

1. כתוב פונקציה שמקבלת מערך של מספרים שלמים ואת גודלו. על הפונקציה לעדכן את המערך בהתאם לחוקיות הבאה:  
כל איבר במערך יש להחליף בסכום האיברים שנמצאים במערך המקורי מתחילת המערך ועד אליו (כולל הוא עצמו).  
לדוגמא, בהינתן המערך הבא:

7	5	-8	3	4	21	-10	-3	2	4
---	---	----	---	---	----	-----	----	---	---

הפונקציה תשנה אותו ל:

7	12	4	7	11	32	22	19	21	25
---	----	---	---	----	----	----	----	----	----

2. כתוב פונקציה שמקבלת כפרמטרים שני מספרים שלמים  $a$  ו- $b$  כאשר  $a < b$ . על הפונקציה לקלוט סדרת מספרים שלמים שמסתיימת בזקיף 1. (הזקיף לא שייך לסדרה). הפונקציה מחשבת ומחזירה את ההפרש בין שני הערכים שמוגדרים להלן:  
הערך הראשון הוא סכום המספרים הנמצאים בתוך הקטע  $((a + b)/2, b)$ ,  
הערך השני הוא סכום המספרים הנמצאים בקטע  $(a, (a + b)/2)$ .  
לדוגמא:  
עבור  $a = -3, b = 13$  והסדרה:

6, 25, -7, 5, -10, -3, 8, 3, 13, -1

הערך הראשון הוא :  $6+8$

הערך השני הוא : 3

והפונקציה תחזיר 11.

הערה: יש לבדוק כי הערכים  $a$  ו- $b$  שנקלטים, אכן מקיימים  $(a < b)$  לפני הקריאה לפונקציה.

3. משולש פסקל הוא סידור (אינסופי) של מספרים בצורת משולש, הנבנה באופן הבא: הקודקוד העליון של משולש פסקל מכיל את המספר 1. כל מספר אחר במשולש, אשר נמצא בשורה  $n$  במקום  $k$  (כאשר המספור מתחיל ב-0) שווה ל-

$$C(n, k) = \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

דוגמא (עבור משולש פסקל עד שורה 4):

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
```

כתוב פונקציה המקבלת מספר  $n$  ובונה משולש פסקל עד שורה  $n$  (כאשר המספור מתחיל ב-0).

ביישום עליך להגדיר פונקצית עזר `void print_space(int i, int n)` המדפיסה רווחים בתחילת שורה  $i$ ,

ופונקציית עזר יעילה (ללא שימוש בפונקציות מתמטיות כמו עצרת וכו')

`void print_pascal_line(int i)`

אשר מדפיסה את שורה  $i$  של משולש פסקל.

עליך לדאוג שהפלט (סידור השורות זו מעל זו) יראה כנדרש (כמו בציור למעלה).  
לצורך ציור אפשר להניח ש- $n < 5$ .

4. זוג מספרים שונים הם מאורסים (Betrothed numbers) אם כל אחד מהם שווה לסכום מחלקיו של האחר פחות 1 (כאשר בין המחלקים סופרים את המחלק 1, אבל אין סופרים את המספר עצמו - proper divisors).

לדוגמא: (48,75) - זוג של מספרים מאורסים :

- מחלקי המספר 48 הם 1, 2, 3, 4, 6, 8, 12, 16, 24, וסכומם הוא 76.
- מחלקי המספר 75 הם 1, 3, 5, 15 ו-25, וסכומם הוא 49.

אפשר להגיד שזוג מספרים הם מאורסים אם כל אחד מהם שווה לסכום מחלקיו של האחר כאשר בין מחלקיו לא סופרים לא את המחלק 1 ולא את המספר עצמו.

כתוב פונקציה יעילה שמקבלת מספר שלם  $n$  ומדפיסה את כל הזוגות המאורסים שלפחות אחד מהם קטן מ- $n$ .

5. שיטת ניוטון-רפסון הינה שיטה רקורסיבית למציאת שורשים של פונקציה. שיטה זו יעילה במיוחד עבור מגוון רחב של פונקציות אי-רציונליות ומאפשרת גם לאדם מן היישוב להעריך את פתרונה, כלומר היא מאפשרת לו למצוא ערך של  $x$  שבו הפונקציה מתאפסת. בהינתן פונקציה כלשהי  $f(x)$  מחשבים סדרת ערכים שמתחילה ב- $x_0 = 1$  והולכת ומתקרבת אל ערך אמיתי של  $x$  שבו הפונקציה מתאפסת. כלומר מתחילים עם ערך התחלתי  $x_0 = 1$ , ולאחריו מחשבים סדרת ערכים  $x_1, x_2, \dots, x_m$  כאשר  $x_m$  הוא תוצאה "מספיק טובה".

נשתמש בשיטה זו בכדי למצוא את  $\sqrt{n}$ . (כלומר, נמצא שורש של הפונקציה  $f(x) = x^2 - n$ )

עבור המקרה המסויים של חישוב  $\sqrt{n}$ , סדרת הערכים  $x_0, x_1, x_2, \dots, x_m$  תחושב באופן הבא:

$$x_0 = 1$$

$$x_{i+1} = x_i - \frac{(x_i^2 - n)}{2x_i}$$

נאמר שהערך שחישבנו עבור  $x_m$  הוא "תוצאה מספיק טובה" אם  $|x_m - x_{m-1}| \leq 0.00001$

כתוב פונקציה `double Newton-Raphson (double x, double n)` אשר משתמשת בנוסחה המתוארת לעיל בכדי לחשב ערך "טוב מספיק" של  $\sqrt{n}$ .  
על הפונקציה שלכם להיות רקורסיבית.  
לצורך חישוב של ערך מוחלט השתמשו בפונקציה `fabs()` הנמצאת בספריה `math.h`.

**יש לאחד את כל הפונקציות הנ"ל בתוכנית אחת על ידי `switch`:**

כגון:

```
int main()
{
    int select=0, i, all_Ex_in_loop=0;
    printf("Run menu once or cyclically?\n(Once - enter 0, cyclically - enter\nother number) ");
    if (scanf("%d", &all_Ex_in_loop) == 1)
        do
        {
            for (i=1; i<=5; i++)
                printf("Ex%d-->%d\n", i, i);
            printf("EXIT-->0\n");
            do {
                select=0;
                printf("please select 0-5 : ");
                scanf("%d", &select);
            } while ((select<0) || (select>5));
            switch (select) {
            case 1: Ex1(); break;
            case 2: Ex2(); break;
            case 3: Ex3(); break;
            case 4: Ex4(); break;
            case 5: Ex5(); break;
            }
        } while (all_Ex_in_loop && select);
    return 0;
}
```

הפונקציות `Ex1()`, ..., `Ex5()` בקוד משמשות להפעלת פונקציות השאלות 1-5 בהתאמה ומטפלות בקליטת הפרמטרים ובהדפסת התוצאות.

**בהצלחה!**