

מדעי המחשב - 2 יחידות לימוד
פתרון בחינת הבטחות

פרק א

שאלה 1:

sum	k	k אי-זוגי?	פלט
0			18
2	1	כן	
	2	לא	
8	3	כן	
	4	לא	
18	5	כן	
	6	לא	

הערה: אין חובת רישום עמודת התנאי.
עמודה זו מסייעת בביצוע המעקב.

שאלה 2:

טענת כניסה: 2 מספרים שלמים.
טענת יציאה: הערך המוחלט של ההפרש שבין שני המספרים.

אפשרות א:	אפשרות ב:
<pre>function sub (a, b : integer) : integer; begin sub := abs (a - b); end;</pre> <p>משפט הזימון: x := sub (5, 5);</p>	<pre>function sub (a, b : integer) : integer; begin if (a > b) sub := a - b else sub := b - a; end;</pre> <p>משפט הזימון: x := sub (5, 5);</p>

הערה: משפט הזימון יקבל כל שני מספרים זהים,

או פעמיים אותו משתנה: x := sub (k, k); בהנחה ש-k מוגדר ומאותחל.

שאלה 3:

א. הוראת קלט למחרוזת: `readln (tel);`

ב. אחת משתי האפשרויות הבאות:

II	I
<code>st1 := concat ('abc', tel);</code>	<code>st1 := 'abc' + tel;</code>

ג.

II	I
אורך המחרוזת הוא 5 <code>writeln (tel[4]);</code> <code>writeln (tel[5]);</code>	אורך המחרוזת הוא k <code>writeln (tel[k-1]);</code> <code>writeln (tel[k]);</code>

שאלה 4:

הערה: קטע תכנית אינו מחייב תיעוד (טענות כניסה ויציאה).

```
for i := 1 to 57 do
  begin
    d1 := a[i] mod 10;           {--- ספרת האחדות ---}
    d100 := a[i] div 100;       {--- ספרת המאות ---}
    if (d100 = d1 * 2) then
      writeln (a[i]);
  end;
```

שאלה 5:

count	num1	num2	num1 > num2	פלט
0				3
	15	8	כן	
1	8	7	כן	
2	7	3	כן	
3	3	4	לא	

א. טבלת מעקב:
(אין חובת רישום של עמודת ההוראה, או עמודת התנאי).

ב. דוגמת קלט: $2, 3, \dots$
(כל סדרת נתונים שבה המספר השני גדול מהראשון)

פרק ב

שאלה 6:

	1	2	3	4	5
a	8	7	6	5	2

א.

פלט	$a[k] - 1 \neq a[k+1]$	k	$a[1] = n$	n	flag = 1	flag
8 7	לא	1	כן	8		1
7 6	לא	2				
6 5	לא	3				
5 2	כן	4				0
הודעה ב					לא	

ב.

n = 8

	1	2	3	4	5
a	8	7	6	5	4

פלט "הודעה א"

ג.

flag ← 0

n = 9

	1	2	3	4	5
a	9	7	6	5	4

במקום (i)
אם במקום כלשהו במערך
אין מספר עוקב-יורד.

n = 9

	1	2	3	4	5
a	8	7	6	5	2

במקום (ii)
אם התא הראשון אינו 9
(שונה מ-n).

שאלה 7:

{--- קלט: 30 שמות, לכל שם 3 ציונים. ---}
 {--- פלט: שם התלמיד וציונו הסופי. מספר התלמידים שקיבלו 100. ---}

```
program grades ;
```

```
var
```

```
  name : string [20];           {--- שם התלמיד ---}
  grd1, grd2, grd3 : integer;   {--- שלושת הציונים ---}
  avg : real;                   {--- ממוצע ---}
  i, count : integer;
```

```
begin
```

```
  count := 0;
```

```
  for i := 1 to 30 do
```

```
    begin
```

```
      write (i, ' → שם התלמיד ה- ');
```

```
      read (name);
```

```
      write (' → 3 ציוני ');
```

```
      read (grd1, grd2, grd3);
```

```
      avg := (grd1 + grd2 + grd3) / 3;
```

```
      if (grd3 > 85) then
```

```
        begin
```

```
          avg := avg + 5;
```

```
          if (avg > 100) then
```

```
            avg := 100;
```

```
          if (avg = 100) then
```

```
            inc (count) ;
```

```
        end;
```

```
      write (avg, ' = ציון ', name, ' = שם ');
```

```
    end;
```

```
  write ('100 קיבלו את הציון ', count);
```

```
end.
```

```
if (avg > 100) or (avg == 100) then
```

```
  begin
```

```
    avg := 100;
```

```
    inc (count) ;
```

```
  end;
```

לפינה 8:

```

{--- קלט: פרטי השכלה וותק של 100 מועמדים. ---}
{--- פלט: כמה מהמועמדים מתאימים לתפקיד בכיר. ---}
program candidates ;
var
  education : integer ;      {--- מספר שנות השכלה ---}
  years : integer ;         {--- מספר שנות ותק ---}
  i, count : integer ;

{--- טענת כניסה: שנות השכלה ומספר שנות ותק. ---}
{--- טענת יציאה: 0 - אם אינו מתאים, 1 - אם מתאים, 2 - אם מתאים לתפקיד בכיר. ---}
function suits (education, years: integer) : integer ;
var
  sel : integer;
begin
  sel := education * years;
  if (sel < 8) then
    suits := 0
  else
    if (sel >= 8) and (sel <= 15) then
      suits := 1
    else
      suits := 2 ;

begin
  count := 0;
  for i := 1 to 100 do
    begin
      write (' מספר שנות השכלה ומספר שנות ותק → ');
      read (education, years);
      if (suits (education, years) = 2) then
        inc (count) ;
    end;
  write (' מתאימים לתפקיד בכיר ', count, ' היו ');
end.

```

פרק גשאלה 9:

```
{--- קלט: מערך דו-ממדי (מטריצה) בגודל 15x18 ---}
{--- פלט: הודעה האם המערך "פרחוני". ---}
{--- מערך פרחוני הוא מערך שיש בו לפחות 5 "פרחים" ---}
```

```
program flower_matrix;
```

```
type
```

```
mat_type = array [1..15,1..18] of integer;
```

```
var
```

```
m : mat_type ;
```

```
i, j, count : integer;
```

```
{--- טענת כניסה: מטריצה m, שורה i ועמודה j ---}
```

```
{--- טענת יציאה: "אמת" אם יש פרח ו-"שקר" אחרת. ---}
```

```
{--- פרח = האיבר שבתא (i, j) שווה לסכום ארבעת האיברים שבארבע פינותיו. ---}
```

```
function is_flower (m : mat_type ; i, j : integer) : boolean ;
```

```
var
```

```
sum : integer;
```

```
begin
```

```
if (i = 1) or (i = 15) or (j = 1) or (j = 18) then
```

```
is_flower := false
```

```
else
```

```
begin
```

```
sum := m [i-1, j-1] + m [i-1, j+1] +
m [i+1, j-1] + m [i+1, j+1] ;
```

```
if sum = m[i, j] then
```

```
is_flower := true
```

```
end
```

```
else
```

```
is_flower := false ;
```

```
end;
```

```
begin
```

```
count := 0;
```

```
for i := 1 to 15 do (*)
```

```
for j := 1 to 18 do
```

```
count := count + flower (m, i, j) ;
```

```
if count >= 5 then
```

```
writeln (' מערך פרחוני ')
```

```
else
```

```
writeln (' מערך לא פרחוני ');
```

```
end.
```

(*) בפונקציה flower נבדק ש-i ו-j

אינם בקצוות.

ללא בדיקה זו, תהיינה הלולאות:

```
for i := 2 to 14 do
```

```
for j := 2 to 17 do ...
```


שאלה 10:

טבלת משתנים:

שם משתנה	טיפוס משתנה	תפקיד המשתנה
bead	char	צבע חרוז: 'G' = ירוק, 'Y' = צהוב, 'R' = אדום.
countG countY countR	integer	מונה חרוזים ירוקים, צהובים, אדומים או
colors	מערך בגודל 3 של שלמים	מערך מוני החרוזים
n	integer	מספר החרוזים בשרשרת
valid	boolean	true שרשרת תקינה, false שרשרת לא תקינה.
count1	integer	מספר מחרוזות
count2	integer	מספר מחרוזות פגומות

תת-בעיות:

קלט-שרשרת -

טענת כניסה: מספר החרוזים.

טענת יציאה: מספר החרוזים הירוקים, צהובים ואדומים שנקלטו.

```
procedure string_input (n : integer; var countG , countY , countR : integer);
```

```
procedure string_input (n : integer; var colors : arr_type);
```

או:

שרשרת-תקינה? (countG, countY, countR)

טענת כניסה: 3 מונים של צבעי החרוזים שנקלטו.

טענת יציאה: "אמת" אם כל המונים שווים, ו-"שקר" אחרת.

```
function valid_string (countG , countY , countR : integer) : boolean;
```

```
function valid_string (colors : arr_type) : boolean;
```

או:

```

{--- קלט: חרוזים בשרשרת. ---}
{--- פלט: מספר השרשראות שיוצרו וכמה מתוכם פגומות. ---}
program strings_of_beads ;
var
    countG, countY, countR : integer;
    count1, count2, n : integer;

procedure string_input (n : integer; var countG , countY , countR : integer);
var
    bead : char ;           {--- צבע החרוז ---}
    i : integer ;
begin
    for i := 1 to n do
        begin
            write (' צבע החרוז: ');
            read (bead);
            if (bead = 'G') then inc(countG)
            else if (bead = 'Y') then inc (countY )
            else inc(countR);
        end;
    end;
end;

function valid_string (countG , countY , countR : integer) : boolean;
begin
    if (countG = countY) and (countG = countR) then
        valid_string := true
    else valid_string := false;
end;

begin
    countG := 0; countY := 0; countR := 0;
    count1 := 0 ; count2 := 0;

    write (' מספר החרוזים בשרשרת: ');
    read (n);
    while (n <> 0) do
        begin
            string_input (n , countG , countY , countR);
            inc (count1) ;
            if not valid_string (countG , countY , countR) then inc (count2) ;
            write (' מספר החרוזים בשרשרת: ');
            read (n);
        end;
    writeln (' מחרוזות ', count1, ' היו ');
    writeln (' מתוכם ', count2, ' פגומות ');
end.

```

if not **valid_string** (...) then ...
: שקול להוראה

if **valid_string** (...) = false then ...

ב. פתרון עם מערך מונים.

```

{--- קלט: חרוזים בשרשרת. ---}
{--- פלט: מספר השרשראות שיוצרו וכמה מתוכם פגומות. ---}
program strings_of_beads ;
type arr_type = array [1..3] of integer ;
var colors : arr_type ;
    count1, count2, n, i : integer;

procedure string_input (n : integer; var colors : arr_type);
var bead : char ;           {--- צבע החרוז ---}
    i : integer ;
begin
    for i := 1 to n do
        begin
            write (' צבע החרוז: ');
            read (bead);
            if (bead = 'G') then inc(colors [1])
            else if (bead = 'Y') then inc (colors [2])
            else inc(colors [3]);
        end;
    end;

function valid_string (colors : arr_type) : boolean;
begin
    if (colors[1] = colors[2]) and (colors[1] = colors[3]) then
        valid_string := true;
    else valid_string := false;
end;

begin
    for i := 1 to 3 do
        colors[i] := 0;
    count1 := 0 ; count2 := 0;
    write (' מספר החרוזים בשרשרת: ');
    read (n);
    while (n <> 0) do
        begin
            string_input (n , colors);
            inc (count1) ;
            if not valid_string (colors) then inc (count2) ;
            write (' מספר החרוזים בשרשרת: ');
            read (n);
        end;
    writeln (' מחרוזות ', count1, ' היו ');
    writeln (' מתוכם ', count2, ' פגומות ');
end.

```

if not **valid_string** (...) then ...
שקול להוראה:

if **valid_string** (...) = false then ...