

גליון תשובות לשאלות רב-ברריות

הקף במצגל את התשובה שבחרת ולכל שאלה יש רק תשובה נכונה אחת. אם תרצה לבטל תשובה שבחרת, סמן עליה X.

דוגמה לתשובה שבחרת: א ב (2) ד ה ו ז ח ט
דוגמה לתשובה שבטלת: א ב ג ד (X) ו ז ח ט

שאלה									
תשובה									
ט	ח	ו	ז	ה	ד	ב	א	1	✓
ט	ח	ו	ז	ה	ד	ב	א	2	X
ט	ח	ו	ז	ה	ד	ב	א	3	✓
ט	ח	ו	ז	ה	ד	ב	א	4	✓
ט	ח	ו	ז	ה	ד	ב	א	5	X
ט	ח	ו	ז	ה	ד	ב	א	6	X
ט	ח	ו	ז	ה	ד	ב	א	7	✓
ט	ח	ו	ז	ה	ד	ב	א	8	✓
ט	ח	ו	ז	ה	ד	ב	א	9	X
ט	ח	ו	ז	ה	ד	ב	א	10	✓
ט	ח	ו	ז	ה	ד	ב	א	11	✓
ט	ח	ו	ז	ה	ד	ב	א	12	✓
ט	ח	ו	ז	ה	ד	ב	א	13	✓
ט	ח	ו	ז	ה	ד	ב	א	14	✓
ט	ח	ו	ז	ה	ד	ב	א	15	X
ט	ח	ו	ז	ה	ד	ב	א	16	✓
ט	ח	ו	ז	ה	ד	ב	א	17	✓
ט	ח	ו	ז	ה	ד	ב	א	18	X
ט	ח	ו	ז	ה	ד	ב	א	19	✓
ט	ח	ו	ז	ה	ד	ב	א	20	✓
ט	ח	ו	ז	ה	ד	ב	א	21	
ט	ח	ו	ז	ה	ד	ב	א	22	
ט	ח	ו	ז	ה	ד	ב	א	23	
ט	ח	ו	ז	ה	ד	ב	א	24	
ט	ח	ו	ז	ה	ד	ב	א	25	

לשימוש פנימי

מספר התשובות הנבונות _____

הציון _____

שם הבודק _____

10, 9, 6, 2

שאלה 1

בחרו מתוך הרשימה מזה המריט שמשותף לתהליכים (threads) שרצים באותו מרחב כתובות (באותו תהליך):

- א. מחסנית (stack) ×
- ב. מצב התהליך (running, ready etc) ×
- ג. program counter ×
- ד. משתנים גלובליים

שאלה 2

מזה הפלט שישגשג יכול להיווצר כתוצאה מההרצה של התוכנית הבאה:

```
int is_terminated = 0;

void sig_handler(int signo){
    printf("3");
}

main(){
    int status;
    signal(SIGCHLD, sig_handler);
    if (fork() == 0){
        is_terminated = 1;
        exit(0);
    }
    if (is_terminated != 1){
        pause(); // waits for a signal
        printf("1");
    }
    wait(&status);
    printf("2");
}
```

3

1

2

- הערות: (1) ניתן להניח כי כל קריאות המערכת "מצלחות"
 (2) ניתן להניח כי הדפסת תו בודד היא פעולה אטומית
 (3) יש לקרוא את הפלטים משמאל לימין

א. 312

ב. 32

ג. 3

ד. אף תשובה קודמת איננה אפשרית

שאלה 3

מערכת זמן אמת (real-time system) מסוגלת לתזמן את המשימות הבאות:

Task	Period (קיימפונת של המשימה)	Execution time (זמן)
A	8	1
B	32	2
C	8	1
D	16	3
E	8	1

כמה משימות מסוג E ניתן לתזמן (בגרסה לאחל שכבר קיימות), כך שהמערכת תוכל עדיין לתזמן את כל המשימות? ניתן לחנח כי זמן החלפת התהליכים (context switch) הוא זניח.

$$H = \frac{32}{2} \leftarrow 32 \text{ Period } 320$$

- א. 0
ב. 1
ג. 3
ד. 6

שאלה 4

מניעת הקיפאון (deadlock) ע"י תקיפת חוקאי "מניעת הדדיות" באמצעות שיתוף משאבים איננה ישימה בכל המקרים כי:

mutual exclusion

א. לא כל משאב ניתן לשיתוף ~~CD-ROM~~ ~~printer~~

ב. גרסת לכניסה למצב לא בטוח (unsafe state)

ג. גרסת לכניסה למצב בטוח (safe state) X

ד. אף תשובה קודמת איננה נכונה

שאלה 5

מבנה פיקוח (monitor) מספק שתי פונקציות לעבודה עם condition variables: wait ו-signal. במימוש של מבנה פיקוח על פי Hoare הוגע שתהליך אשר קרא לפונקציה signal יושחז באופן מיידי ובמקום ירוץ התהליך שזה עתה חזר. מדוע Hoare הציע מימוש זהו בחרו בתשובה הנכונה ביותר.

א. מימוש זה מונע היווצרות של מצבי דשדוש (thrashing) בעת שתהליכים מבצעים פרוצדורות

של monitor. *...thrashing* $\forall x$

ב. מימוש זה מונע היווצרות של מצבי קיטאון (deadlock) בין תהליכים שמשתמשים בפונקציות

של monitor.

ג. מימוש זה מבטיח שרק תהליך אחד ייסצא ב monitor.

ד. כל שלוש התשובות הניל נכונות. \times

המשך הבחינה בעמוד הבא

שאלה 6

למניסם מתרון של מניית יצרן-צרכן באמצעות מבנה מיקוח (monitor). בחרו באיזה מקרה/באיילו מקרים יחידה התחרון עדיין נכון כאשר ההנחה היא שבעקבות signal כל התחליבים אשר היו במצב condition variable ורובים להיות ready to run.

```

monitor ProducerConsumer
condition full, empty;
integer count;

procedure insert(item: integer)
begin
    if count = N then wait(full); // full
    insert_item(item); // insert
    count := count + 1; // increment
    if count = 1 then signal(empty);
end

function remove :integer
begin
    if count = 0 then wait(empty); // empty
    remove = remove_item(); // remove
    count = count - 1; // decrement
    if count = N-1 then signal(full);
end;

count = 0;

end monitor;

procedure producer
begin
    while true do
        begin
            item = produce_item; // produce
            ProducerConsumer.insert(item); // procedure
        end
    end

end

procedure consumer
begin
    while true do
        begin
            item = ProducerConsumer.remove; // consume
            consume_item(item); // function
        end
    end

end
    
```

- א. יצרן-צרכנים
- ב. יצרנים-צרכן
- ג. יצרנים-צרכנים
- ד. כל התשובות חקודמות תן נכונות

שאלה 7

מדוע bit referenced המציינ כי התרחשה התייחסות לדף אינו חייב להיות ממוקם ב TLB (translation lookaside buffer)?

- מכיוון שהביט ממוקם באחד האוגרים (registers) של CPU.
- מכיוון שאם מגננון ניהול הזיכרון משתמש ב TLB, אזי אין צורך באלגוריתמים לחלפת דפים אשר מתבססים על ה bit referenced.
- לכל דף אשר נמצא ב TLB מובטח כי אירעה אליו התייחסות.
- כל התשובות הקודמות נכונות.

reference bit = 1

שאלה 8

כאשר קצב ה-page faults במערכת גבוה, מערכת החמלה נוקטת בפעולה הבאה:

- מפעילה אלגוריתם להוצאת דפים (page replacement algorithm).
- מבצעת swap in של תחילת (תחיליכים) לתגברת קצב ה page faults.
- מקטנה את כמות הזיכרון המוקצה לשירות של TLB (translation lookaside buffer).
- אף תשובה קודמת איננה נכונה.

שאלה 9

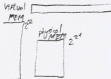
במערכת עם ניהול זיכרון באמצעות דפדף (paging) נתון כי:

- כתובת מדרומה (virtual address) הינה בת 32 bits
- אורך מילת הזיכרון הוא 1 byte
- גודל הזיכרון הפיזי הוא 2048 Mbytes
- גודל הדף הוא 4 Kbytes

$$2.68 = 2048 \text{ Mbytes}$$

$$4096 \text{ b} = 4 \text{ Kbytes}$$

מתי כמות הסיביות הדרושה לתחזוקת מספר המסגרות של הזיכרון הפיזי?



$$2048 \text{ Mbytes} \times 1024 \text{ k} = 2,097,452 = 2.68 \times 2^{27}$$

$$\frac{2^{32}}{2^{11}} = 2^{21}$$

א. 19

ב. 18

ג. 17

ד. 16

שאלה 10

נסמן ב-p את ההסתברות לכך שתהליך מחכה לסיימה של פעולת I/O ו-b נסמן את מספר התהליכים בזיכרון. כדי להגיע לעסיחה $1-p$ המינימלית את ניצול ה-CPU יש להניח את החתחת (הנחות) הבאה (הבאות):

- התהליכים הם בלתי תלויים זה בזה
- התהליכים הם תלויים זה בזה
- אלגוריתם התזמון הוא אלגוריתם ה-RR (round-robin)
- אלגוריתם התזמון הוא האלגוריתם המבוסס על עדיפויות משתנות
- שתי תשובות מבין התשובות הקודמות הן נכונות

שאלה 11

תהליך במצב zombie במערכת הפעלה UNIX חינו:

- כל תהליך שאינו מתקדם עקב מחסור במשאב מסוים
- כל תהליך אשר תהליך האב שלו "מת" (סיים את עבודתו)
- כל תהליך שקיבל סיגנל SIGKILL
- כל תהליך שסיים את עבודתו אך תהליך האב שלו עדיין רץ ולא ביצע wait

שאלה 12

מחר מספר הכניסות (entries) בטבלת הדפים המהפכת (inverted page table):



- מספר הדפים בזיכרון המדומה
- מספר המסגרות (frames) בזיכרון הפיזי
- מספר הדפים המכניסילי שיוכל לדרוש תהליך
- מספר התהליכים המכניסילי שיוכלים להתבצע בו-זמנית במערכת

שאלה 13

אילו מחסונות הבאות לגבי פסיקות חוסרה במערכת הפעלה UNIX איננה נכונה:

א. פסיקות חוסרה נוצרות ע"י interrupt service routine

ב. פסיקות חוסרה יכולה להיווצר ע"י חתקן I/O

ג. פסיקות חוסרה יכולה להיווצר ע"י שערן

ד. אף תשובה קודמת איננה נכונה

שאלה 14

נתונה מחזורות התייחסויות (reference string) הבאה:

← →
0 1 2 3 0 1 4 0 2 3 4

מספר הדפים שהוקצו לתחליף הוא 3. בחינתן שאלגוריתם לחלפת דפים הוא FIFO, מהו מספר

פסיקות דף (page faults) שמתרחש כאשר מדובר ב demand paging?

?

0	1	2	3	0	1	4	0	2	3	4
0	1	2	3	0	1	4	0	2	3	4
0	1	2	3	0	1	4	0	2	3	4
0	1	2	3	0	1	4	0	2	3	4

P P P P - P P P

א. 4

ב. 5

ג. 7

ד. 9

האם ישנה דרך אחרת לחשב את מספר הפגות הדף? - demand paging

שאלה 15

נתונה FAT עם קובץ A שמתחיל בבלוק 2, קובץ B שחלוק חראשון שלו הוא 9 וקובץ C שמתחיל

בבלוק 8. האם ה FAT עקבית (consistent)?

Block	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18
Next		15	12	11	18	-1	16		13	01	03	-1	06	04		10	5	-1	17
block																			

2 → 12 → 6 → 16 → 5 → -1.

9 → 1 → 15 → 10 → 8 → 4 → -1.

8 → 8 → 04 → 16 → 4 → -1.

א. ה- FAT עקבית

ב. ה- FAT איננה עקבית מכיוון של-A ו B ישנו בלוק משותף

ג. ה- FAT איננה עקבית מכיוון של-A ו C ישנו בלוק משותף

ד. ה- FAT איננה עקבית מכיוון של-B ו C ישנו בלוק משותף

שאלה 16

סגמנט ה-data של תהליך במערכת הפעלה UNIX מכיל:

- את חוראות המכונה של התוכנית המתבצעת \times *text segment*
- את המשתנים הגלובאליים של התוכנית *data segment*
- את המשתנים האוטומאטיים (automatic variables) \times *?*
- אף תשובה קודמת איננה נכונה \times

שאלה 17

פונקציה dup במערכת UNIX משמשת ל:

- יצירת תהליך חדש *fork*
- קריאה מתוך pipe
- כתיבה לתוך pipe
- אף תשובה קודמת איננה נכונה \times

שאלה 18

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

- גודל הבלוק במערכת הקבצים הוא 4 Kbytes 4096 bytes
- כתובת הבלוק היא 16 בתים (bytes) 128 bit
- 12 שדות של ה-I-node יכולים לתחזיק ישירות כתובת הבלוק בדיסק
- שדה נוסף אחד נועד לתחזיק כתובת של ה-single indirect block
- שדה נוסף אחד נועד לתחזיק כתובת של ה-double indirect block
- שדה נוסף אחד נועד לתחזיק כתובת של ה-triple indirect block

חשבו מהו הגודל המקסימאלי של קובץ במערכת הקבצים הזו. (עלו את התוצאה ביחידות של

Gbytes כלפי מטה)

$$12 \times 128 \text{ bit} \times 4096 \text{ bytes} = 6.251,456 \text{ bytes}$$

נכונה

- 60 Gbytes
- 61 Gbytes
- 62 Gbytes $\textcircled{3}$
- 64 Gbytes

שאלה 19

מתאמי התקנים (device drivers) במחשב יכולים להיות משני סוגים: מתאמי התקנים שמבצעים בדיקות בדבר סיום עבודת התקן ומתאמי התקנים שמבצעים מתקנים לייצר interrupt בסיום עבודת התקן. מצו החיסרון המובהק של מתאמי התקנים מן הסוג הראשון בחשודאח למתאמי התקנים מן הסוג השני

- א. מתאמי תחקקים מן הסוג הראשון מסובבים למימוש γ
- ב. מתאמי תחקקים מן הסוג הראשון גורמים למספר גדול של פעולות I/O (במקרה של תחקקי I/O) γ
- ג. מתאמי תחקקים מן הסוג הראשון גורמים לברזון CPU עקב busy waiting γ
- ד. כל התשובות חסרות וזכויות

שאלה 20

בסמך תחליטים **דושיים** ייווצרו מעקבות החרצה של התוכנית תבאה כאשר ניתן לחניט הצלחת כל סריאות המערכת!

```
main() {
    pid_1 = fork();
    pid_2 = fork();
    pid_3 = fork();
    pid_4 = fork();
}
```

$$2^4 - 1 = 15$$

- | | | |
|---|----|---|
| x | 2 | N |
| x | 7 | 2 |
| | 15 | ② |
| x | 17 | 7 |



בהצלחה!