

סוג הבדיקה: א. בגרות לבתי ספר על-יסודיים
ב. בגרות לנבחנים חיצוניים
מועד הבדיקה: קיץ תשס"ה, 2005
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. – (25×2) – 50 נקודות
ענוה על שאלות בק במסלול שלמדת,
לפי הוראות בקבוצת השאלות במסלול זה.

סה"כ – 100 נקודות

ג. חומר עזר מותר בשימוש: כל חומר עוז, חוץ ממחשב הניתן לתוכנות.

ד. הוראות מיוחדות:

- (1) את כל התכניות שאתה נדרש לכתוב בשפת מחשב עילית בפרק הראשון
כתב בשפה אחת בלבד – פסקל או C.
(2) רשות על הבדיקה החיצונית של המחברת באיזו שפה אתה כותב – פסקל או C.
(3) רשות על הבדיקה החיצונית של המחברת את שם המסלול שלמדת,
אחד מחמשת המסלולים: מערכות מחשב ואסמבילר, תורת המחשב,
מודלים חישוביים, חישוב מקבילי ומבוזר, תכונות מונחה עצמים.

כתב במחברת הבדיקה בלבד בעמודים נפרדים, כל מה שברצונך לכתוב בטיויטה (ראשי פרקים, חישוביים וכדומה).
רשות "טיויטה" בראש כל עמוד טיוטה. רישום טוויות כלשון על דפים שמחוץ למחברת הבדיקה עלול לפגיעת הבדיקה!

הנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות לנבחנים כאחד.

ב ה צ ל ח ה !

/המשך מעבר לדף/

ה ש א ל ו ת

בשאלוֹן 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 קטע תכנית בפסקל וב-C:

פסקל

```

const N = 8;
type typeArray = array [1.. N] of integer;

function firstCheck (x , num : integer) : boolean;
{           { הפקציה מקבלת ספרה x ומספר שלם num גדול מאפס.
{           { הפקציה מחזירה...
begin
  if num = 0
    then firstCheck := false
  else if (num mod 10 = x)
    then firstCheck := true
    else firstCheck := firstCheck (x , num div 10);
end;

function secondCheck (a : typeArray ; k : integer) : integer;
{           { הפקציה מקבלת מערך a בגודל N, המכיל מספרים שלמים גדולים מאפס
{           { ומספר שלם k בתחום 1 עד N (כולל).
{           { הפקציה מחזירה...
begin
  if (k = N)
    then secondCheck := 0
  else if (firstCheck (a[k] mod 10 , a[k+1]))
    then secondCheck := 1+secondCheck (a , k+1)
    else secondCheck := secondCheck (a , k+1);
end;
/המשך בעמוד 3/

```

C

```

typedef N = 8;
int firstCheck (int x , int num)
//                                     הפונקציה מקבלת ספרה x ומספר שלם num גדול מאפס.
//                                     הפונקציה מחזירה ...
{
    if (num == 0)
        return 0;
    else
        if (num % 10 == x)
            return 1;
        else
            return firstCheck(x , num /10);
};

int secondCheck (int a[] , int k)
//                                     הפונקציה מקבלת מערך a בגודל N, המכיל מספרים שלמים גדולים מאפס
//                                     ומספר שלם k בתחום 0 עד N-1 (כולל).
//                                     הפונקציה מחזירה ...
{
    if (k == (N-1))
        return 0;
    else
        if (firstCheck (a[k] % 10 , a[k+1]) == 1)
            return 1+secondCheck (a , k +1);
        else
            return secondCheck (a , k +1);
}

```

});
 מה מבצעת הפונקציה `firstCheck` ?
 נתנו מערך `char`-`array` `a` בגודל 8.
 א. `a[0] = 'a'`
 ב. `a[0] = 'b'`
 ג. `a[0] = 'c'`
 ד. `a[0] = 'd'`

$$\rightarrow \text{תחילת המערך} \quad \begin{array}{cccccccc} 400 & 200 & 100 & 25 & 351 & 33 & 34 & 4 \end{array}$$

מה יזכיר הימנו (second Check (a , 4)

רשום את המקבב אחר ביצוע הפקנץיה `secondCheck`. אין צורך להראות מעקב אחר הפקנץיה `firstCheck`.

ג. מה מבצעת הפונקציה ?second Check /המשך בעמוד 4/

2. רשימה "חישובונית" L היא רשימה ששדה התוכן שלה מייצג ביטוי חישובוני.

הביטוי החישובוני מורכב משלושה חלקים:

- מספר שלם גדול לפחות

- תו אחד מבין ארבעת התווים:

+ המייצג חיבור

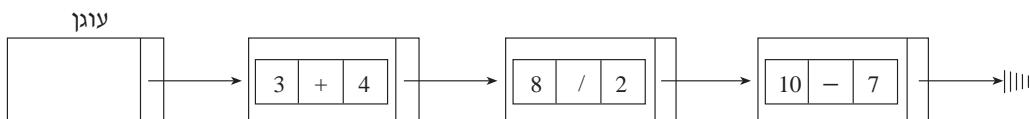
- המייצג חיסור

* המייצג כפל

/ המייצג חילוק

- מספר שלם גדול לפחות

דוגממה לרשימה "חישובונית" L:



א. הגדר בסביבת העבודה את טיפוס שדה התוכן של איבר ברשימה "חישובונית" L.

ב. ממש בסביבת העבודה תת-תכנית calculate, שתקבל רשימה "חישובונית" L

ומוקם k ברשימה. k הוא מקום בו L שאיןו סופי-רשימה ואיןו עוקן-רשימה.

התת-תכנית תחזיר את התוצאה המתקבלת מהביטוי החישובוני הנמצא באיבר

שבמקום k.

ג. ממש בסביבת העבודה תת-תכנית sumExpressions, שתקבל רשימה

"חישובונית" L, ותחזיר את הסכום הכלול של תוצאות הביטויים החישובוניים

הנמצאים ברשימה. בעבור רשימה ריקה יוחזר 0.

עליך להשתמש בתת-תכנית calculate.

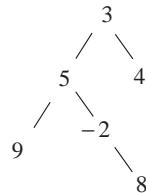
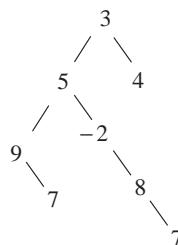
בעבור הרשימה "חישובונית" L בדוגמה הנתונה, התת-תכנית sumExpressions

תחזיר 14.

הערה: אין צורך למש בסביבת העבודה את הפעולות של ממש רשימה.

לפניך פעולה: 3.

<p>הפעולה מקבלת מספר שלם N ועץ בינארי T לא ריק שערכיו הם מספרים שלמים. לכל עלה בעץ T שערכו גדול מ- N, הפעולה מוסיפה בן ימני שערכו N. הנחה: העץ T מאותחל.</p>	<p>הוסף_עלים (N, T)</p>
--	--------------------------------

דוגמה: נתון העץ T לאחר זימון הפעולה **הוסף_עלים (7, T)**, העץ T הוא:כתב אלגוריתם שיממש את הפעולה **הוסף_עלים (N, T)**.

- במכוון "אEROBİKA" יש שיעורי התעמלות אEROBİT. השיעורים מתקיים ביוםים א-ה בשעות 7:00-22:00. כל שיעור מתחילה בשעה עגולה ונמשך שעה אחת. בכל שעה פועל במקוון שיעור אחד בלבד, ויכולים להשתתף בו לכל היותר 25 אנשים.
- אדם יכול להירשם לשיעור אחד בלבד. בהרשמה הוא מצין את היום בשבוע ואת השעה של השיעור שלו. מבקש להירשם, ומוסר את שמו ואת מספר הטלפון הנייד שלו. ההרשמה לשיעור, ביום ובשעה המבוקש, היא למשך שנה שלמה.
- אם אדם מעוניין להירשם לשיעור מסוים ואין מקום בשיעור, הוא נרשם כמתינו ומוסר את הפרטים האלה: השם, מספר הטלפון הנייד, היום והשעה של השיעור המבוקש.
- א. יציג את המידע הדרוש לניהול הרישום לשיעור במקוון "אEROBİKA".
- ב. כתוב אלגוריתם **משתתפים_בשיעור** שידפיס את השמות ואת מספרי הטלפון של כל המשתתפים בשיעור מסוים. רשום את הפרמטרים שהאלגוריתם מקבל ותיעוד.
- ג. כתוב אלגוריתם **מקום_פנוי** שיקבל יום ושעה של שיעור שיש בו מקום פנוי, ויבזץ אליו את הרעיון מבין המתינוים זהה השיעור שביקש. האלגוריתם ימחק את רישומו של האדם כמתינו. רשום את הפרמטרים שהאלגוריתם מקבל ותיעוד.

פרק שני (50 נקודות)

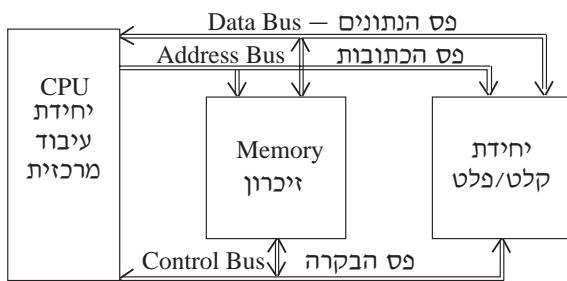
בפרק זה שאלות בבחמישה מסלולים שונים. ענה רק על שאלות במסלול שלמדת.

מערכות מחשב ואסמבילר

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

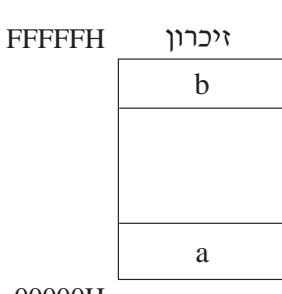
5. לפניך תיאור סכמטי של מערכת ובה יחידת עיבוד מרכזית, פסי מעבד, זיכרון,

יחידת קלט / פלט.



הזיכרון מכיל יחידת זיכרון לקריאה/כתיבה ויחידת זיכרון לקריאה בלבד. אחת מהן ממוקמת בתחילת מרחב הכתובות של הזיכרון, והאחרת ממוקמת בסוף מרחב הכתובות של הזיכרון. הגודל של כל יחידה הוא 64K בתים.

א. לפניך איור של הזיכרון ובו מסומנים האזוריים: a המתחילה בכתובת H00000, ו- b המסתום בכתובת FFFFFFFH.



(1) איזו יחידת זיכרון יש למקום בתחילת מרחב הכתובות של הזיכרון – אזור a,

ואיזו יחידת זיכרון יש למקום בסוף מרחב הכתובות של הזיכרון – אזור b ? נמק.

(2) מהי כתובות הסיום של יחידת הזיכרון הממוקמת באזור a, ומהי כתובות ההתחלה של יחידת הזיכרון הממוקמת באזור b ?

ב. מידע עבור מיחידת העיבוד המרכזית בפס הכתובות ובפס הנתונים אל היחידות השונות.

בailo קווים בפס הקריאה משתמשת יחידת העיבוד המרכזית כדי לקבוע איזו יחידה תקבל את המידע? נמק. בתשובהך התייחס לכל אחת משלוש היחידות: יחידת קלט/פלט, יחידת זיכרון לקריאה/כתיבה, יחידת זיכרון לקריאה בלבד.

.6. בתחילת תכנית במשפט אסמלר הוגדרו נתונים בצורה זו:

DATA SEGMENT

```

KAM DB 04H
ARR1 DB 04H, 10H, 1CH, 5DH
ARR2 DB 02H, 08H, 0DH, 1EH
DATA ENDS

```

שים לב: הנתונים והכתובות הם הקסדצימליים.

לפניך קטע התכנית:

```

MOV      AX, DATA
MOV      DS, AX
XOR      CX, CX
LEA      BX, ARR1
LEA      DI, ARR2
LEA      SI, KAM
MOV      CL, KAM
MOV      AH, 0
AGA1:   MOV      DH, [DI]
        INC      DI
        SHL      DH, 1
        CMP      DH, [BX]
        JNZ      COUNT1
        INC      AH
COUNT1: MOV      [BX], DH
        INC      BX
        LOOP    AGA1
        MOV      [SI], AH
SOF:    NOP

```

- א. עקב בערת טבלת מעקב אחר ביצוע קטע התכנית, ורשום מה יהיה התוכן של תאי הזיכרון.
- ב. מה יהיה התוכן של תא הזיכרון KAM בסיום קטע התכנית?
- ג. מה מבצע קטע התכנית?

7. סדרה חשבונית היא סדרה שבה ההפרש בין שני איברים עוקבים הוא גודל קבוע. תלמיד כתב במשפט אסמלר קטע תכנית לחישוב האיברים של סדרה חשבונית. האיבר הראשון של הסדרה נמצא בתא הזיכרון FIRST, וההפרש בין שני איברים עוקבים נמצא בתא הזיכרון DIFF. קטע התכנית סופר כמה איברים איזוגניים יש בסדרה. חישוב הסדרה יסתתיים כאשר חושבו COUNT איברים או כאשר יש גלישה באיבר האחרון שחוושב. הנ吐ונים הוגדרו בצהה זו:

DATA SEGMENT

FIRST	DB 243
DIFF	DB 7
COUNT	DW 5
DATA ENDS	

לפניך קטע התכנית שכתב התלמיד. מהקטע הושמו פקודות ואופרנדים.

START:	MOV	AX, DATA
	MOV	DS, AX
	MOV	CX, COUNT
	MOV	DL, 0
	MOV	BL, DIFF
	XOR	AX, AX
	MOV	AL, FIRST
AGAIN:	AND	AL, []
	JZ	NEXT
	INC	DL
NEXT:	MOV	AL, []
	ADD	AL, BL
	[]	FINISH
	MOV	FIRST, AL
	[]	[]
FINISH:	NOP	

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

במפעל למשקאות ערכו מבחן טעימה ל-8 סוגים חדשים של מיצים שסומנו במספרים 0-7. ב מבחן הטעימה השתתפו 5 טועמים מומחים. כל טעם בדק כל סוג של מיץ פעם אחת, וסימן בטופס מיוחד אם סוג המיץ לטעמו (1) או לא (0).

דוגמה: לפניך טופס של אחד הטועמים שציין שסוגי המיצים 0, 4, 5, 7 הם לטעמו:

7	6	5	4	3	2	1	0	סוג המיץ
תוצאות הטעימה								
1	0	1	1	0	0	0	1	

תוצאות הטעימה של הטועמים מאוחסנות בזיכרון החל מהכתובת TASTER, כל טופס מאוחסן בבית אחד.
עבור כל סוג מיץ נוצר במערך RESULTS מספר הטועמים שמייצ זה היה לטעם.

הנתונים הוגדרו כך:

TASTER DB 0B1H , 03H , 2BH , 0A6H , 0AAH : תוצאות הטעימה של הטועמים:
RESULTS DB 8 DUP(0) : מערך לצבירת תוצאות הטעימה:

שים לב: הנתונים הם הקסדצימליים.

לפניך קטע תכנית המטפל בתוצאות טעימה של טעם אחד:

LEA	BX , TASTER
MOV	DL , 1
MOV	SI , 0
MOV	AL , [BX]
LOOP1: SHR	AL , 1
JC	YES
CMP	SI , 7
JZ	FINISH
INC	SI
JMP	LOOP1
YES: ADD	RESULTS[SI] , DL
JMP	LOOP1
FINISH:	

א. קטע התכנית אינו מבצע את הנדרש. הוסף הוראות, כדי שקטע התכנית יבצע את הנדרש. ציין היכן בקטע התכנית תמוך את ההוראות שהוספה.

ב. הוסף הוראות לקטע התכנית, כדי שייטפל בתוצאות הטעימה של 5 טועמים. הקטע יסכם לכל סוג מיץ את מספר הטועמים שציינו שהוא לטעם, ויאחסן תוצאה זו במקום המתאים במערך RESULTS.

הערה: בסעיפים א-ב אין צורך להעתיק את קטע התכנית.

תורת המחשב

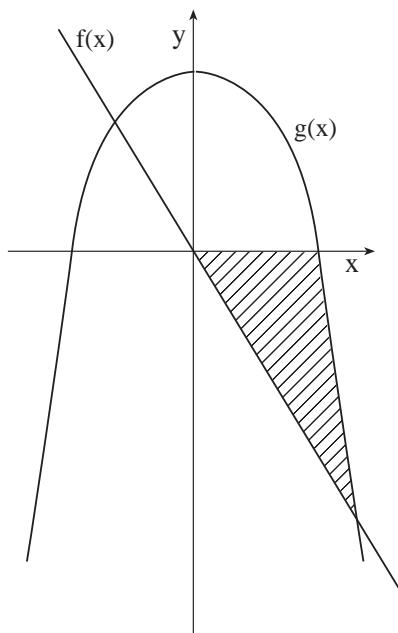
אם למדת מסלול זה, ענה על שתיים מהשאלות 9 - 12 (לכל שאלה – 25 נקודות).

9. נתונות שתי הפונקציות: $f(x) = -2x$

$$g(x) = -x^2 + 6$$

כתוב תכנית, שתשתמש בשיטת הטרפז, לחישוב השטח הטעום בין הפונקציה $f(x)$,

הפונקציה $g(x)$ וציר ה- x (השטח המוקווקו בציור), בדיק ש 10^{-6} .



10. נגידר סדרה של גרפים לא מקוונים באופן זהה:

בגרף G_0 יש צומת אחד, המסומן ב- V_0 , ואין בו קשתות.

כל גרף G_n יוגדר בעזרת הגרף G_{n-1} באופן זהה:

— נוסיף לגרף G_{n-1} צומת V_n .

— אם n זוגי, נוסיף לגרף G_{n-1} קשתות שיחברו את הצומת V_n

אל כל הצלמתים שהאינדקס שלהם אי-זוגי.

— אם n אי-זוגי, נוסיף לגרף G_{n-1} קשתות שיחברו את הצומת V_n

אל כל הצלמתים שהאינדקס שלהם זוגי.

א. צייר את הגרפים: G_0, G_1, G_2, G_3, G_4 . שים לב: 0 הוא מספר זוגי.

ב. מה תהיה דרגת הצומת V_n בגרף G_n ?

ג. מה תהיה דרגת הצומת V_{n+1} בגרף G_{n+1} ?

ד. לפניך טענה.

עבור n כלשהו שלם וחובי:

לכל הצלמתים בגרף G_n יש אותה דרגה

או

קיימים שני מספרים שלמים וחוביים x ו- y , כך שלחלק מהצלמתים

בגרף G_n יש דרגה x ולשאר הצלמתים בגרף G_n יש דרגה y .

קבע אם הטענה נכונה או אינה נכונה, וنمוק את קביעותך.

11. לפניך השפה L מעל הא"ב $\{a, b\}$:

$$L = \{ a^m b^n a^n \mid m, n \geq 1 \text{ (} m + n \text{ זוגי)} \}$$

בנה אוטומט מחסנית שיקבל את השפה L .

12. לסטודנטים במכוללה מסוימת מתקבלים תלמידים על פי ציוניהם בבחינות הבגרות. הקייטריוונים לקבלו ללימודים במכוללה הם אלה:
- * תעודת בגרות ובה לפחות 5 מקצועות.
 - * הציונים במתמטיקה, באנגלית ובמקצוע נוסף צריכים להיות גבוהים מ- 84.
 - * כל הציונים חייבים להיות גבוהים מ- 74.

במכוללה מוקודדים את הציונים באופן זהה:

- A : 100-95
- B : 94-85
- C : 84-75
- D : 74-55

האותיות המייצגות את הציונים מוקלדות כמחוזות משמאלי ימין באופן זהה:

האות הראשונה שמקלדת מייצגת את הציון במתמטיקה,
האות השנייה שמקלדת מייצגת את הציון באנגלית,
ולאחר מכן מוקלדות האותיות המייצגות את הציונים במקצועות האחרים.

א. לפני חמש מחוזות, ו- 7.

כל אחת מהמחוזות מייצגת ציונים של תלמיד המבקש להתקבל ללימודים
במכוללה.

קבע כל אחת מהמחוזות אם התלמיד יתקבל ללימודים במכוללה.
نمק את קביעותיך.

- | | |
|------------|-----|
| AACCCCBBCA | i |
| ABABDBB | ii |
| BBBBBBA | iii |
| AAAA | iv |
| AACCCC | v |

ב. בנה אוטומט סופי דטרמיניסטי, שיקבל כקלט מחוזות המייצגים ציונים של תלמיד
ויבדוק אם התלמיד יתקבל ללימודים במכוללה.

מודלים חישוביים

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

13. לפניך השפה L מעל הא"ב $\{a, b\}$:

$$L = \{a^m b^n a^n \mid m, n \geq 1\}$$

בנה אוטומט מחסנית שיקבל את השפה L .

14. בנה מכונה טיורינג שתקבל קלט מילה מעל הא"ב $\{a, b, c\}$.

המכונה תבודוק אם לפניהם a , בצדד לו, יש c אחד לפחות.

אם כן – המכונה תעוצר במצב מקבל, אם לא – המכונה לא תעוצר.

אם הקלט הוא המילה הריקה, המכונה תעוצר במצב מקבל.

דוגמה:

בעבור מילת הקלט $bccabcbcccab$ המכונה תעוצר במצב מקבל.

בעבור מילת הקלט $cbabbcab$ המכונה לא תעוצר.

15. לפניך שלוש השפות L_1, L_2, L_3 :

$$L_1 = \{a^n b^{n+2} \mid n \geq 0\}$$

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

$$L_3 = \{a^{2n} b^{n+3} \mid n \geq 0\}$$

א. קבע לכל אחת מהשפות L_1, L_2, L_3 אם היא רגולרית או אינה רגולרית.

נמק את קביעותיך.

ב. מהי השפה $L_2 \cap L_1$? נמק.

ג. האם מספר ה- d -ים במילה השיכת לשפה $(L_2 \cdot R \cdot L_2)$ הוא זוגי? נמק.

ד. קבע לכל אחת מהמלילים w_1, w_2 אם היא שיכת לשפה $\bar{L}_1 \cap L_3$.

נמק את קביעותיך.

$$w_1 = aaaabbba$$

$$w_2 = bbb$$

16. לסטודנטים במכלה מסוימת מתקבלים תלמידים על פי ציוניהם בבחינות הבגרות. הクリיטריונים לקבלת לימודיים במכלה הם אלה:
- * תעודת בגרות ובה לפחות 5 מקצועות.
 - * הציונים במתמטיקה, באנגלית ובמקצוע נוסף צריכים להיות גבוהים מ- 84.
 - * כל הציונים חייבים להיות גבוהים מ- 74.

במכלה מוקדדים את הציונים באופן זהה:

- A : 100-95
 B : 94-85
 C : 84-75
 D : 74-55

האותיות המייצגות את הציונים מוקלדות כמחוזות משמאלי לيمן באופן זהה:

האות הראשונה שМОקלדת מייצגת את הציון במתמטיקה,
 האות השנייה שМОקלדת מייצגת את הציון באנגלית,
 ולאחר מכן מוקלדותאותיות המייצגות את הציונים במקצועות האחרים.

א. לפניך חמש מחוזות, 1-7.

כל אחת מהמחוזות מייצגת ציונים של תלמיד המבקש להתקבל ללימודים במכלה.

קבע כל אחת מהמחוזות אם התלמיד יתקבל ללימודים במכלה.
 נמק את קביעותיך.

- | | |
|------------|-----|
| AACCCCBBCA | i |
| ABABDBB | ii |
| BBBBBBA | iii |
| AAAAA | iv |
| AACCCC | v |

ב. בנה אוטומט סופי דטרמיניסטי, שיקבל כקלט מחוזות המייצגים ציונים של תלמיד ויבדוק אם התלמיד יתקבל ללימודים במכלה.

חישוב מקבילי ומבוזר

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

17. לפניך תכנית ובה שני תהליכי Proc1 ו- Proc2 המתבצעים במקביל. התכנית משתמשת במשתנה גלובלי N.

<pre>program TestSh; var N: integer; process Proc1; var A: integer; begin read(A); while N < A do begin N := N + 1; write(N); end; end;</pre>	<pre>process Proc2; var B, C: integer; begin read(B); C := B; while B > 0 do begin B := B - 2; if B = N then begin N := 0; B := C; end; end; end;</pre>
<pre>begin {main} N := 0; parbegin Proc1; Proc2; parend; end.</pre>	

בכל אחד משני הסעיפים א' ו- ב' אתה נדרש להציג תסריט. عليك לרשום את התהיליך המבוצע בכל שלב, את ההוראה המבוצע, את ערכי המשתנים ואת הפלט (אם יש פלט).

- א. בעבור הקלט 3 ל- A ו- 4 ל- B הציג תסריט עד שיתקבלו הפלטים (משמאלי לימין):

. 1 , 2 , 1 , 2

- ב. בעבור הקלט 1 ל- A ו- 3 ל- B הציג תסריט עד שיתקבלו הפלטים (משמאלי לימין):

. 0 , 0 , 1

במצפה הכוכבים "עין לשחקים" נותנים שירותים טלקופ דרך האינטרנט למוניים בלבד. לכל מוני יש מספר מוני. אין שני מוניים עם אותו מספר מוני. התוור לשימוש בטלקופ מנהל כמערכת מבוזרת בעזרת תוכנה המותקנת על מחשבים המוניים. מחשבים המוניים מחוברים תמיד לרשת, ואין מנתקים אף פעם. התוכנה בנוייה על בסיס אלגוריתם RA, ופועלת כל הזמן במחשב של כל מוני. בתוכנה יש שני תהליכי: תהליך ראשי ותהליך לטיפול בבקשת, הפעלים במקביל ומאפשרים ניהול תור בלי מחשב מרכזי. כאשר שני מוניים או יותר מבקשים להשתמש בטלקופ באותו זמן, ניתנת עדיפות למוני שמספר המוני שלו קטן יותר.

לפניך האלגוריתם:

תהליך ראשי:
<p>"אני רוצה להשתמש בטלקופ" → false</p> <p>בצע תמיד:</p> <p>קטע לא קרייטי { מהלך חיים וגיל }</p> <p>"אני רוצה להשתמש בטלקופ" → true</p> <p>עבור כל מחשב Z שאינו המחשב שלי בצע:</p> <p>שלח בקשה (כתובת מחשב Z, מספר המוני שלי, כתובת המחשב שלי)</p> <p>עבור מספר מוניים פחות אחד בצע:</p> <p>קבל אישור</p> <p>{ השימוש בטלקופ }</p> <p>"אני רוצה להשתמש בטלקופ" → false</p> <p>עבור כל המוניים המתאימים בצע:</p> <p>שלח אישור</p>
תהליך לטיפול בבקשת:
<p>בצע תמיד:</p> <p>קבל בקשה (מספר המוני של השולח, כתובת מחשב שלח)</p> <p>אם לא "אני רוצה להשתמש בטלקופ"</p> <p>שלח אישור (כתובת מחשב שלח)</p> <p>אחרת אם מספר המוני שלי < מספר המוני של השולח אז</p> <p>שלח אישור (כתובת מחשב שלח)</p> <p>אחרת</p> <p>הכנס לרשימת המתאימים (כתובת מחשב שלח)</p>

בסעיפים א-ג שלפני, כאשר אתה מציג תסריט עלייך לרשום את התהיליך המתבצע, את ההוראה המתבצעת, ואת ערכי הפרמטרים בהוראות הכלולות פרמטרים.

א. נניח שמנוי 100 אינו מעוניין להשתמש בטלסקופ, והוא מקבל ממוני 50
וממוני 102 בקשות להשתמש בטלסקופ.

i אם מני 100 שולח אישור לשניהם? נמק את תשובתך.

ii אם מני 100 יכול להניח שמנוי 50 ישמש בטלסקופ לפני מני 102? נמק
את תשובתך.

ב. הנה שני המנויים היחידים במערכת הם מני 1 ומני 2.
שניהם שולחים זה לזה בקשות לשימוש בטלסקופ.

בנהנעה שכל אחד משני מנויים אלה שולח בקשה למניי האחר לפני שהוא מקבל את
הבקשה שנשלחה אליו, האם ייתכן שמנוי 1 יוכל להשתמש בטלסקופ בלי שמנוי 2
יימצא ברשימה המתאימים שלו?

אם כן – הצג תסריט מתאים, אם לא – הסבר מדוע.

ג. הצג תסריט שבו שני מנויים פונים יחד לשימוש בטלסקופ (נמצאים יחד בקטע
הקריטי).

19. נבחרת בית ספר בקפיצה לגובה, שבה 12 קופצים, מקיימת תחרות הינה פנימית. כל קופץ מקבל מספר בין 1 ל-12, והוא קופץ שלוש קפיצות. התוצאה הקובעת בעבור כל קופץ היא התוצאה הטובה ביותר משלוש הקפיצות שלו. כדי למצוא את תוצאה הקפיצת הטובה ביותר משלוש התוצאות הקובעות של 12 קופצים נבחרת, פיתחו אלגוריתם הכלול 4 תהליכי המבצעים במקביל:
- * תהליך 1, תהליך 2, תהליך 3 – הם תהליכי זהים.
* תהליך 4 מוצא את התוצאה הטובה ביותר משלוש התוצאות הקובעות של 12 קופצים.

האלגוריתם משתמש במשתנים הגלובליים מטיפוס שלם: N , $TempMax$, $TotalMax$ ובספורים: $Sem2$, $Sem1$. כמו כן, האלגוריתם משתמש ב פעולה האטומית **קלוט-קופץ** (P, A, B, C) המתניתה עד שמתקבלים 4 קלטים: למשתנה P – מספר הקופץ, ולמשתנים A, B, C – תוצאות הקפיצות של הקופץ P .

תהליך 3 $\{ \text{זהה ל} \text{תהליך 1} \}$	תהליך 2 $\{ \text{זהה ל} \text{תהליך 1} \}$	תהליך 1 כל עוד $N \leq 12$ בצע $N \leftarrow N + 1$ קלוט-קופץ (P, A, B, C) $\text{Max} \leftarrow A$ אם $B > \text{Max}$ אז $\text{Max} \leftarrow B$ אם $C > \text{Max}$ אז $\text{Max} \leftarrow C$ wait (Sem1) $\text{TempMax} \leftarrow \text{Max}$ signal (Sem2)
		תהליך 4 עברו I מ-1 עד 12 בצע wait (Sem2) אם $\text{TempMax} > \text{TotalMax}$ אז $\text{TotalMax} \leftarrow \text{TempMax}$ signal (Sem1)
		האלגוריתם הראשי $N \leftarrow 1$ $\text{TempMax} \leftarrow 0$ $\text{TotalMax} \leftarrow 0$ $\{ \text{ספ/or} \} \text{ Sem1} \leftarrow 1$ $\{ \text{ספ/or} \} \text{ Sem2} \leftarrow 0$ parbegin תהליך 1; תהליך 2; תהליך 3; תהליך 4; parened הדפס TotalMax

א. באיזה מנגנון תיאום האלגוריתם משתמש, ובאיזה מנגנון זה בא לידי ביטוי באלגוריתם?

ב. באלגוריתם הנתון יש כשל מקבילי, שבגללו אחד או יותר מהתהליכיים אינם מסתויים, ולכן באלגוריתם הראשי לא מודפס דבר.

i. הסבר מדוע נוצר הכשל המקבילי זהה.

ii. הוסף לאלגוריתם הוראות כדי למנוע את הטעיה הטעיה המקבילי.

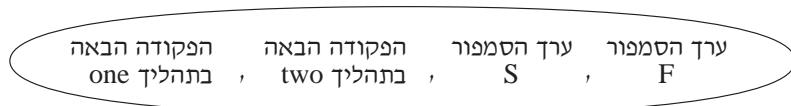
ג. המאמין מעוניין לקבל את מספרו של הקופץ שהשיג את התוצאה הטובה ביותר. הרכיב את האלגוריתם המקבילי כדי שגם מטרה זו תושג.

20. לפניך תכנית מקבילית:

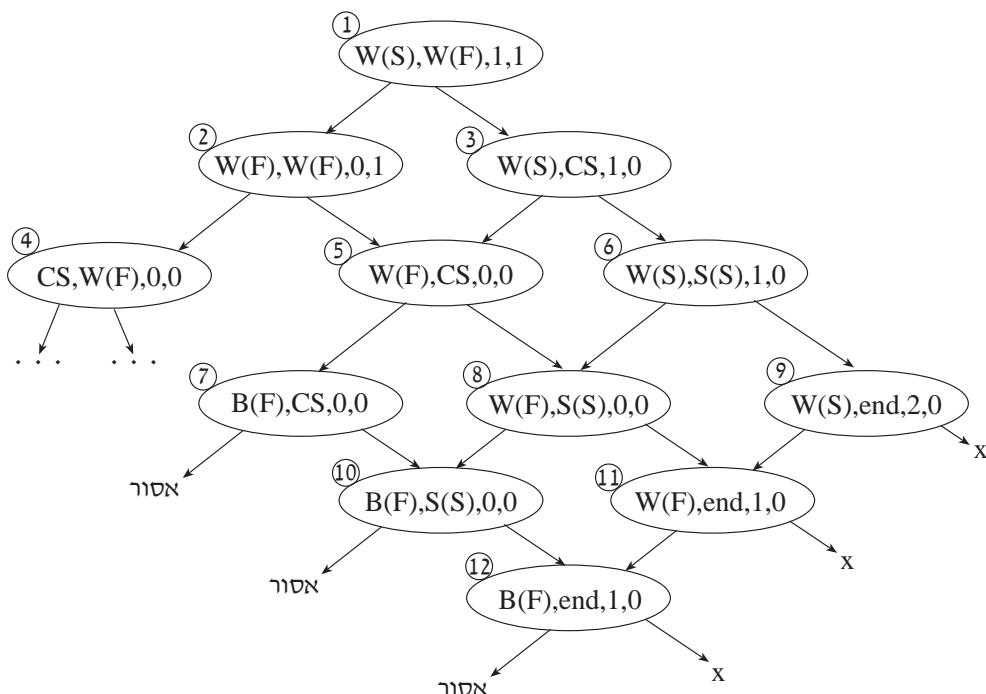
<pre> program best; var F: semaphore := init(1); S: semaphore := init(1); process one; begin wait(S); wait(F); {קטע קרייטי} signal(S); end; begin parbegin one; two; parend; end. </pre>	
<pre> process two; begin wait(F); {קטע קרייטי} signal(S); end; </pre>	

תלמיד בנה תרשימים מצבאים לתוכנית, אך לא סיים אותו.

שדר הצגת המידע במצב בתרשים מתואר בציור שלפניך:



CS מייצג את הקטע הクリיטי.
B מייצג תהליך חסום על ידי סמפור.



- א. העתק למחברתך את מצב מס' 4, והשלם את תרשימים המצבים. מס'ר את המצבים שהווספה מס'ר 13 ויאלץ.
- ב. האם ניתן קיפאון בתכנית? נמק באמצעות תרשימים המצבים.
- ג. האם יש בתכנית מניעה הדדית? הוכח באמצעות תרשימים המצבים או באופן מילולי.

תכנות מונחה עצמים

אם למדת מסלול זה ואתה כותב בגאווה, ענה על שתיים מהשאלות 21-24 שבעמודים 22-28.
(לכל שאלה – 25 נקודות)

אם למדת מסלול זה ואתה כותב ב- #C, ענה על שתיים מהשאלות 25-28 שבעמודים 30-36.
(לכל שאלה – 25 נקודות)

21. לפניך תכנית:

```
public class Stam
{
    public void B(int m , int n)
    {   System.out.println (m   n) ;   }
    public void f ()
    {   System.out.println("inside Stam");   }
}
public class B extends Stam
{
    protected int y;
    public B(int x)
    {   this.y = x;   }
}
public class C extends Stam
{
    public int x;
    public static void g()
    {   System.out.println("inside C ");   }
}
public class Test
{
    public static void main (String[] args)
    {
        C temp = new C();
    }
}
```

ענה על הסעיפים א-ה, בהתייחס לתוכנית הנתונה.

- א. האם אפשר לשנות את הערך של הוכונה x המוגדרת במחלקה C, מתוך שיטה
במחלקה Stam ?
אם כן – כתוב כיצד, אם לא – נמק מדוע.
- ב. i כתוב בשיטה main, הוראה או הוראות לזמן השיטה B המוגדרת
במחלקה Stam .
ii כתוב בשיטה main, הוראה או הוראות לזמן השיטה B המוגדרת
במחלקה B .
- ג. לפניו שלוש הוראות, i-iii .
כל אחת מהן, קבע אם אפשר להוסיף אותה בשיטה main .
נמק את קביעותיך בעזרת מאפיינים של תכונות מונחה עצמים.
Stam obj = new B(7); i
Stam obj = new B(); ii
B obj = new Stam(); iii
- ד. האם אפשר לזמן את השיטה () f המוגדרת במחלקה Stam , מתוך שיטה
במחלקה B ?
אם כן – כתוב את הזמן, אם לא – נמק מדוע.
- ה. האם אפשר לזמן את השיטה () g המוגדרת במחלקה C , מתוך השיטה main ?
אם כן – כתוב את הזמן, אם לא – נמק מדוע.

.22 נתנות שלוש המחלקות: NameC , NameB , NameA ומחלקה ראשית Test . עוקב בעזרת טבלת מעקב אחר ביצוע השיטה main במחלקה Test , ורשות את הפלט. על הטבלה לכלול את ערכי המשתנים, ובמעבר כל עצם – את ערכי התכונות שלו.

```
public class NameA
{
    protected int x;
    public NameA(int x)
    { this.x = x; }
    public int getX()
    { return x; }
    public String toString()
    { return "NameA:x= "+ x; }
    public String display()
    { return "display NameA"; }
}

public class NameB extends NameA
{
    protected int y;
    public NameB(int x , int y)
    {
        super(x);
        this.y = y;
    }
    public String toString()
    { return "NameB: x= "+ getX()+ "y= "+ this.y; }
}

public class NameC extends NameB
{
    protected int z;
    public NameC(int x , int y , int z)
    {
        super(x , y);
        this. z = z;
    }
    public String toString()
    { return "NameC: x= "+ x + " y= "+ y + " z= " + z; }
    public String display()
    { return "display NameC"; }
}
```

```
public class Test
{
    public static void main (String [] args)
    {
        NameC nc1 = new NameC(1, 2, 7);
        System.out.println (nc1.toString());
        NameA people [] = new NameA [4];
        people [0] = new NameA(6);
        people [1] = new NameB(8, 9);
        people [2] = new NameC(4, 0, 11);
        people [3] = new NameC(3, 2, 4);
        people [3] = (NameA) people [2];
        int count = 0;
        for (int i=0; i < people.length; i++)
        {
            System.out.println (people [i].display());
            if (people[i]. getX() > 5)
            {
                count++;
            }
        }
        System.out.println(count);
        count = 0;
        for (int i=0; i < people.length; i++)
        {
            if (people[i] instanceof NameC)
            {
                count++;
            }
        }
        System.out.println(count);
    }
}
```

23. חברת מוניות החליטה למחשב את ניהול המוניות שלה. לצורך כך הגדרו את המחלקה

מונית (Taxi), שלה התכונות האלה:

- מספר רישוי – String taxiId
- שם נהג – String driverName
- מספר מקומות לנוסעים (מלבד הנהג) – int numPass
- האם המונית פנואה – boolean available

התכונה available מתחילה ל- true

כמו כן הוגדרו פועלות האחזר:

getId() , getName() , getNumPass() , isAvailable()

א. ממש בגיאויה במחלקה Taxi את השיטות הבונות האלה:

 i. שיטה המקבלת כפרמטרים שם נהג ומספר רישוי,

 ii. שיטה המקבלת כפרמטרים שם נהג, מספר רישוי, מספר מקומות לנוסעים,

 iii. שיטה המקבלת כפרמטרים שם נהג, מספר רישוי, מספר מקומות לנוסעים, ומתחילה בהתאם את taxiId , driverName ו- numPass .

ב. ממש בגיאויה במחלקה Taxi שיטה taxiBusy המעדכנת את המונית להיות

לא פנואה.

ג. בחברה רוצים להגדיר את המחלקה תחנת מוניות (TaxiStation), המשמשת במחלקה מונית (Taxi). תחנה שייכות לכל היותר 80 מוניות. התכונות של המחלקה TaxiStation הן:

- שם תחנה
- מספר מוניות השייכות לתחנה
- מערך של המוניות השייכות לתחנה

המוניות השייכות לתחנה מיוצגות במערך המוניות באופן רצוף מתחילה המערך. ממש בינויו את המחלקה TaxiStation. שימוש המחלקה יכול רק את הסעיפים i - iii שלפניך:

- i הגדרת התכונות של המחלקה.
- ii שיטה המוסיפה מונית לתחנה. השיטה מקבלת את שם הנהג, את מספר הרישוי, ואת מספר המיקומות לנוסעים.
- iii שיטה המקבלת מספר נוסעים, ומחזירה מספר רישיוי של מונית פנוייה כלשהי המתאימה להסעת מספר נוסעים כמפורט, ומעדכנת את המוניות להיות לא פנויות. אם אין מונית פנוייה מתאימה, השיטה מחזירה 1.

24. המחלקה Rational מגדירה את טיפוס הנתונים **שבר** (מספר רציונלי). שבר מיוצג על ידי המונה שלו ועל ידי המכנה שלו.

לפניך חלק מהשיטות של המחלקה Rational:

Rational(int n1 , int n2)	שיטה בונה המחזיר שבר שהמונה שלו הוא n_1 והמכנה שלו הוא n_2
int getN1()	שיטה המחזיר את המונה של שבר
int getN2()	שיטה המחזיר את המכנה של שבר

המחלקה G מגדירה את טיפוס הנתונים **קבוצה**, שהוא אוסף עצמים מטיפוס מספר העצמים בקבוצה אינו קבוע. אין חשיבות לסדר הופעת העצמים בקבוצה.

לפניך חלק מהשיטות של המחלקה G:

G()	שיטה הבונה קבוצה ריקה
void gInsert(Rational num)	שיטה המכניתה לקבוצה את השבר num
Rational gRemove()	שיטה המחזיר שבר <u>בלשוח</u> הנמצא בקבוצה ומוחקת אותו מהקבוצה
boolean gEmpty()	שיטה המחזיר "אמת" אם הקבוצה ריקה, ו"שקר" אחרת

משמש בינויו, במחלקה G, את השיטות בסעיפים א-ב. אפשר להשתמש בשיטות הנתונות של המחלקות Rational ו- G בלבד למש אתן.

G buildG(int n , int m)	שיטה מקבלת מספר שלם וחוביי n , ומספר שלם וחוביי m . השיטה מחזירה קבוצה המכילה את כל השברים שהמונה שלהם הוא מספר בין 1 ל- n (כולל), והמכנה שלהם הוא m .
-------------------------	--

Rational sum(int m)	שיטה מקבלת מספר שלם m , ומחזירה שבר שהוא הסכום של השברים בקבוצה שהמכנה שלהם הוא m . הנה שיש בקבוצה לפחות שבר אחד שהמכנה שלו m . הערה: המונה של השבר המוחזר הוא סכום המוניהם של השברים האלה, והמכנה שלהם הוא m .
---------------------	---

שים לב

השאלות בתכנית מונחה עצמים

לפתרונות ב- #C החל מעמ' 30

שים לב: אם אתה כותב ב- C#, ענה על שתיים מהשאלות 25-28.

25. לפניך תכנית:

```
public class Stam
{
    public void B(int m , int n)
    {   Console.WriteLine(m + n);   }
    public void f()
    {
        Console.WriteLine("inside Stam");
    }
}
public class B : Stam
{
    protected int y;
    public B(int x)
    {
        this.y = x;
    }
}
public class C : Stam
{
    public int x;
    public static void g()
    {
        Console.WriteLine ("inside C");
    }
}
public class Test
{
    public static void Main()
    {
        C temp = new C();
    }
}
```

ענה על הסעיפים א-ה, בהתייחס לתוכנית הנתונה.

א. האם אפשר לשנות את הערך של הוכנה x המוגדרת במחלקה C, מתוך שיטה
במחלקה Stam?

אם כן – כתוב כיצד, אם לא – נמק מדוע.

ב. i כתוב בשיטה Main, הוראה או הוראות לזמן השיטה B המוגדרת
במחלקה Stam.

ii כתוב בשיטה Main, הוראה או הוראות לזמן השיטה B המוגדרת
במחלקה B.

לפניך שלוש הוראות, i-iii.

לכל אחת מהן, קבע אם אפשר להוסיף אותה בשיטה Main.

نمק את קבועותיך בעזרת מאפיינים של תכונות מונחה עצמים.

Stam obj = new B(7); i

Stam obj = new B(); ii

B obj = new Stam(); iii

האם אפשר לזמן את השיטה f() המוגדרת במחלקה Stam, מתוך שיטה
במחלקה B?

אם כן – כתוב את הזימנו, אם לא – נמק מדוע.

ה. האם אפשר לזמן את השיטה g() המוגדרת במחלקה C, מתוך השיטה Main?

אם כן – כתוב את הזימנו, אם לא – נמק מדוע.

.26 נתונות שלוש המחלקות: NameC , NameB , NameA . Test ומחלקה ראשית עוקב בעזרת טבלת מעקב אחר ביצוע השיטה Main במחלקה Test , ורשום את הפלט. על הטבלה לכלול את ערכי המשתנים, ובמעבר כל עצם – את ערכי התכונות שלו.

```
public class NameA
{
    protected int x;
    public NameA(int x)
    {   this.x = x;   }
    public int X
    {
        get
        {   return x;   }
    }
    public override string ToString()
    {   return " print NameA:x = " + x;   }
    public virtual string Display()
    {   return "display NameA";   }
}

public class NameB : NameA
{
    protected int y;
    public NameB(int x, int y):base (x)
    {   this.y = y;   }
    public override string ToString()
    {   return " print NameB: x = " + x + "y = " + this.y;   }
}

public class NameC : NameB
{
    protected int z;
    public NameC (int x , int y, int z):base (x , y)
    {   this.z = z;   }
    public override string ToString()
    {   return " print NameC: x = " + x + " y = " + y + " z = " + z;   }
    public override string Display()
    {   return "display NameC";   }
}
```

```
public class Test
{
    public static void Main (string [] args)
    {
        NameC nc1 = new NameC(1, 2, 7);
        Console.WriteLine(nc1.ToString());
        NameA [] people = new NameA [4];
        people [0] = new NameA(6);
        people [1] = new NameB(8, 9);
        people [2] = new NameC(4, 0, 11);
        people [3] = new NameC(3, 2, 4);
        people [3] = (NameA) people [2];
        int count = 0;
        for (int i=0; i < people.Length; i++)
        {
            Console.WriteLine(people[i].Display());
            if (people[i].X > 5)
            {
                count++;
            }
        }
        Console.WriteLine(count);
        count = 0;
        for (int i=0; i < people.Length; i++)
        {
            if (people[i] is NameC)
            {
                count++;
            }
        }
        Console.WriteLine(count);
    }
}
```

27. חברת מוניות החליטה למחשב את ניהול המוניות שלה. לצורך כך הגדרו את המחלקה

מונית (Taxi), שלה התכונות האלה:

- מספר רישוי – string taxiId
- שם נהג – string driverName
- מספר מקומות לנוסעים (מלבד הנהג) – int numPass
- האם המונית פנואה – bool available

התכונה available מתחילה ל- true.

כמו כן הוגדרו פעולות האחזר:

GetId() , GetName() , GetNumPass() , IsAvailable()

א. ממש ב- C# במחלקה Taxi את השיטות הבוניות האלה:

 i. שיטה המקבלת כפרמטרים שם נהג ומספר רישוי,

 ii. מתחילה בהתאם ל- numPass ו- taxiId , ומתחילה את driverName ו- taxiId .

ל להיות 4.

 iii. שיטה המקבלת כפרמטרים שם נהג, מספר רישוי, מספר מקומות לנוסעים,

 iv. מתחילה בהתאם ל- numPass ו- taxiId , driverName ו- taxiId .

ב. ממש ב- C# במחלקה TaxiBusiy שיטה המעדכנת את המונית להיות

לא פנואה.

ג. בחברה רוצים להגדיר את המחלקה תחנת מוניות (TaxiStation), המשמשת במחלקה מונית (Taxi). תחנה שייכות לכל היותר 80 מוניות. התכונות של המחלקה TaxiStation הן:

- שם תחנה
- מספר מוניות השייכות לתחנה
- מערך של המוניות השייכות לתחנה

המוניות השייכות לתחנה מיוצגות במערך המוניות באופן רצוף מתחילה המערך.

ממש ב-#C את המחלקה TaxiStation. מימוש המחלקה יכול רק את הפעולות ו-iii שלפניך:

- i הגדרת התכונות של המחלקה.
- ii שיטה המוסיפה מונית לתחנה. השיטה מקבלת את שם הנהג, את מספר הרישוי, ואת מספר המיקומות לנוסעים.
- iii שיטה המקבלת מספר נוסעים, ומחזירה מספר רישיי של מונית פנוייה כלשהי המתאימה להטעת מספר נוסעים כUMBOKSH, ומעדכנת את המוניות להיות לא פנויות. אם אין מונית פנוייה מתאימה, השיטה מחזירה -1.

28. המחלקה Rational מגדרה את טיפוס הנתונים **שבר** (מספר רצינלי). שבר מיוצג על ידי המונה שלו ועל ידי המכנה שלו.

לפניך חלק מהשיטות של המחלקה Rational:

Rational(int n1 , int n2)	שיטה בונה המחזיר שבר שהמונה שלו הוא n1 והמכנה שלו הוא n2
int GetN1()	שיטה המחזיר את המונה של שבר
int GetN2()	שיטה המחזיר את המכנה של שבר

המחלקה G מגדרה את טיפוס הנתונים **קבוצה**, שהוא אוסף עצמים מטיפוס מספר העצמים בקבוצה אינו קבוע. אין חשיבות לסדר הופעת העצמים בקבוצה.

לפניך חלק מהשיטות של המחלקה G:

G()	שיטה הבונה קבוצה ריקה
void GInsert(Rational num)	שיטה המכניסה לקבוצה את השבר num
Rational GRemove()	שיטה המחזיר שבר <u>בלשה</u> הנמצא בקבוצה ומוחקתו אותו מהקבוצה
bool GEmpty()	שיטה המחזיר "אמת" אם הקבוצה ריקה, ו"שקר" אחרת

ממש ב-C#, במחלקה G, את השיטות בסעיפים א-ב. אפשר להשתמש בשיטות הנתונות של המחלקות Rational ו- G בלי ממש אותן.

G BuildG(int n , int m)	שיטה מקבלת מספר שלים וחויבי n, ומספר שלם וחויבי m. השיטה מחזירה קבוצה המכילה את כל השברים שהמונה שלהם הוא מספר בין 1 ל- n (כולל), והמכנה שלהם הוא m.
-------------------------	---

Rational Sum(int m)	שיטה מקבלת מספר שלים m, ומחזיר שבר שהוא הסכום של השברים בקבוצה שהמכנה שלהם הוא m. הנח שיש בקבוצה לפחות שבר אחד שהמכנה שלו m. הערה: המונה של השבר המוחזר הוא סכום המונאים של השברים האלה, והמכנה שלהם הוא m.
---------------------	---

ב ה צ ל ח !