שדות של ה-I-node יכולים להחזיק ישירות כתובת הבלוק בדיסק

• שדה נוסף אחד נועד להחזיק כתובת של ה-single indirect block

• עוד שדה נוסף אחד נועד להחזיק כתובת של ה-double indirect block

• ועוד שדה נוסף אחד נועד להחזיק כתובת של ה-triple indirect block

גודלו של קובץ מסוים במערכת הוא 1000 Kbyte. מהי כמות הבלוקים שדרושה להחזקת קובץ זה במערכת הקבצים (לא כולל את הבלוק שמכיל את ה-i-node של הקובץ)?

א. 1000

ב. 1005

ג. 1010

ד. 1011

+ שאלה 10

בחרו מהי הפעולה היקרה ביותר במונחים של מעברי בלוקים של הדיסק (disk block transfers) בהנחה שלא קיימים נתונים רלוונטיים בזיכרון המטמון (buffer cache):

א. פתיחת קובץ באמצעות open

ב. קריאת בלוק אחד באמצעות read

ג. קריאת תו אחד באמצעותgetc

ד. התשובות א ו-ב הן הנכונות

המשך הבחינה בעמוד הבא







+ שאלה 11

בחרו סיגנל (signal) אשר אי-אפשר להתעלם ממנו (באמצעות SIG\_IGN):

- א. SIGINT
- ב. SIGKILL
- ג. SIGSEGV
- ד. SIGALRM

+ שאלה 12

איזו פעולה מן הפעולות הבאות אפשר לבצע אך ורק במצב ראשוני (kernel mode) במערכת ההפעלה Linux?

- א. חסימת פסיקות החומרה (disabling hardware interrupts)
- ב. החלפת תהליכונים (thread switch) כאשר מדובר בספריית תהליכונים ברמת המשתמש
- ג. השמת ערך במשתנה גלובאלי
- ד. את כל שלוש הפעולות הנ"ל יש לאפשר אך ורק במצב ראשוני

בהצלחה!



