

מדעי המחשב - 2 יחידות לימוד  
פתרון בחינת הבגרות

פרק א

שאלה 1:

- א. יחזור 7
- ב. יחזור 0
- ג. טענת כניסה: 2 מספרים שלמים a ו-b. טענת יציאה: אם הראשון גדול יותר יוחזר הראשון, אחרת יוחזר ההפרש שבין שני המספרים.

שאלה 2:

פלט	num>i	num	i	pul
7				100
	T	10	1	90
	F	1	2	45
	F	3	3	15
	T	8	4	7

טבלת מעקב לקלט: 10, 1, 3, 8 →  
פלט האלגוריתם: 7

הערה: עמודת בדיקת התנאי אינה חייבת להיות בטבלה. הימצאותה עוזר לבצע את המעקב.  
T = True, F = False

שאלה 3:

נדרש קטע תבנית. שם המערך A:

```
int i, count_1 = 0, count_10 = 0;
for (i = 0; i < 28; i++)
    if (A[i] < 10)
        count_1 ++;
    else
        count_10 ++;
if (count_10 > count_1)
    printf (" דו ספרתיים ");
else
    printf (" חד-ספרתיים ");
```

ניתן לשאול על מנת החלוקה ב-10:

```
if (A[i] / 10 == 0)
    count_1 ++;
else
    count_10 ++;
```

```
if (counter > (28-counter))
    printf (" דו ספרתיים ");
else
    printf (" חד ספרתיים ");
```

ניתן להשתמש במונה אחד שיספור למשל רק את הדו ספרתיים, ולבדוק:

**שאלה 4:**

דרך אחת	דרך אחרת	
puts (st1);	printf ("%s", st1);	א.
strncpy (st2, st1, 4); writeln (st2);	for (i = 0 ; i < 4 ; i++) printf ("%c", st1[i]);	ב.
ptr = strchr (st1, 'Y'); (*) if (ptr != NULL) printf ("yes"); else printf ("no");  ניתן להחליף את השורה המסומנת ב- (*) בהוראה: ptr = strstr (st1, "Y");	found = 0 ; for (i = 0 ; i < 10 ; i++) if (st1[i] == 'Y') found = 1 ; if (found == 1 ) printf ("yes"); else printf ("no");	ג.

**שאלה 5:**

	0	1	2	3
0	5	3	7	4
1	0 2	1 2	4 2	9 2
2	7 8	9	8	8

t	s
0	2
1	
2	
3	
4	

## פרק ב

## שאלה 6:

```

sum = 0 ;
printf("(M/F)? מין המשתתף: ");
scanf("%c", &ch);
if (ch == 'M')
    for (i = 0 ; i < 10 ; i++)
    {
        printf(" ? %d הישג בתחרות → ", i);
        scanf("%d", &score); //--- הציון מספר שלם
        sum = sum + score ;
    }
else
    for (i = 0 ; i < 7 ; i++)
    {
        printf(" ? %d הישג בתחרות → ", i);
        scanf("%d", &score);
        sum = sum + score ;
    }
printf(" %d : הסופי הציון ", sum);

```

## הערות:

- (1) ניתן היה לחסוך בכתיבה, ולבצע:
- ```

if (ch == 'M')
    limit = 10 ;
else
    limit = 7 ;

```
- ולכתוב רק לולאה אחת שמתבצעת מ-0 ועד limit (לא כולל).
- (2) ניתן לפתור את השאלה באמצעות שימוש בפונקציה score המקבלת כפרמטר את מספר המשחקים (7 או 10) ומחזירה את הסכום. לולאת הקלט תתבצע בתוך הפונקציה.

## שאלה 7:

--- החלפת ערכי המערך בסכום המספרים המתחלקים ב-3 בין 1 לתוכן התא ---

```

#include <stdio.h>

#define N 19
typedef int arr_type [N];

int sum_mod_3 (int n);
void arr_kelet (arr_type arr);

void main ()
{
    arr_type change ;
    int i ;

    arr_kelet (change) ;
    for (i = 0 ; i < 19 ; i++)
        change [i] = sum_mod_3 (change[i]) ;
}

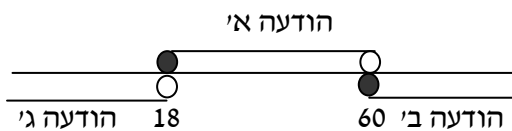
void arr_kelet (arr_type arr)
{
    קלט של מספרים חיוביים הקטנים מ-100 למערך הנתון
}

```

```
//--- טענת כניסה: מספר שלם וחיובי n הקטן מ-100
//--- טענת יציאה: מוחזר סכום כל המספרים בין 1 ל-n (לא כולל) המתחלקים ב-3 ללא שארית.
int sum_mod_3 (int n)
{
    int i, sum = 0 ;
    for (i = 0 ; i < n ; i++)
        if (i % 3 == 0)
            sum = sum + i ;
    return sum ;
}
```

**שאלה 8:**

א. תחום הערכים:



ב.

| נימוק            | a   | פלט     |
|------------------|-----|---------|
| $18 \leq a < 60$ | 50  | הודעה א |
| $60 \leq a$      | 100 | הודעה ב |
| $a < 18$         | 10  | הודעה ג |

```
if (a < 18 ) printf (" הודעה ג' ");
else if (a < 60) printf (" הודעה א' ");
else printf (" הודעה ב' ");
```

פרק גשאלה 9:

משתנים עיקריים:

| שם המשתנה | טיפוס המשתנה                          | תפקיד המשתנה                                                             |
|-----------|---------------------------------------|--------------------------------------------------------------------------|
| T         | מערך בגודל 100 של מספרים בתחום 0 - 50 | מערך הטיולים. מציין המערך = מספר הטיול. תוכן המערך = מספר הנרשמים לטיול. |
| dd        | מספרי שלם                             | יום. למעשה אין חשיבות ליום, כי מדובר בראשון בחודש.                       |
| mm        | מספרי שלם (בתחום 1 - 12)              | חודש.                                                                    |
| yy        | מספרי שלם                             | שנה                                                                      |

תת-משימות:

- הגדרת המערך ואתחולו (איפוס כל האיברים).
- בדיקת-תקינות-תאריך**: פעולה המקבלת חודש ושנה ומחזירה "אמת" אם התאריך מאוחר ל-1.8.2004 ו-"שקר" אחרת.
- טיולים-ללא-ביקוש**: פעולה המקבלת מערך טיולים ומדפיסה את כל הטיולים שמספר הנרשמים להם הוא 0.
- קלט-מספר-טיול**: פעולה המחזירה מספר טיול תקין (בתחום 1 עד 100).

//--- "שלם וסע" סוכנות נסיעות ---

#include &lt;stdio.h&gt;

#define TRUE 1

#define FALSE 0

#define N 100

typedef int arr\_type [N];

//--- פונקציות התכנית ---

int valid\_date (int month , int year) ;

void no\_demand\_trip (arr\_type T) ;

int get\_num () ;

```

void main ()
{
    arr_type T = {0} ;
    int trip_num, dd, mm, yy ;

    trip_num = get_num () ;
    while (trip_num != 999)
    {
        printf (" → הקש תאריך דרכון (יום, חודש, שנה) ");
        scanf ("%d %d %d", &dd, &mm, &yy) ;
        if (valid_date (mm, yy) && T[trip_num-1] < 50) T[trip_num-1] ++ ;
        trip_num = get_num () ;
    }

    no_demand_trip (T) ;
}

--- פעולה המקבלת חודש ושנה ומחזירה "אמת" אם התאריך מאוחר (או שווה) ל- 1.8.2004 ---
--- ו-"שקר" אחרת. הנחה: חודש ושנה תקינים.
int valid_date (int month , int year)
{
    if ((year > 2004) || (year = 2004 && month >= 8))
        return TRUE;
    return FALSE;
}

--- פעולה המקבלת מערך טיולים ומדפיסה את מספרי הטיולים שיש להם 0 נרשמים ---
void no_demand_trip (arr_type T)
{
    int i ;

    printf {" \n מספרי הטיולים שלא היו להם נרשמים : " } ;
    for (i = 0 ; i < N ; i++)
        if (T[i] == 0) printf (" %d", i+1) ;
    printf ("\n") ;
}

--- פעולה המחזירה מספר טיול תקין (בין 1 ל-N כולל) ---
int get_num ()
{
    int num ;

    do {
        printf (" → הקש מספר טיול (בין 1 ל- 100) ");
        scanf ("%d", &num) ;
    } while ( ( num < 1 || num > N) && num != 999) ;

    return num ;
}

```

מספר הטיול בין 1 ל-100 לכן יש להתייחס למספר התא במערך  $T$  כדי שתהיה התאמה.

ניסוח אלגנטי יותר:  $\text{return (year > 2004) || (year == 2004 \&\& (month \geq 8)) ;}$

## שאלה 10:

```

//--- "מטריצה רביעייה k" ---
#include <stdio.h>
#define TRUE 1
#define FALSE 0
typedef int mat_type [12][12];
void mat_kelet (mat_type M);
int square_k (mat_type M, int k);
void main ()
{
    mat_type M;
    int i = 30;
    mat_kelet (M);
    while (i > 0 && square_k (M, i) == FALSE)
        i--;
    if (i == 0)
        printf (" אין רביעייה k ");
    else
        printf (" נמצאה רביעייה (%d ערך מקסימאלי)", i);
}
void mat_kelet (mat_type M)
{
    המערך נתון. אין צורך במימוש הפעולה בבחינה.
}
//--- פעולה המקבלת מטריצה ומספר שלם (בתחום 1 עד 30) ---
//--- ומחזירה "אמת" אם קיימת רביעיית תאים (2x2) שערכם k ---
//--- ו-"שקר" אחרת. ---
int square_k (mat_type M, int k)
{
    int i, j, found = FALSE;
    for (i = 0; i < 11; i++) //--- i ≤ 10 ---
        for (j = 0; j < 11; j++) //--- j ≤ 10 ---
            if (M[i][j] == k)
                if ((M[i][j+1] == k) && (M[i+1][j] == k) && (M[i+1][j+1] == k))
                    found = TRUE;
    return found;
}

```

|        |          |
|--------|----------|
| i, j   | i, j+1   |
| i+1, j | i+1, j+1 |

ניתן לייעל את הפונקציה ולעצור את החיפוש כשנמצאת רביעיית k:

```
int square_k (mat_type M, int k)
{
    int i = 0, j, found = FALSE ;
    while ( i < 11 && ! found )           //--- 0 ≤ i ≤ 10 ---
    {
        j = 0 ;
        while ( j < 11 && ! found )       //--- 0 ≤ j ≤ 10 ---
        {
            if (M[i][j] == k)
                if ((M[i][j+1] == k) && (M[i+1][j] == k) && (M[i+1][j+1] == k))
                    found = TRUE ;
            j ++ ;
        }
        i ++ ;
    }
    return found ;
}
```