

מבוא למדעי המחשב

תרגיל מסכם

הנחיות:

- הגשה בבודדים או זוגות.
- קראו את השאלות בעיון לפני שתתחילו בפתרון.
- הקפידו לתעד את הקוד שלכם בהערות באנגלית.
- בתרגיל זה מותר להשתמש בפונקציות מהספריות `stdio.h`, `stdlib.h`, `math.h`, `stdbool.h` בלבד, שנלמדו בהרצאות.
- שימו לב: ציון נקבע לא רק על סמך נכונות של הקוד אלא גם על סמך צורתו (הגדרת משתנים נכונים, שימוש ב-`const`, קבועים, קריאות וכו').

מיון

כתבו תוכנית המקבלת מהמשתמש מספר שלם n , לאחר מכן מקבלת מהמשתמש מטריצה בגודל $n \times n$ וממיינת את שורות המטריצה באופן הבא:

- בשורות הזוגיות המספרים ימוינו מהקטן לגדול
- בשורות האי זוגיות המספרים ימוינו מהגדול לקטן

לאחר המיון הנ"ל התוכנית תדפיס את הפלט למסך כך שבין כל שני מספרים רווח בודד.

שימו לב שנמספר את השורות בצורה כזו שהשורה הראשונה מספרה יהיה 0, השורה השנייה מספרה 1 וכן הלאה.

חשוב: אין לכתוב 2 פונקציות מיון שונות. חשבו על דרך לפתור את הבעיה בלי לשכפל את הקוד

דוגמת הרצה:

קלט:

```
4
1 6 3 2
8 1 9 7
0 3 4 1
1 2 3 4
```

פלט:

```
1 2 3 6
9 8 7 1
0 1 3 4
4 3 2 1
```

הפעולה סופר-עצרת שנשמנה במ!! מוגדרת באופן הבא:

$$N!! = \prod_{i=0}^N i! = N! \cdot (N-1)! \cdot (N-2)! \cdot \dots \cdot 1! \cdot 0!$$

מכיוון שפעולת העצרת מוגדרת רק עבור מספרים שלמים וחייביים כמובן שפעולת הסופר עצרת מוגדרת רק עבור מספרים שלמים וחייביים.

כתבו פונקציה רקורסיבית אשר מקבלת בקלט את המספר N , מחשבת ומחזירה את $N!!$.

דוגמאות:

$$\text{Super_factorial1}(4)=288$$

$$\text{Super_factorial1}(6) = 24883200$$

את פעולת הסופר-עצרת ניתן לחשב גם בצורה הבאה:

$$N!! = \prod_{i=1}^N i^{N-i+1} = N \cdot (N-1)^2 \cdot (N-2)^3 \cdot \dots \cdot 1^N$$

כתבו פונקציה רקורסיבית אשר מקבלת בקלט את המספר N , מחשבת ומחזירה את $N!!$.

דוגמאות:

$$\text{Super_factorial2}(4)=288$$

$$\text{Super_factorial2}(6) = 24883200$$

כתבו תוכנית אשר מקבלת שני מספרים a ו- b כאשר a מספר שלם ואי שלילי, ו- b שלם וגם $2 \geq b \geq 1$ (כלומר b שווה 1 או 2), ומחשבת את $a!!$ בצורה הבאה:

אם $1=b$ אזי $a!!$ יחושב באמצעות הפונקציה מסעיף ג'.

אם $2=b$ אזי $a!!$ יחושב באמצעות הפונקציה מסעיף ד'.

אתם יכולים להניח כי הקלט תקין (כלומר a שלם ואי שלילי ו- b שווה 1 או 2)

דוגמאות הרצה:

קלט:

פלט:

288

קלט:

6 2

פלט:

24883200

Backtracking

ניתן לייצג לוח סודוקו שגודל שורותיו ועמודותיו $N \times N$ על ידי המטריצה $a[N][N]$, כאשר $a[i][j]$ מכיל מספר בין 1 ל- N אם התא תפוס על ידי המספר הזה ו-0 אם התא ריק.

כתבו תוכנית המקבלת מהמשתמש את גודל לוח הסודוקו, ולאחריו את לוח הסודוקו (אתם יכולים להניח כי הקלט תקין – כלומר כל המספרים בלוח הם בין 1 ל- N), לאחר קבלת הקלט התוכנית תחשב פתרון לסודוקו.

חוקי המשחק:

- יש למלא את הלוח במספרים בין 1 ל- N
- כל מספר מופיע פעם אחת בכל שורה, אסור שיופיעו שני מספרים זהים באותה שורה
- כל מספר מופיע פעם אחת בעמודה, אסור שיופיעו שני מספרים זהים באותה עמודה
- שימו לב: לצורך התרגיל, אין הגבלה על המצאות מספרים זהים בתתי ריבועים

הוראות לגבי המימוש:

- המימוש צריך להיות רקורסיבי, ולעשות שימוש ב-*Back Tracking*
- אם הלוח פתיר פלט התוכנית יהיה הלוח.
- אם הלוח אינו פתיר פלט התוכנית יהיה *No Solution*!
- התוכנית צריכה להיות יעילה מספיק כדי לסיים תוך מספר שניות את הדוגמאות שנשפק.

דוגמאות הרצה:

קלט:

```
9
710000609
200003000
000150008
007000090
006000700
020000400
100029000
000300004
905000086
```

פלט:

```
712438659
291763845
364152978
457816392
586941723
823597461
148629537
679385214
935274186
```

קלט:

9
710000619
200003000
000150008
007000090
006000700
020000400
100029000
000300004
905000086

פלט (שימו לב כי בשורה הראשונה יש שני אחדים):

No Solution!