



מס' מבחן

שם הקורס: מערכות הפעלה

קוד הקורס: 10363

בחנית סמסטר: קיץ

השנה: 2017

מועד: 5/11/17

תאריך הבחינה:

שעת הבחינה:

משך הבחינה: 3 שעות

השאלון לא יבדק בתום הבחינה

ע"י המרצה

מרצה: גדי פסח

הוראות לנבחן:

- חומר עזר שימושי לבחינה:

כל חומר כתוב

- 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 השאלון הבחינה

מבנה הבחינה והנחיות לפתרון:

בבחינה 6 שאלות. יש לענות על כולם.

משקל השאלות:

שאלה 1: 20 שאלה 2: 20 שאלה 3: 20 שאלה 4: 10 שאלה 5: 15 שאלה 6: 15

שאלה 1

יש אנשים שרוצים לעלות על גשר מעל נהר כדי לצפות במשט ספינות. הגשר יכול לשאת עד 5000 ק"ג בבת אחת. לכן אם מישהו רוצה לעלות על הגשר וכתוצאה מכך המשקל הכולל על הגשר יעבור את המקסימום המותר (5000 ק"ג) אז עליו להמתין עד שאיש או אנשים ירדו מהגשר.

סעיף א: כתבו מוניטור Bridge שמבטיח שהמשקל הכולל על הגשר לא יחרוג מהמקסימום המותר.

המוניטור צריך לכלול את הפרוצדורות הבאות:

void getOnBridge (int weight), void getOffBridge (weight)

כל אחד מהאנשים מיוצג ע"י חוט. לפני שמישהו עולה על הגשר הוא קורא לפרוצדורה getOnBridge. לפני שהוא יורד מהגשר הוא קורא לפרוצדורה getOffBridge. לשתי הפרוצדורות יש ארגומנט אחד: המשקל של האדם בק"ג.

הגדירו משתנים ומשתני תנאי של המוניטור לפי הצורך. יש לתת ערך התחלתי למשתנים הרגילים. הפרוצדורות שעליכם לכתוב כולן קצרות מאוד.

סעיף ב: האם הפתרון שרשמתם בסעיף א מאפשר "הרעבה" כלומר תרחיש שבו מישהו שיכול היה לעלות על הגשר אינו עושה זאת (במשך זמן לא מוגבל) בגלל אנשים אחרים שעולים במקומו. הניחו שהתור של כל משתנה



תנאי מנוהל כ- FIFO כלומר כאשר עושים signal (או broadcast) למשתנה התנאי -- התהליך הוותיק ביותר בתור הוא שיכנס למוניטור ראשון.

## שאלה 2

השאלה עוסקת במערכת UNIX עם inodes נתונים ה- inodes הבאים

```
inode #1
  direct pointer: 158

inode #345:
  direct pointer: 970

inode #401
  direct pointer: 888

inode #444:
  direct pointer: 199
  direct pointer: 200

inode #567:
  direct pointer: 806

#inode #580:
  direct pointer: 967
  direct pointer: 979

#inode #901:
  direct pointer: 888
```

נתון גם התוכן של הבלוקים הבאים בפורמט נוח לקריאה. חלק מהבלוקים מכילים תוכן של directories. חלקם data blocks.

```
block 158:
  name: stam   inode#: 56
  name: foo    inode# 57
  name: one    inode#: 901
  name: two    inode#: 567
  name: three  inode#: 444

block 199:
  The name is Bond

block 200:
  James Bond

block 806:
  Afeka
```



block 888:  
 name: two      inode#: 345  
 name: three     inode#: 567  
 name: four      inode#: 765

block 967:  
 super man

block 970  
 name: three     inode#: 580  
 name: one       inode#: 155  
 name: stam      inode#: 822

block 979:  
 Tel Aviv.

השאלה היא:

בהנחה ש- inode מספר 1 הוא שלה- root directory. מה התוכן של הקובץ /one/two/three ?

בהנחה שבהתחלה רק ה- inode שלה- root directory נמצא בזכרון.

מי הם ה- inodes והבלוקים שיובאו מהדיסק במהלך האיתור של התוכן של הקובץ הנ"ל? יש לרשום את הסדר שבו יובאו ה- inodes והבלוקים מהדיסק.

### שאלה 3

השאלה עוסקת בזכרון וירטואלי. מרחב הכתובות של תהליך כולל שמונה דפים וירטואליים. ברגע מסוים טבלת הדפים של התהליך נראית כך (לא כל המידע מופיע כאן).

Virtual page #	Valid	Page frame #	Protection	Modified bit
0	1	38	read write	0
1	1	17	read write	1
2	1	25	read write	1
3	1	13	read	0
4	0	on disk		
5	1	14	read write	1
6	1	18	read execute	0
7	1	34	read	0

נניח שה TLB (Translation Lookaside Buffer) של המעבד כולל באותו רגע את הכניסות הבאות

Valid	Virtual Page#	Modified bit	Protection	Page frame #
1	0	0	read write	38
1	7	0	read	34
1	2	1	read write	25
1	6	0	read execute	18



נניח שכרגע מסגרות פיזיות 80, 81, 82 ... הן פנויות ובמקרה הצורך דפים וירטואליים שיטענו מהדיסק -- יטענו לתוכם. נניח עוד שאם יש צורך לפנות כניסה ב- TLB אז מפנים קודם את הכניסה הראשונה, לאחר מכן את השניה וכן הלאה.

נניח שדף 4 אמור להיות בעל הרשאות כתיבה וקריאה. מה יהיו השינויים (אם יהיו) בטבלת הדפים וב- TLB כאשר יתבצעו הגישות הבאות לזכרון (בסדר זה):

סעיף א: כתיבה לדף 0.

סעיף ב: קריאה מדף 4

סעיף ג: קריאה מדף 5

#### שאלה 4

This question is about safe states (like in the Banker's Algorithm). A system has four processes and four allocatable resources. The current allocation and resources still needed are as follows:

	Allocated	additional resources needed
Process A	1,0,0,1	2,0,0,1
Process B	0,0,0,0	0,0,2,3
Process C	0,0,1,0	<b>X</b> , 1,0,0
Process D	2,1,0,1	0,0,0,2

Available Resources = (2, 0, 1, 1)

What is the largest value of x for which this is a safe state ?

Give a short explanation.

#### שאלה 5

נניח שדרייבר עבור דיסק מקבל בקשות לקריאה מצילינדרים 72, 80, 10, 90 (לאו דווקא בסדר הזה).

תנו דוגמא שבה זה לא משנה אם הדרייבר פועל לפי האלגוריתם

First Come First Served או לפי האלגוריתם Shortest Seek First

או לפי אלגוריתם המעלית (לא מעגלי) כי כולם יתנהגו בצורה זהה בדוגמא זו.

עליכם לציין היכן ממוקם הזרוע של הדיסק בהתחלה,

מה סדר הגעת הבקשות (רלוונטי ל- First Come First Served)

ובאיזה כיוון נעה הזרוע בהתחלה (רלוונטי למעלית).

#### שאלה 6

. נניח שנעשה שימוש בשתי רמות של טבלאות דפים. כתובת וירטואלית כוללת 32 סיביות ומתחלקת לשלוש שדות. 10 הסיביות השמאליות ביותר הן אינדקס לטבלת הדפים ברמה העליונה. 9 הסיביות שאחריהן הן אינדקס לטבלת דפים ברמה שניה. הסיביות הנותרות הן offset לתוך הדף.

כל כניסה בטבלת דפים היא בגודל 8 בתים

סעיף א. מה הגודל של דף אחד ?



סעיף ב. תהליך משתמש בשמינית התחתונה ובשמינית העליונה של מרחב  
הכתובות הוירטואלי שלו. כמה זכרון תופסים כל טבלאות הדפים שלו ?

## בהצלחה!

כל הזכויות שמורות ©. 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

# פתרון



## פתרונות בחינה במערכות הפעלה סמסטר 2017 ג מועד XX

שאלה 1

יש אנשים שרוצים לעלות על גשר מעל נהר כדי לצפות במשט ספינות.

הגשר יכול לשאת עד 5000 ק"ג בבת אחת. לכן אם מישהו רוצה לעלות

על הגשר וכתוצאה מכך המשקל הכולל על הגשר יעבור את המקסימום המותר

(5000 ק"ג) אז עליו להמתין עד שאיש או אנשים ירדו מהגשר.

סעיף א: כתבו מוניטור Bridge שמבטיח שהמשקל הכולל על הגשר לא יחרוג

מהמקסימום המותר.

המוניטור צריך לכלול את הפרוצדורות הבאות:

void getOnBridge (int weight), void getOffBridge (weight)

כל אחד מהאנשים מיוצג ע"י חוטי. לפני שמישהו עולה על הגשר הוא

קורא לפרוצדורה getOnBridge. לפני שהוא יורד מהגשר הוא קורא

לפרוצדורה getOffBridge. לשתי הפרוצדורות יש ארגומנט אחד: המשקל של

האדם בק"ג.

הגדירו משתנים ומשתני תנאי של המוניטור לפי הצורך. יש לתת ערך התחלתי

למשתנים הרגילים.

הפרוצדורות שעליכם לכתוב כולן קצרות מאוד.

solutions 2017C XX 1

סעיף ב: האם הפתרון שרשמתם בסעיף א מאפשר "הרעבה" כלומר

תרחיש שבו מישהו שיכול היה לעלות על הגשר אינו עושה זאת (במשך זמן לא מוגבל) בגלל אנשים אחרים שעולים במקומו. הניחו שהתור של כל משתנה תנאי מנוהל כ- FIFO כלומר כאשר עושים signal (או broadcast) למשתנה התנאי -- התהליך הוותיק ביותר בתור הוא שיכנס למוניטור ראשון.

פתרון

סעיף א

monitor Bridge {

int total = 0; /\* total weight on bridge \*/

int MAX = 5000;

condition ok\_to\_get\_on\_bridge;

void getOnBridge (int weight)

{

while (total + weight > MAX)

wait (ok\_to\_get\_on\_bridge);

total = total + weight;

}

void getOffBridge (int weight)

{

total = total - weight;

broadcast (ok\_to\_get\_on\_bridge);

solutions 2017C XX 2

פתרון  
10303  
C



}

## סעיף ב'

יש אפשרות להרעבה. יתכן תרחיש שבו בכל פעם שמישהו מתעורר (בגלל ה- broadcast) והוא עורך בדיקה האם הוא יכול לעלות על הגשר התשובה היא לא (כי משקלו גדול מדי). אילו היה עורך את הבדיקה ברגע מתאים יותר התשובה הייתה כן.

## שאלה 2

השאלה עוסקת במערכת UNIX עם inodes  
נתונים ה- inodes הנאים

inode #1

direct pointer: 158

inode #345:

direct pointer: 970

inode #401

direct pointer: 888

inode #444:

direct pointer: 199

direct pointer: 200

inode #567:

direct pointer: 806

#inode #580:

direct pointer: 967

direct pointer: 979

#inode #901:

direct pointer: 888

נתון גם התוכן של הבלוקים הנאים בפורמט נוו לקריאה. חלק מהבלוקים מכילים  
תוכן של directories, חלק מהם blocks.data.

block 158:

name: stam inode#: 56

name: foo inode# 57

name: one inode#: 901

name: two inode#: 567

name: three inode#: 444

block 199:

The name is Bond



block 200:

James Bond

block 806:

Afeke

block 888:

name: two inode#: 345

name: three inode#: 567

name: four inode#: 765

block 967:

super man

block 970

name: three inode#: 580

name: one inode#: 155

name: stam inode#: 822

block 979:

Tel Aviv.

השאלה היא:

בהנחה ש- inode מספר 1 הוא של ה- root directory, מה התוכן של  
קובץ /one/two/three ?

בהנחה שבהתחלה רק ה- inode של ה- root directory נמצא בזכרון.

מי הם ה- inodes והבלוקים שינאמו מהדיסק במהלך האיתור של התוכן של הקובץ תחיל  
?

יש לרשום את הסדר שבו ינאמו ה- inodes והבלוקים מהדיסק.

### פתרון

התוכן הוא Super man Tel Aviv.

סדר הנאמה מהדיסק:

בלוק 158 (ה- root directory)

(/one של) inode #901

בלוק 888 (התוכן של /one)

(/one/two של) inode #345

בלוק 970 (התוכן של /one/two)

inode #580 (של /one/two/three)

בלוקים 979, 967, ה- data של /one/two/three)

### שאלה 3

השאלה עוסקת בזכרון וירטואלי. מרחב הכתובות של תהליך כולל שמונה דפים  
וירטואליים. ברגע מסוים טבלת הדפים של התהליך נראית כך (לא כל המידע  
מופיע כאן).



Virtual page #	Valid	Page frame #	Protection	Modified bit
0	1	38	read write	0
1	1	17	read write	1
2	1	25	read write	1
3	1	13	read	0
4	0	on disk		
5	1	14	read write	1
6	1	18	read execute	0
7	1	34	read	0

נניח שה TLB (Translation Lookaside Buffer) של המעבד כולל באותו רגע את הכניסות הבאות

Valid	Virtual Page#	Modified	Protection	Page
-------	---------------	----------	------------	------

solutions 2017C XX 7

	bit		frame #
1	0	read write	38
1	0	read	34
1	1	read write	25
1	0	read	18

נניח שכרגע מסגרות פיזיות 80, 81, 82 ... הן פנויות ובמקרה הצורך דפים וירטואליים שיטענו מהדיסק -- יטענו לתוכם. נניח עוד שאם יש צורך לפנות כניסה ב- TLB אז מפנים קודם את הכניסה הראשונה, לאחר מכן את השנייה וכן הלאה.

נניח שדף 4 אמור להיות בעל הרשאות כתיבה וקריאה.

מה יהיו השינויים (אם יהיו) בטבלת הדפים וב- TLB כאשר יתבצע

הגישות הבאות לזכרון (בסדר זה):

סעיף א: כתיבה לדף 0.

סעיף ב: קריאה מדף 4

סעיף ג: קריאה מדף 5

פתרון

סעיף א. כתיבה לדף 0 זה הו TLB. כניסה של דף 0 ב- TLB מעדכנים את ה- bit Modified ל- 1. אין שינוי בטבלת הדפים.

solutions 2017C XX 8



**סעיף ב. קריאה מדף 4**

זה יגרום ל- page fault שבעקבותיו יטען חודף לזכרון והכניסה המתאימה בטבלת הדפים תהיה:

Virtual page = 4, Page frame = 80, Protection = read, write  
Modified = 0,

הכניסה תטען גם ל- TLB לכניסה הראשונה:

Valid = 1 Virtual page = 4, Modified = 0, Protection = read  
write, Page Frame = 80.

בכניסה בטבלת הדפים של דף וירטואלי מספר 0 (שיפונה מה- TLB כדי לענות מקום לדף 4) ה- bit Modified תעודכן ל- 1.

**סעיף ג:** קריאה מדף 5: זה TLB miss

הכניסה המתאימה לדף 5 תטען מטבלת הדפים לכניסה השניה של ה- TLB:  
Valid = 1 Virtual page = 5, Modified = 1, Protection = read  
write, Page Frame = 14.

בכניסה בטבלת הדפים של דף וירטואלי מספר 7 (שיפונה מה- TLB כדי לענות מקום לדף 5) אין צורך להכניס שינוי (ה- bit Modified נשאר אפס).

**שאלה 4**

This question is about safe states (like in the Banker's Algorithm). A system has four processes and four allocatable resources. The current

allocation and resources still needed are as follows:

	Allocated	additional resources needed
Process A	1,0,0,1	2,0,0,1
Process B	0,0,0,0	0,0,2,3
Process C	0,0,1,0	<b>X</b> ,1,0,0
Process D	2,1,0,1	0,0,0,2

Available Resources = (2, 0, 1, 1)

What is the largest value of x for which this is a safe state ?

Give a short explanation.

**פתרון**

A רץ ראשון ומחזיר את המשאבים שלו. עכשיו

Available = (3, 0, 1, 2)

D רץ שני ומחזיר את המשאבים שלו. עכשיו

Available = (5, 1, 1, 3)



אם  $5 \leq x < C$  יוכל לרוץ (אחרת יהיה קפאון). עכשיו

$$\text{Available} = (5, 1, 2, 3)$$

עכשיו B יוכל לרוץ.

התשובה היא: הערך המקסימלי של  $x$  עבורו המצב יהיה בטוח הוא 5.

#### שאלה 5

נניח שדרייבר עבור דיסק מקבל בקשות לקריאה מצילינדריים

72, 80, 10, 90 (לאו דווקא בסדר הזה).

תנו דוגמא שבה זה לא משנה אם הדרייבר פועל לפי האלגוריתם

First Come First Served או לפי האלגוריתם Shortest Seek First

או לפי אלגוריתם המעלית (לא מעלית) כי כולם יתנהגו בצורה זהה בדוגמא זו.

עליכם לציין היכן ממוקם הזרוע של הדיסק בהתחלה,

מה סדר הגעת הבקשות (רלוונטי ל- First Come First Served)

ובאיזה כיוון נעה הזרוע בהתחלה (רלוונטי למעלית).

#### פתרון

סדר הגעת הבקשות (משמאל לימין): 10, 72, 80, 90

הזרוע ממקמת בהתחלה בצילינדר 10 (או בצילינדר נמוך יותר) והיא נעה כלפי מעלה (משמאל לימין) בצילינדרים המעלית). יש פתרון נוסף סימטרי (הזרוע

תנוע בכיוון ההפוך).

#### שאלה 6

נניח שנעשה שימוש בשתי רמות של טבלאות דפים. כתובת וירטואלית

כוללת 32 סיביות ומתחלקת לשלוש שדות. 10 הסיביות השמאליות ביותר הן אינדקס לטבלת הדפים ברמה העליונה. 9 הסיביות שאחריהן הן אינדקס לטבלת דפים ברמה שניה. הסיביות הנותרות הן offset לתוך הדף.

כל כניסה בטבלת דפים היא בגודל 8 בתים

סעיף א. מה הגודל של דף אחד?

סעיף ב. תהליך משתמש בשמיניות התחתונה ובשמיניות העליונה של מרחב הכתובות הוירטואלי שלו. כמה זכרון תופסים כל טבלאות הדפים שלו?

#### פתרון

סעיף א. ה- offset מיוצג עי"י  $32 - 10 - 9 = 13$  סיביות. מכאן שכל דף

כולל  $2^{13}$  בתים כלומר כל דף הוא בגודל 8 kilobyte.

סעיף ב.

יש  $2^{10}$  כניסות בטבלת הדפים ברמה העליונה. כל כניסה בטבלת כוללת 8 בתים

ומכאן שגודל הטבלה  $2^{13} * 8 = 2^{10}$  בתים (כלומר 8 kilobyte)

יש  $2^9 = 512$  כניסות בכל טבלת דפים ברמה השניה.

כל כניסה כוללת 8 בתים ומכאן שהגודל של כל טבלה כזאת הוא

$$4 \text{ kilobyte} = 2^{12} \text{ bytes} = 8 \text{ bytes} * 2^9$$

רבע מהכניסות של טבלת הדפים ברמה העליונה מכילות מצביעים

לטבלאות דפים ברמה שניה. (אלו הכניסות שמתאימות לכתובות שמחיליות



(בצד שמאל) ב- 000 והכתובות שמתחילות ב- 111. אלו מהוות את השמיניות

התחתונה והשמיניות העליונה בהתאמה של מרחב הכתובות).

שאר הכניסות בטבלת הדפים ברמה העליונה הן לא valid.

לכן יש בסך הכל  $2^8 = 2^{10}/4$  טבלאות דפים מרמה שניה.

בנוסף לכך יש טבלת דפים אחת מרמה ראשונה. בסה"כ הזכרון שתופסות כל טבלאות הדפים הוא בגודל

$$2^{13} + 2^8 * 2^{12} \text{ bytes} = 2^{13} + 2^{20} \text{ bytes} =$$

$$1 \text{ Megabyte} + 8 \text{ Kilobyte}$$