

מס' שאלון - 515
20
בפברואר 2020

מס' מועד 84

שאלון בחינת גמר
20594 - מערכות הפעלה

משך בחינה: 3 שעות

בשאלון זה 9 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 (16 נקודות)

(6 נק') א. אם יש לנו זיכרון RAM אינסופי, האם עדיין יש יתרון במנגנון זיכרון וירטואלי?

הסבירו את תשובתכם.

כן, כי יש יתרון בזיכרון וירטואלי. לדוגמה, אם יש לנו זיכרון וירטואלי, נוכל להשתמש בזיכרון וירטואלי כדי לשמור על נתונים חשובים, גם אם הזיכרון הפיזי מלא. זה מאפשר לנו להשתמש בזיכרון וירטואלי כדי לשמור על נתונים חשובים, גם אם הזיכרון הפיזי מלא.

(10 נק') ב. נתון קטע קוד הבא שרץ מעל מערכת הפעלה Linux:

```
#define N SOME_NUMBER
#define M ANOTHER_NUMBER
```

```
int X[N];
for (j=0; j<30; ++j){
    for (i = 0; i < N; i += M){
        X[i] = X[i] + 1;
    }
}
```

נניח ש TLB מתמלא באופן מעגלי ונניח כי יש בו 64 כניסות. אילו ערכים של N ושל M יגרמו ל-TLB-miss בכל איטרציה של הלולאה הפנימית? (הניחו כי TLB מתרגם כתובת דף וירטואלי לכתובת דף פיזי וגודל הדף הוא

4KB)

על פי הנתונים, $N = 65M$ ו- $M = 4096$ (4KB). לכן, בכל איטרציה של הלולאה הפנימית, נקבל TLB-miss בגלל העובדה ש- M אינו חלק של N .

שאלה 2 (15 נקודות)

אלגוריתם lottery scheduling הינו אלגוריתם תזמון הסתברותי. כל תהליך במערכת מקבל מספר כלשהו של "כרטיסי הגרלה" (לכל כרטיס מספר ייחודי); בכל שלב, האלגוריתם מגריל מספר של אחד מכרטיסי ההגרלה שחולקו, והתהליך שמחזיק בכרטיס הגרלה זה מקבל את המעבד לפרק זמן קצוב.

5) (נק') א. נניח כי לאלגוריתם יש 20 כרטיסים. כיצד עליו לחלק אותם בין ארבעה תהליכים (A, B, C, D) כך שהתהליכים יקבלו 10%(A), 5%(B), 60%(C), 25%(D) בהתאמה?

$\rho'_{\text{כח}} = 12 - C$ $\rho'_{\text{כח}} = 2 - A$

 $\rho'_{\text{כח}} = 3 - D$ $\rho'_{\text{כח}} = 1 - B$

ב. כיצד ניתן להגיע באמצעות אלגוריתם זה לביצועים הדומים לאלו המושגים על-ידי אלגוריתם round-robin?

[illegible]

ג. איך ניתן להגיע באמצעות אלגוריתם זה לביצועים הדומים לאלו המושגים על-ידי אלגוריתם shortest remaining time first?

(ניתן להניח כי יש preemption, התהליכים לא מגיעים יחד, אבל זמן הביצוע ידוע כאשר התהליך מגיע.)

לחלק את הכתובים היום הפוך עמך הרבה חזרה

[illegible]

שאלה 3 (24 נקודות)

נתונה מערכת עם 8 דיסקים של 1TB כל אחד. בסעיפים א-ה ננסה לעשות השוואה בין RAID0, RAID1 ו-RAID5

א. (6 נק') עבור כל אחת ממערכות RAID ציינו כמה נתונים נוכל לשמור במערכת זו.

8TB	בליט	ג'רמ' 8TB	RAID0	✓
8TB	מחלק 4TB "חול" ואז 4TB	ג'רמ' 8TB	RAID1	✓
8TB	מחלק 7TB Data ואז 1TB	ג'רמ' 8TB	RAID5	✓

ב. (6 נק') נניח שאנחנו רוצים לבצע רק קריאות של בלוקים בודדים ונניח שקצב הקריאה של כל דיסק הינו 100 בלוקים לשנייה. כמו כן נניח שהקריאות שלנו אקראיות. עבור כל אחת ממערכות RAID ציינו מהו קצב הקריאה המקסימאלי שנוכל לצפות במערכת זו.

800 בלוק'ס	לוקח גל פסק	RAID0	✓
800 בלוק'ס	גל פסק / גל פסק	RAID1	
800 בלוק'ס	גל פסק	RAID5	

ג. (6 נק') נניח שאנחנו רוצים לבצע רק כתיבות של בלוקים בודדים ונניח שקצב כתיבות של כל דיסק הינו 100 בלוקים לשנייה. כמו כן נניח שהכתיבות שלנו אקראיות. עבור כל אחת ממערכות RAID ציינו מהו קצב כתיבות המקסימאלי שנוכל לצפות במערכת זו.

800 בלוק'ס	RAID0	✓
400 בלוק'ס ג'רמ' 400 ואז 400	RAID1	✓
400 בלוק'ס ג'רמ' 400 ואז 400	RAID5	✗

2-

ד. (6 נק') מהו המספר המינימאלי של דיסקים שצריכים ליפול כדי שייתכן איבוד נתונים

1 - 8 בלוק'ס ג'רמ' 8TB	RAID0	✓
2 - 4 בלוק'ס ג'רמ' 4TB ואז 4TB	RAID1	
2 - 7 בלוק'ס ג'רמ' 7TB Data ואז 1TB	RAID5	

תזכורת:**RAID0**

בשיטה זו נתונים פזורים על פני דיסקים של מערך בפיסות (strips) בגודל קבוע. כלומר הנתונים של הדיסק הלוגי שמערכת ההפעלה "רואה" למעשה מחולקים לפיסות ומפוזרים על דיסקים שונים של המערך.

RAID1

שיטת level 1 מכפילה את כמות הדיסקים שהיו ב- RAID level 0 ומשתמשת בהם לגיבוי.

RAID5

level 5 משתמשת ב-parity strips כאשר הם מפוזרים על פני כל הדיסקים של המערך (למשל בשיטת ה-round-robin).

חלק ב (25 נקודות)

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

שאלה 4

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

link - קישור
soft-link - קישור רך
hard-link - קישור קשה
path - דרך
counter - מונה
pointers - מצביעים
sp - stack pointer

שאלה 5

הסבירו מה נעשה בשורות קוד הבאות?

```
movl current_thread, %eax
movl %esp, (%eax)
```

כאשר

```
typedef struct thread thread_t, *thread_p;

struct thread {
    int sp; /* current stack pointer */
    char stack[STACK_SIZE]; /* the thread's stack */
    int state; /* running, runnable, waiting */
};

thread_p current_thread;
```

thread_t struct
sp - stack pointer
stack - thread's stack
state - running, runnable, waiting

שאלה 6

והסבירו את עקרון הלוקליות בזמן ובמקום.

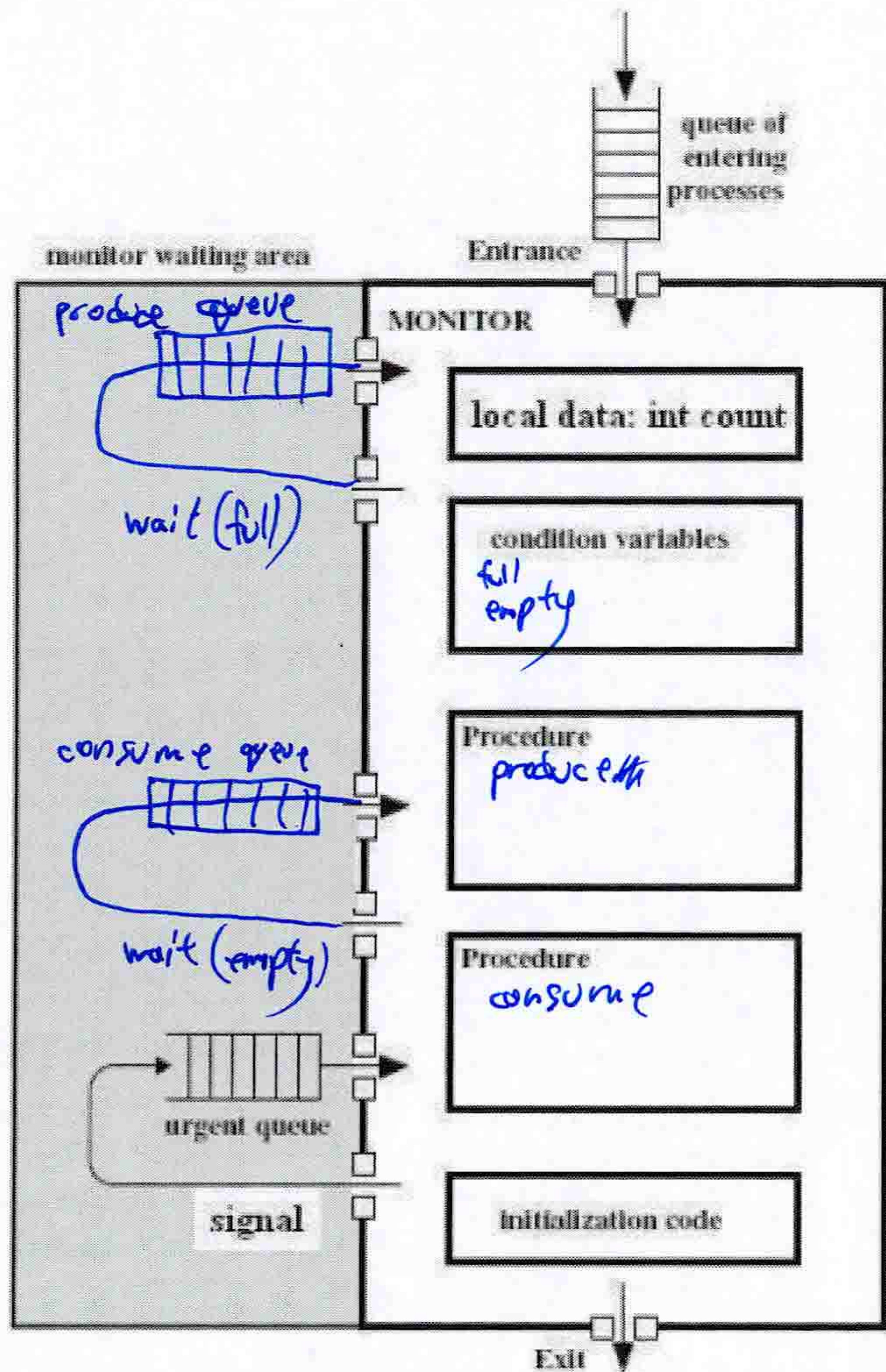
עקרון הלוקליות בזמן ובמקום.
 עקרון הלוקליות בזמן: כל תהליך צריך להסתיים לפני שיתחיל תהליך אחר.
 עקרון הלוקליות במקום: כל תהליך צריך להסתיים לפני שיתחיל תהליך אחר.
 שאלה 7

מדוע נועלים דפים בזיכרון בעת העברת נתונים על ידי DMA (Direct Memory Access)?

ע"י נעילה של דפים בזיכרון, נמנעם מהעברת נתונים לאי-מצב.
 במהלך DMA, כל דף שצריך להועברת נתונים צריך להיות נעול.
 שאלה 8

שאלה 8

השלימו בשרטוט את תור התהליכים הממתנינים על משתנה תנאי כלשהו במבנה פיקוח (monitor).



חלק ג (20 נקודות)

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

שאלה 9

להלן טבלת הדפים של תהליך רץ. כל המספרים הם עשרוניים. כל הכתובות הן כתובות ב-bytes.
גודל דף הוא 1024 bytes.

Virtual Page Number	Valid Bit	Reference Bit	Modify Bit	Page Frame Number
0	1	1	0	4
1	1	1	1	7
2	0	0	0	-
3	1	0	0	2
4	0	0	0	-
5	1	0	1	0

חשבו את הכתובות הפיסיות, אם זה אפשרי, של הכתובת הווירטואלית 5499 :

5499 - 5.625 = 379

- א. 379
- ב. 173
- ג. 28
- ד. 14

שאלה 10

בחרו את 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' וב' הן הנכונות

שאלה 11

מדוע תהליכים באים באינטראקציה עם מערכת ההפעלה באמצעות traps?

- א שימוש ב traps מהיר יותר מאשר קריאה ישירה לפונקציות בקוד של מערכת ההפעלה.
- ב השימוש ב traps מונע conditions race כאשר מספר תהליכים משתמשים במשאב משותף בו זמנית.
- ג זאת הדרך היחידה שבאמצעותה ניתן לאפשר למספר תהליכים לחלוק קוד משותף.
- ד זאת הדרך שבאמצעותה שירותי מערכת הפעלה שניתנים לתהליך רץ יכולים ליהנות מהרשאות בלתי מוגבלות.

מחוז סאודי-ערבית מוגבלות:

trap מסייד את המעבד user-mode ל-kernel-mode כפי לרכש פסל/ שביש
אולי/ ~~לעצמה~~ ~~מחשבתה~~ . אף איתר להימלט/ 14 סביבה איתר/ ~~פסל/~~ שביש

שאלה 12
עברה פ' היתה הקריאה ב' .

לפניכם פסאודו-קוד של משחק רובוטים שבו 3 רובוטים בצבעים (אדום, כחול, ירוק) מבצעים תזוזות בסדר כלשהו:

<pre> int sem[3]; sem[0] = 1; sem[1] = 1; sem[2] = 1; </pre>		
<pre> R_Robot (){ while(true) { down(sem[0]); <Make Move> up(sem[1]); } } </pre>	<pre> G_Robot (){ while(true) { down(sem[1]); <Make Move> up(sem[2]); } } </pre>	<pre> B_Robot (){ while(true) { down(sem[2]); <Make Move> up(sem[0]); } } </pre>
<pre> int main(){ run_new_thread(R_Robot); run_new_thread(G_Robot); run_new_thread(B_Robot); } </pre>		

האם סדר התזוזות של הרובוטים נקבע על ידי השימוש בסמפורים כדלהלן?
אם כן, מהו הסדר?

- א. כן. אדום, ירוק, כחול וחוזר חלילה
- ב. כן. אדום, כחול, ירוק וחוזר חלילה
- ג. כן. אדום, כחול, אדום, כחול, ירוק וחוזר חלילה
- ד. לא. הסדר ייקבע על ידי מתזמן (scheduler)

בהצלחה!