

סוג הבדיקה:
א. בגרות לבתי ספר על-יסודיים
ב. בגרות לנבחנים חיצוניים
מועד הבדיקה: קיץ תשס"ד, 2004
מספר השאלה: 603, 899205

מדינת ישראל
משרד החינוך התרבות והספורט

מדי^י המחשב ב'

2 ייחדות לימוד (השלמה ל-5 י"ל)

הוראות לנבחן

- א. משך הבדיקה: שלוש שעות.
- ב. מבנה השאלה ופתח ההערכה: בשאלון זה שני פרקים.
פרק ראשון – בפרק זה ארבע שאלות,
ומהן יש לענות על שתיים.
- פרק שני – בפרק זה שאלות בחמשה מסלולים שונים.
עינה על שאלות בק במסלול שלמדת,
לפי הוראות בקבוצת השאלות במסלול זה.
- סה"כ – 100 נקודות
- ג. חומר עזר מותר בשימוש: כל חומר עזר (חוץ מחשב הנitin לתכנות).
- ד. הוראות מיוחדות:
- (1) את כל התכניות שאתה נדרש כתוב בשפת מחשב עילית בפרק הראשון
כתבו בשפה אחת בלבד – פסקל או C.
- (2) רשום על הרכבה החיצונית של המחברת את שם המסלול שלמדת
(אחד מחמשת המסלולים: מערכות מחשב ואסטובלר, תורת המחשב,
מודלים חישוביים, חישוב מקבילי ומבודר, תכונות מונחה עצמים).
- (3) רשום על הרכבה החיצונית של המחברת באיזו שפה אתה כותב – פסקל או C.

כתבו במחברת הבדיקה בלבד, בעמודים נפרדים, כל מה שברצונך לכתוב בטויטה (ראשי פרקים, חישוביים וכדומה).
רשום "טויטה" בראש כל עמוד טויטה. רישום טויטות כלשון על דפים שמוחוץ למחברת הבדיקה עלול לגרום לפסילת הבדיקה!

הנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות ולנבחנים כאחד.

ב ה צ ל ח ה !

/המשך מעבר לדף/

ה ש א ל ת

בשאלון זה שני פרקים: פרק ראשון ופרק שני.
עליך לענות על שאלות משני הפרקים, לפי ההוראות בכל פרק.

פרק ראשון (50 נקודות)

ענה על שתיים מהשאלות 1-4 (לכל שאלה — 25 נקודות).

.1. לפניך אלגוריתם:

סוד 1 (n , x)

{הפעולה מקבלת מספרים שלמים n ו- x , גדולים או שווים ל-0, ומחזירה מספר שלם}

$$(1) \text{ אם } x = 0 \text{ אז החזר } 0$$

$$(2) \text{ אחרת החזר } \text{סוד 1} (n , x - 1) + \text{סוד 2} (n , x - 1)$$

סוד 2 (n , x)

{הפעולה מקבלת מספרים שלמים n ו- x , גדולים או שווים ל-0, ומחזירה מספר שלם}

$$(1) \text{ אם } x = 0 \text{ אז החזר } 1$$

$$(2) \text{ אחרת }$$

$$(2.1) \text{ אם } x > n \text{ אז }$$

$$x \leftarrow n \quad (2.1.1)$$

$$(2.1.2) \text{ החזר } \text{סוד 1} (x , n) + \text{סוד 2} (x , n)$$

$$(2.2) \text{ אחרת החזר } \text{סוד 1} (x - 1 , n) + \text{סוד 2} (x - 1 , n)$$

א. מה יחזיר הziomon סוד 2 (10 , 3) ? רשם את המעקב אחר ביצוע האלגוריתם.

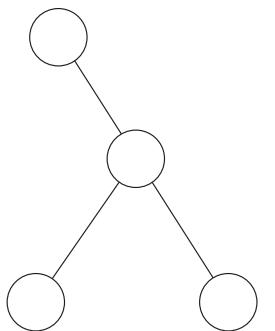
ב. מה יחזיר הziomon סוד 1 (10 , סוד 2 (3 , 10)) ?

ג. נתון $n < x_1$ וגם $n > x_2$.

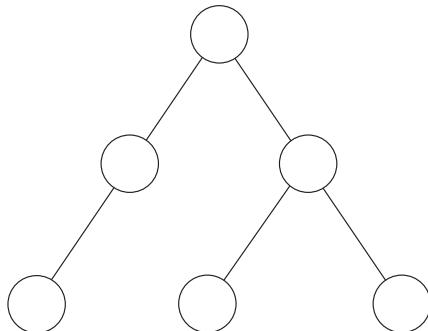
האם מתקיים : $\text{סוד 2} (n , x_1) = \text{סוד 2} (n , x_2)$? נמק.

2. עץ ביניי T הוא עץ "פְּרוֹ וּרְבָּוֹ", אם קיימים בו צומת אשר יש לו לפחות שני נכדים, וכל נכד מבן אחר. הערה: נכד הוא רק בן של בן.

דוגמה



לא עץ "פְּרוֹ וּרְבָּוֹ"



עץ "פְּרוֹ וּרְבָּוֹ"

כתב אלגוריתם **האם_פְּרוֹ_וּרְבָּוֹ?(T)**, המקבל עץ ביניי T
ומחזיר 'אמת' אם הוא עץ "פְּרוֹ וּרְבָּוֹ", ואחרת — 'שקר'.

.3

במשחק קופייה "הטל וצבור" יש 15 סיבובים. המשחק משתתפים 17 שחקנים, וכל אחד מהם ממוקף במספר שלם בין 1 ל-17. במהלך המשחק כל שחקן מטיל קופייה. בהטלה הקופייה מתקבל מספר בין 1 ל-6.

- בכל אחד מ-15 הסיבובים כל שחקן מבצע בתורו את ארבע הפעולות האלה:
- (i) בוחר מספר שלם, ומכוון מהו הערך שלו.
 - (ii) מטיל את הקופייה 5 פעמים, ומסכם את המספרים שהתקבלו ב-5 ההטלות.
 - (iii) מחשב את הערך המוחלט של ההפרש בין המספר שהכוון ב-(i) ובין הסכום שהיחסב ב-(ii).
 - (iv) מעדכן את מספר הנקודות הכלול שלו בדרך זו:
אם ההפרש שהיחסב ב-(iii) הוא 0, מספר הנקודות הכלול שלו מתאפס. אחרת — הוא מוסיף את הערך המוחלט של ההפרש למספר הנקודות הכלול שלו.

השחקן שצבר הכי פחות נקודות ב-15 סיבובים הוא המנצח.

אם בסיום המשחק יש ליותר שחקן אחד את המספר הכי נמוך של נקודות, אז מתבצע סיבוב נוספת. בל השחקנים משתתפים בסיבוב הנוסף, ומספר הנקודות של כל אחד מהם הוא מספר הנקודות שצבר מתחילת המשחק. כל עוד אין מנצח, יתקיים סיבוב נוסף.
לפניך פועלות מתוך משך הטיפוס **משחק** בסביבת העבודה (כתובות בפסקול וב-C).
הנח שקייםת הדרה של טיפוס הנתונים `game_type`.

משחק בפסקול

הפעולה מחזירה משחק מאותחל. מספר הנקודות הכלול של כל משתתף מאותחל ל-0.	procedure game_init (var g: game_type);
הפעולה מקבלת משחק <code>g</code> ונתוני שחקן עבור סיבוב אחד: ב- <code>num</code> — את מספר השחקן, ב- <code>announce</code> — את הערך שעליו הכריז השחקן, ב- <code>sum</code> — את סכום 5 ההטלות של השחקן בסיבוב זה. הפעולה מעדכנת את הnikod הכלול של השחקן במשחק. <u>הנחות:</u> <code>g</code> מאותחל, <code>num</code> — שלם בין 1 ל-17, . <code>sum</code> ≥ 5 , <code>announce</code> ≥ 5	procedure update_player_score (var g: game_type; num, announce, sum: integer);
הפעולה מקבלת משחק <code>g</code> . אם המשחק הסתיים, הפעולה מחזירה את המספר של השחקן המנצח, אחרת — הפעולה מחזירה 0.	function winner (g: game_type): integer;
הפעולה מחזירה את סכום 5 ההטלות של הקופייה.	function throw_5_times: integer;

ממשק ב-C

<p>הפעולה מחזירה משחק מאותחל. מספר הנקודות הכלול של כל משתמש מאותחל ל-0.</p>	game_type game_init ()
<p>הפעולה מקבלת משחק g ונתוני שחקן עבור סיבוב אחד: num — את מספר השחקן, $announce$ — את הערך שעליו הכריז sum — את סכום 5 ההטLOTות של המשחק בסיבוב זה. הפעולה מעדכנת את הנקוד הכלול של השחקן במהלך המשחק. <u>נקודות:</u> g מאותחל, num — שלם בין 1 ל-17, $.sum \geq 5$, $announce \geq 5$</p>	void update_player_score (game_type *g, int num, int announce, int sum) השחקן,
<p>הפעולה מקבלת משחק g. אם המשחק הסתיים, הפעולהמחזירה את המספר של השחקן המנצח, אחרת — הפעולהמחזירה 0.</p>	int winner (game type g)
<p>הפעולהמחזירה את סכום 5 ההטLOTות של הקובייה.</p>	int throw_5_times ()

- א.** יציג את טיפוס הנתונים `game_type` בסביבת העבודה.
ב. ממש בסביבת העבודה פעולה מקבלת משחק מאותחל, שבו מספר הנקודות ההתחלתי של כל משתמש הוא 0. הפעולה מರיצה משחק, ומחזירה את המספר של השחקן המנצח.

4. נגידר "בלוק" במחסנית כראף של לפחות שני איברים זהים.

א. נתונה מכחנית M , שאיבריה הם מספרים שלמים.

כטוב תת-תכנית בסביבת העבודה שמקבלת את המחסנית M , ומחזירה מחסנית חדשה. עבור כל "블록" שבמחסנית M יש במחסנית החדשה איבר, שערכו הוא המכפלה של האיבר שב"блок" במספר הפעמים שהוא מופיע בו".
אנו חשים רשות לסדר האיברים במחסנית המוחזרת

הארה: אין צורך למש בנסיבות העבודה את הפעולות של משק המחסנית.

דוגמאות

המחסנית המוחזרת	מחסנית M
-8	5
0	5
15	5
	0
	0
	0
	0
	3
	2
	-4
	-4

ב. מהי סיבוכיות זמן הריצה של התת-תכנית שכתבת בסעיף א? הסביר.
הנח שסיבוכיות זמן הריצה של כל אחת מן הפעולות של משקל המחשבנית
היא $O(1)$.

פרק שני (50 נקודות)

בפרק זה שאלות בחמישה מסלולים שונים. ענה רק על שאלות במסלול למדת.

מערכות מחשב ואסטנבלר

אם למדת מסלול זה, ענה על שתיים מהשאלות 5-8 (לכל שאלה – 25 נקודות).

5. יש לבנות מערכת חיבורים למחשב קיים, כדי חירום לשימרת מספרי הטלפון של משטרת ישראל ושל מגן דוד אדום.
במערכת יש רכיב זיכרון מסווג ROM עם שמונה כתובות של 8 סיביות כל אחת, וכן שני לחצני חירום.
לחיצה על הלחצן הראשון תגרום לתוכנת המחשב להתקשר למשטרת ישראל,
ולחיצה על הלחצן השני לתוכנת המחשב להתקשר למגן דוד אדום.

א. תכנן את מערכת החיבורים (ראה תרשימים):

. (i) לצורך חיבור זיכרון ה-ROM ל쿄י המחשב, החל מכתובת 0A520h.

. (ii) לצורך חיבור שני הלחצנים ל쿄י המחשב, החל מכתובת 0F6h.

ב. סרטט את טבלת המיפוי של כתובות הקלט/פלט.

ג. סרטט את מערכת החיבורים למעבד לצורך חיבור רכיבי הזיכרון לכתובות המתקבלות ממפת הזיכרון.



.6

בתחילת תכנית במשפט אסמלר הוגדרו הנתונים בצורה זו:

ARR						FIRST		משתנה
00	01	02	03	04	05	06	07	כתובות
04	10	03	07	5B	10	0	0	נתון

שים לב: הנתונים והכתובות הם הקסדצימליים.

לפניך קטע התוכנית:

	LEA	SI , ARR
	MOV	CX , 5
	MOV	AH , [SI]
	MOV	AL , AH
AGA1 :	INC	SI
	CMP	AH , [SI]
	JLE	CONT1
	MOV	AH , [SI]
CONT1 :	CMP	AL , [SI]
	JNB	CONT2
	MOV	AL , [SI]
CONT2 :	LOOP	AGA1
	LEA	DI , FIRST
	MOV	[DI] , AH
	MOV	[DI + 1] , AL
SOF :	NOP	

.א. עקב בעזרת טבלת מעקב אחר הביצוע של קטע התוכנית, ורשות מה יהיה התוכן של תא הזיכרון שכתובותיהם מ- 00 עד 07 .

.ב. מה מבצע קטע התוכנית? בתשובה תתייחס לתוכן האוגרים AL ו- AH .

.ג. החלף את ההוראה LOOP AGA1

בהוראה אחרת / הוראות אחרות, כך שקטע התוכנית יבצע את הנדרש
בלי להשתמש בהוראה LOOP .

.7 סדרת פיבונצ'י היא סדרת מספרים, אשר החל מהאיבר השלישי כל איבר הוא סכום של שני האיברים ש לפניו. שני האיברים הראשונים בסדרה הם 0 ו-1:

... 13 8 5 3 2 1 0

לפניך קטע של תכנית בשפת אסמבלי, המחשב את סדרת פיבונצ'י עד למספר בסדרה שערכו 5 (כולל), ושומר את המספרים שהתקבלו במערך זיכרון החל מכתובת BUFFER. מקטע התכנית הושמו פקודות ואופרנדים.

START:	MOV	SI , 2
	LEA	<u> </u> , [_____]
	MOV	AL , 0
	MOV	AH , 1
	MOV	[BX +SI -2] , AL
	MOV	[BX +SI -1] , AH
	INC	AL
	INC	AH
	MOV	[BX +SI] , AL
	ADD	SI , 1
	MOV	DH , AH
	ADD	AH , AL
	MOV	AL , DH
	MOV	[BX +SI] , <u> </u>
	<u> </u>	<u> </u> , 1
	MOV	DH , AH
	ADD	<u> </u> , <u> </u>
	MOV	AL , DH
	MOV	[BX +SI] , AL
	ADD	SI , 1
	MOV	[BX +SI] , AH
SOF2:	NOP	

העתק למחברתך את השורות שבהן הושמו פקודות או אופרנדים, והשלם את החלקים החסרים כך שקטע התכנית יבצע את הנדרש.

8. לפניך שישה היגדים, (1)-(6).

קבע לכל אחד מהם אם הוא נכון או לא נכון, ונמק את תשובתך.

- (1) סדרת הוראות של קלט/פלט, הנדרשת במקרה של גישה ישירה לקבצים, זהה לסדרת הוראות של קלט/פלט, הנדרשת במקרה של גישה סדרנית.
- (2) במחשב הפועל עם שני מעבדים ניתן לשטר את זיכרונו ה- RAM לשנייהם.
- (3) במערכת שיתוף זמנים רק תהליך אחד נמצא בזיכרון הראשי בזמן מסוים.
- (4) במערכת רבת-מעבדים תהליך אינו יכול לעבוד עם יותר מעבד אחד בו-זמנית.
- (5) במערכת רבת-תהליכיים התהליך, שיש לו העדיפות הנמוכה ביותר מבין כל התהליכיים הפעילים, לא יקבל הקצת מעבד (מצב RUN), כל עוד לא הסתיימו התהליכיים שיש להם עדיפות גבוהה יותר.
- (6) במערכת עם זיכרון וירטואלי ניתן להוציא תהליך, שאינו ביצוע, מחוץ לזכרו ממשי.

תורת המחשב

אם למדת מסלול זה, ענה על שתיים מהשאלות 9-12 (לכל שאלה – 25 נקודות).

.9. לפניך האלגוריתם **תעלומה**:

תעלומה (E, Y, N, X_0)

{ האלגוריתם מקבל מספר שלם N ושלושה מספרים ממשיים

$$Y = 0 \text{ אם } (1)$$

$$\text{או } 0 \text{ אז החזר } 0 \quad (1.1)$$

$$\text{אחרת } \quad (1.2)$$

$$F \leftarrow 0 \quad (1.2.1)$$

$$\text{כל עוד } 0 = F \text{ בצע } \quad (1.2.2)$$

$$X_2 \leftarrow X_0^{(N-1)} \quad (1.2.2.1)$$

$$X_1 \leftarrow ((N-1) * X_0 + Y / X_2) / N \quad (1.2.2.2)$$

$$F \leftarrow 1 \text{ או } E > |X_0 - X_1| \text{ אם } \quad (1.2.2.3)$$

$$X_0 \leftarrow X_1 \quad (1.2.2.4)$$

$$\text{החזר } X_0 \quad (1.2.3)$$

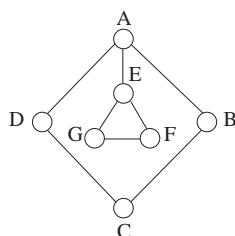
.א. עקב בעורת טבלת מעקב אחר ביצוע האלגוריתם **תעלומה** (25, 2, 49, 0.01).

ורשום מה יוחזר.

.ב. מה מבצע האלגוריתם **תעלומה** (E, Y, N, X_0) ?

10. יהיה גרף G , קשיר, לא מכוון, בלי קשרות מרובות (אין יותר מקשת אחת בין שני קדקודים) ובלי לולאות עצמאיות.
- נגידר "מסלול מעגלי" בגרף G כמסלול המתחיל ומסתיים באותו קדקוד, וכל קדקוד אחר במסלול מופיע בו רק פעם אחת.
- יהיה "כיסוי מעגלים" של גרף G אוסף של "מסלולים מעגליים" ב- G , כך שכל קדקוד בגרף G נמצא במסלול מעגלי אחד בדיווק.

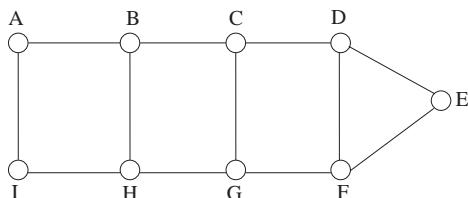
דוגמה



נתון הגרף $G1$:

"כיסוי מעגלים" של הגרף $G1$ הוא אוסף המסלולים הממעגליים: {EFGE, ABCDA}

א. לפניו הגרף $G2$:



(i) רשות "כיסוי מעגלים" לgraf $G2$.

(ii) האם הכיסוי שרשמת הוא "כיסוי מעגלים" ייחיד? אם כן – נמק מדוע, אם לא – רשות "כיסוי מעגלים" נוספים.

ב. תן דוגמה לגרף קשיר, לא מכוון, בלי קשרות מרובות ובלי לולאות עצמאיות,

שבו הדרגה של כל אחד מהקדקודים היא לפחות 2 ולאו "כיסוי מעגלים".

ג. יהיה גרף G , קשיר, לא מכוון, בלי קשרות מרובות ובלי לולאות עצמאיות.

הגרף G' הוא גרף המכיל את כל הקדקודים של הגרף G ורק את הקשרות המופיעות ב"כיסוי מעגלים" שלו.

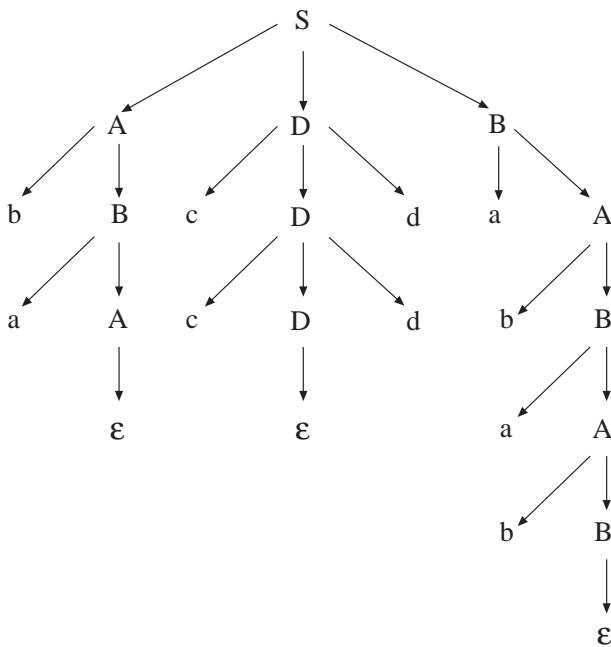
נסח תנאי על הגרף G שיבטich כי יש עבورو גראף G' קשיר.

11. לפניך השפה L מעל הא"ב $\{a, b, c\}$:

$$L = \left\{ \begin{array}{l} \text{אוסף כל המילים המתחילה ב- } b \text{ ומסתיימת ב- } b, \\ \text{יש בהן לפחות } c \text{ אחד, ומספר ה- } a \text{ בהן זוגי} \end{array} \right\}$$

בנה אוטומט סופי דטרמיניסטי, שיקבל את השפה L .

12. לפניך עז גזירה המשתמש בכל כללי הגזירה של דקדוק G :



א. רשם את הדקדוק G .

ב. קבע כל אחת מהמילים (i)-(iv) שלפניך – אם היא ניתנת לגזירה על ידי

הדקודק G שרשmot בסעיף א.

אם כן – רשום עז גזירה, אם לא – נמק מדייע.

babcda (i)

cccdada (ii)

abbccdddbba (iii)

baab (iv)

ג. תאר במילים את השפה הנוצרת על ידי הדקדוק G .

מודלים חישוביים

אם למדת מסלול זה, ענה על שתיים מהשאלות 13-16 (לכל שאלה – 25 נקודות).

13. לפניך השפה L מעלה הא"ב $\{a, b, c\}$:

$$L = \left\{ \begin{array}{l} \text{אוסף כל המילים המתחילה ב- } b \text{ ומסתיימות ב- } b, \\ \text{יש בהן לפחות } c \text{ אחד, ומספר ה- } a \text{ בהן זוגי} \end{array} \right\}$$

בנה אוטומט סופי דטרמיניסטי, שיקבל את השפה L .

14. נגידר פעולה חדשה על שלוש שפות L_C, L_B, L_A באופן הזה:

$L_B \# L_A \# L_C$ היא קבוצת כל המילים הנמצאות ב- L_A , אך לא נמצאות ב- L_B וגם לא נמצאות ב- L_C .

a. לפניך שלוש השפות L_1, L_2, L_3 מעלה הא"ב $\{a, b\}$:

$$L_1 = \{a^n b^m a^n \mid n, m > 0\}$$

$L_2 = \{ \text{מספר האותיות } a \text{ שונה ממספר האותיות } b \}$

$L_3 = \{ \text{מספר האותיות } b \text{ הוא אי-זוגי} \}$

. $L_1 \# (L_2, L_3)$ השיכת לשפה (i)

הסביר מדוע המילה שייכת לשפה זו.

. $L_1 \# (L_2, L_3)$ השיכת לשפה (ii)

הסביר מדוע המילה אינה שייכת לשפה זו.

. $\{a, b\} \# (L_2, L_3)$ מעלה הא"ב (iii)

b. נתון כי L_A, L_B, L_C הן שפות רגולריות.

הוכיח שהשפה $L_A \# (L_B, L_C)$ היא רגולרית.

$$L_1 = \{ a^n b^n \mid n > 0 \}$$

לפניך שלוש השפות $L_3 - L_1$:

$$L_2 = \{ a^n b^n d a^k b^k \mid n, k \geq 0 \}$$

$$L_3 = \{ ba \mid \text{המקילות את הרצף } \{a, b\} \}$$

לפניך חמש הטענות (i)-(v):

$$\text{aaaabbbb} \in L_1^2 \quad (\text{i})$$

$$R(L_1) \cdot d \cdot R(L_1) \subset R(L_2) \quad (\text{ii})$$

$$L_1 \cap L_2 = (L_2 \cup L_3) \cap \{d\} \quad (\text{iii})$$

$$L_1 = (R(L_3) \cap L_1) \quad (\text{iv})$$

$$\bar{L}_3 \subset L_1 \quad (\text{v})$$

קבע כל אחת מהן אם היא נכונה או לא, ונמק את קביעותך.

16. בנה מכונת טיורינג שעל סרט הזיכרון שלה תהיה מילת קלט לא ריקה מעל

הא"ב $\{a, b\}$.

המכונה תכתוב את האות האחורייה במליה במקום האות הראשונית. כל האותיות, חוץ מהאות הראשונית, לא ישתנו.

דוגמה

סרט הזיכרון של המכונה לפני תחילת פעולתה:

↔	a	a	b	a	b	Δ	Δ
---	---	---	---	---	---	---	---	-------

סרט הזיכרון של המכונה בסוף פעולתה:

↔	b	a	b	a	b	Δ	Δ
---	---	---	---	---	---	---	---	-------

חישוב מקבילי ומבוזר

אם למדת מסלול זה, ענה על שתיים מהשאלות 17-20 (לכל שאלה – 25 נקודות).

17. לפניך תכנית, ובה שני תהליכיים P1 ו- P2 המבצעים במקביל. התכנית משתמשת במשתנה גלובלי X מטיפוס שלם, המאותחל ל-1.

```
process P1;
begin
  while X < 0 do
    begin
      X := X + 3;
    end;
    write(X);
  end;
```

```
process P2;
begin
  while X > 0 do
    begin
      X := X - 3;
    end;
    write(X);
  end;
```

הערה: בכל אחד משני הטעיפים א-ב אתה נדרש לתאר תסריט. عليك לרשום את התהליכי המבצע בכל שלב, את ההוראה המבצעת, את ערכי המשתנים, ואת הפלט (אם יש פלט).

- א. תאר תסריט שבו התכנית אינה מסתיימת.
ב. תאר תסריט סופי שבו מתקבל הפלט: -2, -2.

.18 לפניך אלגוריתם, ובו שלושה תהליכי א-ג המתבצעים במקביל.

האלגוריתם קולט מספרים שלמים בתחום 1-10 (כולל). בסוף רישימת הקלט יופיע המספר 0 פעמיים. האלגוריתם ידפיס 'כן', אם בראשימת הקלט הופיעו כל המספרים השלמים שבין 1 ל- 10 (כולל), אחרת – האלגוריתם ידפיס 'לא'.
האלגוריתם משתמש במשתנים глובליים: N, SUM, P, ובערך A בגודל 10 שכל איבריו מאותחלים ל- 0. כל המשתנים הם מטיפוס שלם.

תהליך ג	תהליך ב	תהליך א
קרא N כל עוד 0 < N בעצ $A[N] \leftarrow 1$ קרא N	קרא N כל עוד 0 < N בעצ $A[N] \leftarrow 1$ קרא N	$SUM \leftarrow 0$ עבור P מ- 1 עד 10 בעצ $SUM \leftarrow SUM + A[P]$ אם $SUM = 10$ אז הדפס 'כן' אחרת הדפס 'לא'

האלגוריתם לא תמיד מושג את מטרתו.

א. נסח, באופן מילולי ועל ידי תרשימים תלויות, את דרישות התיאום שיש להוסיף לאלגוריתם כדי שתמיד ישג את מטרתו.

ב. העתק למחברתך את התהליכי א-ג, והוסף להם מנגןון תיואום באמצעות סמפורים, שיבטיח כי האלגוריתם תמיד ישג את מטרתו.

לפניך תכנית, ובה ארבעה תהליכיים P1-P4 המתבצעים במקביל. התכנית משתמשת בלוח ופטקים. כל הפטקים על הלוח הם במבנה של-tag זיהוי ומספר שלם. על הלוח יש כמוות פטקים עם tag' זיהוי 'A', 'B', 'C', ושולשה פטקים עם tag' 'N'. על הפטקים עם tag' זיהוי 'N' רשומים המספרים האלה: על פטק אחד רשום מספר הפטקים מסווג 'A', על פטק שני רשום מספר הפטקים מסווג 'B', ועל פטק שלישי רשום מספר הפטקים מסווג 'C'. הצמדת פטקים אלה אל הלוח נעשית באמצעות התהיליך init_board. מטרת התכנית היא להציג את סכום כל המספרים הרשומים על הפטקים עם tag' זיהוי 'A', 'B', 'C'.

program P;	
process P1; var N, S, I, D : integer; begin S := 0; removenote ('N', N); for I := 1 to N do begin removenote ('A', D); S := S + D; end; postnote ('S', S); end;	process P2; var N, S, I, D : integer; begin S := 0; removenote ('N', N); for I := 1 to N do begin removenote ('B', D); S := S + D; end; postnote ('S', S); end;
process P3; var N, S, I, D : integer; begin S := 0; removenote ('N', N); for I := 1 to N do begin removenote ('C', D); S := S + D; end; postnote ('S', S); end;	process P4; var S, D, I : integer; begin S := 0; for I := 1 to 3 do begin removenote ('S', D); S := S + D; end; writeln (S); end;
begin init_board; parbegin P1, P2, P3, P4; parend; end.	

- א. התכנית אינה מSIGGA את מטרתה. הסבר מדיע.
ב. רוני ויעל הציעו שתי דרכי לשנות את התכנית כדי שהיא תשיג את מטרתה.

הדרך של רוני:

נשנה את מבנה הפטקים עם תג הזיהוי 'A' , 'B' , 'C' , כך שכולם יהיו עם תג זיהוי אחד 'G' .

בנוספ, בתהליכיים P1 , P2 , P3 בכל ההוראות מסווג `removenote` יוחלפו הtags 'A' , 'B' , 'C' בתג 'G' .

הדרך של יעל:

במקומות שלושת הפטקים עם תג הזיהוי 'N' , יהיו שלושה פטקים שונים:
פטק אחד עם תג זיהוי 'X' שיכיל את מספר הפטקים עם תג זיהוי 'A' ,
פטק שני עם תג זיהוי 'Y' שיכיל את מספר הפטקים עם תג זיהוי 'B' ,
פטק שלישי עם תג זיהוי 'Z' שיכיל את מספר הפטקים עם תג זיהוי 'C' .

בנוספ, נעשה את השינויים האלה:

בתהליך P1 –

במקומות הוראה `removenote ('X', N)` `removenote ('N', N)` בתהליך P2 –

במקומות הוראה `removenote ('Y', N)` `removenote ('N', N)` בתהליך P3 –

במקומות הוראה `removenote ('Z', N)` `removenote ('N', N)`

כל אחת מהדריכים שהוצעו, קבוע אם היא מביאה לכך שהתוכנית תשיג את מטרתה, ונמק את קביעתך.

20. קבוצה של N גנרים ערוכה סביב מבצר. הגנרים צריכים להחליט אם לתקוף את המבצר או לסתות. כל גנرל מודיע פומבי לכל שאר הגנרים את דעתו בנוגע להחלטה אם לתקוף או לסתות. כל הودעה לכל גנרל נשלחת באמצעות שליח אחר, וכל שליח מגיע ליעדו. כל גנרל מקבל את כל הודעות שנשלחו אליו, מוסיף להן את דעתו, וסופר את מספר הdecisions بعد תקיפה ואת מספר decisions בעוד נסיגה. הוא קובע את החלטתו הסופית על פי הרוב. במקרה של תיקו ההחלטה היא לסתות. יתכן שהיו שליחים "מבולבלים" שייעברו הודעות שקריות.

לפניך האלגוריתם שכל גנרל מבצע:

בחר_דעה $\leftarrow X$

שלח לכל שאר הגנרים (X)

שלח לכל שאר הגנרים (X)

$A \leftarrow 0$

$R \leftarrow 0$

אם ("התקפה" = X) אז $A \leftarrow 1$

אחרת $R \leftarrow 1$

עבור I מ-1 עד $(N-1) * 2$ בצע

קבל M

אם ("התקפה" = M) אז $A \leftarrow A + 1$

אחרת $R \leftarrow R + 1$

אם $R > A$ אז החלט לתקוף

אחרת החלט לסתות

א. הוכח כי לקבוצה של 4 גנרים, גם בלי הודעות שקריות, יתכן מצב שבו שני גנרים יקבעו החלטות שונות.

ב. שנה את האלגוריתם כך שכל הגנרים יקבעו החלטות זהות כאשר אין הודעות שקריות.

ג. התבסס על האלגוריתם אחרי השינוי שערכת בסעיף ב, והוכח או הפרך את הטענה: בקבוצה של 4 גנרים מספיק שליח אחד בלבד לעבר הודעה שקרית, כדי שני גנרים יקבעו החלטות שונות.
/המשך בעמוד 21/

שים לב:

השאלות ב串联ול

תכנות מונחה עצמים

בעמוד הבא.

תכנות מונחה עצמים

אם למדת מסלול זה, ענה על שתיים מהשאלות 21-24 (לכל שאלה – 25 נקודות).

21. עיין בתכנית ש לפניך, וענה על הסעיפים א-ד שאחריה (בעמוד הבא).

```
#include <iostream.h>
class A{
public:
    void f ()
    {
        cout << "inside A" << endl;
    }
    void m ()
    {
        cout << "hello" << endl;
    }
private:
    int num;
};

class B : public A{
public:
    void f ()
    {
        cout << "inside B" << endl;
    }
private:
    int num;
};

class C{
public:
    void func (A a)
    {
        a.f();
    }
private:
    int num;
    B b;
};

void main ()
{
    B b;
    C c;
    c.func(b);
}
```

/המשך בעמוד 23/

(**שים לב: סעיפי השאלה בעמוד הבא.**)

- א.** (i) מהו היחס בין המחלקות A ו- B ? הסבר.
(ii) מהו היחס בין המחלקות B ו- C ? הסבר.
- ב.** (i) האם אפשר לשנות את הערך של num שבמחלקה A מתוך המחלקה B ?
אם כן – כתוב כיצד, אם לא – נמק מודיע.
- האם אפשר לזמן את הפונקציה () f שבמחלקה A מתוך המחלקה B ?
אם כן – כתוב כיצד, אם לא – נמק מודיע.
- ג.** (i) האם אפשר לזמן את הפונקציה () f שבמחלקה B מתוך המחלקה C ?
אם כן – כתוב כיצד, אם לא – נמק מודיע.
- האם אפשר לשנות את הערך של num שבעצם b מתוך המחלקה C ?
אם כן – כתוב כיצד, אם לא – נמק מודיע.
- האם אפשר לזמן את הפונקציה () f שבמחלקה A מתוך המחלקה C ?
אם כן – כתוב כיצד, אם לא – נמק מודיע.
- ד.** (i) ניתן לזמן את הפונקציה () m מפונקציית ה- main בנסיבות ההוראה:
b.m();
הסביר מודיע.
- האם זימנו הפונקציה func מתוך ה- main הוא תקין? אם כן – נמק מודיע,
וכתוב מה יודפס. אם לא – נמק מודיע.

.22. עקוב בעזרת טבלת מעקב אחר התכנית שלפניך, ורשות מה יהיה פלט התכנית.

```
#include <iostream.h>

class A {
public:
    A (int a , int b) { x = a;
                        y = b;
                        cout << "A constructor :"
                        << '(' << x << ',' << y << ')' << endl; }

    ~ A( ) { cout << "A destructor :" << '(' << x << ',' << y << ')' << endl; }

protected:
    int x,y;

};

class B : public A {
public:
    B (double k , int a , int b) : A (a , b)
    { m=k;
      cout << "B constuctor :"
      << '(' << m << ',' << x << ',' << y << ')' << endl; }

    B (double k , int a) : A (a , a)
    { m=k;
      cout << "B constuctor :"
      << '(' << m << ',' << x << ',' << y << ')' << endl; }

    ~ B( ) { cout << "B destructor :" << '(' << x << ',' << y << ')' << endl; }

private:
    double m;

};

void main ( )
{
    A s(24 , 6);
    B q(4.5 , 13 , 48);
    B r(3.14 , 20);

}
```

.23. במשרד התקשורת מעוניינים לפתח מערכת תוכנה. המערכת תרכז מידע על מפעילי רשתות טלפון ניידים, ועל המשתמשים בטלפונים ניידים.

יש ארבעה מפעילי רשתות טלפון ניידים, ולכל אחד מהם יש לא יותר מ-500,000 משתמשים בטלפונים ניידים.

כל מפעיל רשת שומר לכל אחד מהמשתמשים בטלפונים ניידים ברשות שלו את הפרטים האליה:

שם המשתמש, מספר תעודה זהה שלו, כתובתו, מספר הטלפון של המכשיר הנייד, סוג המכשיר.

במערכת התוכנה רוצים להציג פועלה ClientsPerDevice שתקבל סוג מכשיר, ותדפיס פרטים של כל המשתמשים בסוג המכשיר זהה אצל ארבעת מפעילי הרשתות. הפרטים שיודפסו עבורו כל משתמש הם: מספר הטלפון של המכשיר הנייד, שם המשתמש, כתובתו, שם מפעיל הרשת.

א. הגדר בלי שימוש ב- C++ את המחלקות הדרשיות לפיתוח מערכת התוכנה.

לכל מחלוקת ציין תכונות וرك את הפונקציות החברות הדרשיות למימוש הפעולה ClientsPerDevice. לכל פונקציה חברה כתוב כותרת ותיעוד.

ב. הגדר וממש ב- C++ את הפעולה ClientsPerDevice. אתה יכול להיעזר בפעולות שהגדרת בסעיף א בלי לממש אותן.

.24. חברת העוסקת בפיתוח משחקים מחשב מעוניינת למחשב את משחק הקלפים "קלף לי", שבו יכולים לשתף 2-8 שחקנים. במשחק משתתפים "שחקנים אנושיים" (בני אדם) ו"שחקנים ממוחשבים". בכל משחק משתתפים לפחות "שחקן אנושי" אחד ולפחות "שחקן ממוחשב" אחד.

במשחק יש 120 קלפים, וכל קלף יש שלושה מאפיינים:

- ערך – מספר שלם בתחום 1-12;
- צורה מבין שלוש הצורות – עיגול, ריבוע, משולש;
- צבע מבין ארבעת הצבעים – אדום, צהוב, כחול, ירוק.

במשחק יש שתי ערכות של קלפים: ערמה אחת היא קופפה מסווגת וסדר הקלפים בה אקראי, וערמה שנייה שמתחלילה בקלט גלי אחד ונערכות לפי כללי המשחק. בתחילת המשחק כל שחקן מקבל 5 קלפים מהקופה. קלף נוסף מהקופה נחיש ומונח בנפרד גלי לעיני כל השחקנים, ועליו תיערים הערמה השנייה. כל שחקן רשאי להניח בתورو קלף אחד על גבי הקלו החשוף, על פי ארבעת הכללים

האלה:

- * שחקן יכול להניח קלף, אם הוא מאותה צורה או מאותו צבע של הקלף הנמצא בראש הערמה, או קלף שהערך שלו גבוה מן הערך של הקלף שבראש הערמה.
- * אם אין לשחקן קלף שהוא יכול להניח בראש הערמה, עליו למשוך קלף אחד מן הקופה, והתוור עובר לשחקן הבא.
- * המנצח במשחק הוא השחקן הראשון שנוטר בלי קלפים.
- * אם הקופה התroxנה מקלפים לפני שיש מנצח, ואין שחקן שיוכל להניח קלף נוסף, המשחק יסתתיים בלי הכרעה.

הקלפים של "שחקן ממוחשב" ממוקמים תמיד לפי ערך הקלף בסדר עולה.

(שים לב: המשך השאלה בעמוד הבא).

לפניך חלק מההגדירות בלי שימוש של שלוש המחלקות:

קלף (Card), ערמת קלפים (Collection), שחקן (Player).

```
class Card
{
public:
    int getValue(); // אחזור ערך של קלף
    int getShape(); // אחזור צורה של קלף
    int getColor(); // אחזור צבע של קלף
private:
    int value; // ערך של קלף
    int shape; // צורה של קלף
    int color; // צבע של קלף
};

class Collection
{
public:
    void addCard(Card x); // הוספת קלף לערימה
    Card removeCard(); // הוצאת קלף מעירמה והחזרתו
    int getNumOfCards(); // החזרת מספר הקלפים בערימה

private:
    int numOfCards; // מספר קלפים בערימה
    Card cards[120]; // הקARDS בערימה
};

class Player
{
public:
    void moveCard(Collection x); // הוצאת קלף מיידי השחקן והנחתו בראש ערימה
private:
    int playerNum; // מספר שחקן
    Collection hand; // הקלפים שבידי השחקן
};
```

א. הגדר בלי שימוש ב- C++ את המחלקות "שחקן אנושי" ו"שחקן ממוחשב".

ל"שחקן הממוחשב" בשונה מה"שחקן האנושי" יש פעולה מיוחדת של מיוון הקלפים.

ב. ממש במחלקה Player פעולה שתמשוך קלף מן הקופה ותעביר אותו לשחקן.

ג. הגדר ב- C++ את המשתנים הדרושים לניהול המשחק "קלף ליא".

ב הצלחה !

זכות היוצרים שמורה למדינת ישראל
אין להעתיק או לפרסם אלא ברשות משרד החינוך התרבות והספורט