מבוא למדעי המחשב מ'/ח' (234114/7)

תרגיל בית 2

מועד אחרון להגשה: 9.5.2022

המתרגל האחראי על תרגיל זה: **מג'ד ח'ורי**

עדכונים כדרך ה Piazza -בלבד!!

E-mail: majd-khoury@campus.technion.ac.il (במקרים מנהלתיים חריגים)

שעת קבלה רגילה: יום ב' 10:30-11:30

שעות קבלה מיוחדת לשאלות על התרגיל יפורסמו בהמשך באתר הקורס.

הנחיות:

- הגשה ב**בודדים**. עליכם לכתוב את הפתרונות לבד ולהגיש ביחידים.
- קראו את השאלות בעיון לפני שתתחילו בפתרונן.
- הקפידו **לתעד** את הקוד שלכם בהערות באנגלית.
- מלבד מילואים, לא יתקבלו תרגילים אחרי מועד הגשה. הגשה באיחור לאחר מועד הגשה נחשבת כאי-הגשה.
- כל יום מילואים = יום דחייה. על מנת לקבל את הדחייה, עליכם לשלוח באי-מייל למתרגל האחראי על תרגיל זה עותק של האישור המראה שהייתם במילואים (טופס 3010). אם האישור יגיע אליכם בתאריך מאוחר, יש להודיע על כך למתרגל האחראי על התרגיל.
- **לא ניתן לערער על תוצאות הבדיקה האוטומטית.**
- **שימו לב! הבדיקה הינה אוטומטית, ולכן הקפידו להדפיס בדיוק בפורמט שהתבקשתם ובדקו עם DiffMerge את הפלט שלכם מול הפלט של הדוגמאות שקיבלתם.**
  - השתמשו ב-redirection כדי להפנות את הפלט לקובץ טקסט.
  - וודאו את האותיות הגדולות והקטנות לפי הדוגמאות וההסברים בתרגיל.
  - הורדת שורה **אחת** בסוף כל שורה שהודפסה, אפילו אם היא האחרונה בתוכנית.
  - אין להדפיס רווחים שלא התבקשתם להדפיס (בתחילת שורה או בסופה).
- בתרגיל זה מותר להשתמש בפונקציות מהספריות stdio.h, stdbool.h, אלא אם כן נאמר אחרת. החומר הנדרש לתרגיל זה שייך להרצאות 1-5 ולתרגולים 1-5. אין להשתמש בחומר שאינו מופיע במצגות אלה.
- ההגשה הינה אלקטרונית ו**בבודדים** דרך אתר הקורס. קובץ ההגשה יהיה מסוג **zip** (ולא אף פורמט אחר) ויכיל בתוכו את הקבצים הבאים בלבד, ללא כל תיקיות:
  - קובץ **students.txt** עם שמך **באנגלית**, מספר תעודת הזהות וכתובת האי-מייל שלך.
  - קובץ פתרון **hw2q1.c** עבור שאלה 1.
  - קובץ פתרון **hw2q2.c** עבור שאלה 2.
  - קובץ פתרון **hw2q3.c** עבור שאלה 3.
- **חובה לשמור את קוד אישור ההגשה שמקבלים מהמערכת לאחר שמגישים, עד לסיום הקורס.**
- יש להקפיד להגיש את כל הקבצים בדיוק עם השמות שמופיעים לעיל. הגשה שלא תעמוד בתנאי זה **לא תתקבל ע"י המערכת!** אם המערכת לא מקבלת את התרגיל שלכם, חפשו את הפתרון לבעיה באתר הקורס תחת הכפתור FAQ.

בדיקה ידנית:

בנוסף לבדיקה האוטומטית, התרגיל ייבדק גם בבדיקה ידנית. הבדיקה תתמקד בנושאים הבאים:

- אורך כל פונקציה לא יעלה על 16 שורות קוד. הגבלה זו תקפה לכל הפונקציות, כולל main.

  רק שורות ריקות, שורות עם סוגר מסולסל בלבד, ושורות עם הערות בלבד לא נספרות.

  אסור לכתוב כמה פקודות שונות משמעותית באותה השורה, למשל כותרת תנאי\לולאה צריכה להיות בשורה נפרדת מגוף התנאי\לולאה.
- רוחב שורה - רוחב כל שורה (**כולל הערות והזחות**) לא יעלה על 75 תווים. ניתן לסמן אורך של שורה בקודבלוקס (ראו מסמך מצורף).
- קבועים בקוד:
  - יש להגדיר בעזרת #define כל קבוע משמעותי. שמות קבועים צריכים להיות באותיות גדולות בלבד, אם השם מכיל יותר ממילה אחת מילים יופרדו בעזרת מקף תחתון (למשל STUDENTS_NUM).
  - אין להשתמש בערכי ASCII ישירות. יש להשתמש בייצוג של התווים (למשל 'a').
- הזחות:
  - שיטת הזחה מקובלת – הזחת קוד בכל בלוק, למשל:

```
int main()
{
	// your code here
	while (…)
	{
		// your code here
	}
}
```

- אין להשתמש במשתנים גלובאליים או סטטיים.
- שמות משתנים/קבועים/פונקציות צריכים להיות אינפורמטיביים, להעיד על מטרתם.
- בהירות הקוד ותיעוד:
  - יש לתעד את הקוד באמצעות הערות באנגלית בלבד. במידה ויש כמה שורות קוד שניתן להסביר בקצרה מה המטרה שלהן, יש לשים הערה בהתחלה ואין צורך לתעד כל שורה.
  - יש לתעד פונקציות – לפני הפונקציה להוסיף הערה שמסבירה בקצרה מה הפונקציה עושה ומה המשמעות של הפרמטרים שלה (גם עבור הפונקציות שמוגדרות בתרגיל 1).
  - התיעוד צריך להיות אינפורמטיבי, כלומר יש להסביר מה המטרה של שורות הקוד ולא לכתוב את הקוד במילים.
- שכפול קוד שלא לצורך (למשל ריבוי קוד זהה במספר מקרי if-else שונים).
- אי-עמידה באחת מדרישות התרגיל (שימוש בחומר שהיה אסור בתרגיל וכו').

באופן כללי – הקפידו על כתיבת קוד מסודר ומובן ככל שניתן תוך יישום העקרונות שנלמדו בכיתה. מותר לכם לממש פונקציות עזר משלכם ולהשתמש בהן.

שאלה 1 – מבוא לפונקציות

1. מספר ברצף של מספרים שלמים נקרא מקסימום-מקומי אם הוא גדול או שווה מכל שכניו (מימין ומשמאל).
כתבו פונקציה שחתימתה:

```
int is_locally_max (int seq[N], int ind)
```

המקבלת רצף של מספרים שלמים seq ואינדקס ind, ומחזירה 1 אם המספר ב- seq[ind] הוא מקסימום-מקומי ו- 0 אחרת

2. מספר ברצף של מספרים שלמים נקרא זוגי-בריבוע אם המספר זוגי וגם המיקום שלו ברצף זוגי. מספר ברצף של מספרים שלמים נקרא אי-זוגי-בריבוע אם המספר אי-זוגי וגם המיקום שלו ברצף אי-זוגי.
מספר ברצף מספרים שלמים נקרא בעל זוגיות בריבוע אם המספר זוגי בריבוע או אי-זוגי בריבוע.

כתבו פונקציה שחתימתה:

```
int is_odd_even (int seq[N], int ind)
```

המקבלת רצף של מספרים שלמים seq ואינדקס ind, ומחזירה 1 אם המספר ב- seq[ind] בעל זוגיות בריבוע ו- 0 אחרת

3. דרגת המקסומום-זוגיות של רצף מוגדרת להיות מספר האיברים שהם מקסימליים-מקומיים או בעלי זוגיות בריבוע אך לא את שניהם.
כתבו תוכנית (main) הקולטת מהמשתמש איברים (מספרים שלמים) של רצף באורך 10 ומדפיסה את דרגת המקסימום-זוגיות של הרצף.

הערות למימוש:

- בסעיפים א' ו-ב' אין להשתמש בלולאות. בכל הסעיפים אין להשתמש במשפטי תנאי.
- אין צורך לבדוק את תקינות הקלט בשאלה זו, כלומר מובטח שהקלט מכיל 10 מספרים שלמים בטווח הייצוג של int.
- במימוש הפונקציות בסעיפים א'-ב' אפשר להניח שמתקיים $ind \geq 0$ וקטן מאורך הרצף.
- האינקדסים של המספרים ברצף מתחילים מ- 0.
- ההגדרה של שכן אינה ציקלית: איבר בתחילת הרצף אינו שכן של איבר בסוף הרצף.
- יש להיעזר בפונקציות מסעיפים א'-ב' בכתיבת התוכנית בסעיף ג'. חישוב המקסימליות-מקומית או זוגיות-ריבוע של איבר נעשת אך ורק בשימוש בפנוקציה אלה (ופונקציות הנקראות מתוכן, אם מימשתם כאלה).
- בשאלה זו יש להניח N=10 (ולהגדיר זאת ב-define).
- הוסיפו תיעוד לשתי הפונקציות בסעיפים א' ו- ב'.

דוגמאות:

```
Please enter a sequence:
0 1 2 3 4 5 6 7 8 9
Maximum-Parity degree: 9
```

```
Please enter a sequence:
0 9 1 8 2 7 3 6 4 5
Maximum-Parity degree: 5
```

שאלה 2 - היסטוגרמה

בשאלה זו תכתבו תוכנית אשר מחשבת את מספר המופעים של ספרה לפני ספרה אחרת ברצף של ספרות מ- 0 עד 9.

כתבו תוכנית הקולטת רצף של תווים עד סוף הקלט (EOF) ומדפיסה את הנתונים הבאים עבורן:

- התוכנית בהתחלה תדפיס את ההודעה "Enter a sequence of characters"
- התוכנית תיקבל רצף של תווים מהשתמש קלט עד קבלת EOF.
- התכנית תדפיס את מספר הספרות שהופיעו. "Number of digits: %d\n"
- התוכנית תדפיס את תת הרצף הארוך היותר של ספרות עוקבות.
  "Longest digit subsequence: %d\n"
- עבור כל ספרה, התוכנית תדפיס את השכיחות של הספרה, המוגדרת להיות כמספר המופעים של הספרה חלקי מספר כל הספרות המופיעות ברצף, עד קירוב שתי ספרות לאחר הנקודה העשרונית, בסדר מ- 0 עד 9 ומופרדים על ידי רווח יחיד. ראו הדוגמה למטה.
- הפונקציה תדפיס מטריצה בגודל $n \times n$, כאשר n הוא מספר הספרות השונות שהופיעו ברצף התווים. בשורה ה- i ועמודה ה- j במטריצה התוכנית תדפיס את מספר הפעמים שבה הספרה i הופיעה לפני הספרה ה- j ברצף התווים (לא בכרח עוקבות ברצף התווים). הפונקציה תדפיס מעל העמודות ומשמאל לשורות רק את הספרות שהופיעו ברצף התווים ומסדר עולה (מעל העמודות בסדר עולה מימין לשמאל, משמאל לשורות בסדר עולה מלמעלה למטה). על האיברים המודפסים בשורה אחת במטריצה מופרדים על ידי תו רווח יחיד.

דוגמאות:

```
Enter a sequence of characters:
012a”we228228
Number of digits: 9
Longest digit subsequence: 6
Frequencies: 0.11 0.11 0.56 0.00 0.00 0.00 0.00 0.00 0.22 0.00
Sequential:
  0 1 2 8
0 0 1 5 2
1 0 0 5 2
2 0 0 10 8
8 0 0 2 1
```

הסבר:
מספר הספרות הוא 9 (0 אחד, 1 אחד, חמשה 2, שני 8)
אורך הרצף האורך ביותר של ספרות עוקבות הוא 6 (אורך הרצף 228228)
שכיחות הספרה 0 היא $\frac{1}{9} = 0.11$, שכיחות הספרה 1 היא $\frac{1}{9} = 0.11$, שכיחות הספרה 2 היא $\frac{5}{9} = 0.56$, שכיחות הספרה 8 היא $\frac{2}{9} = 0.22$, שכיחות שאר הספרות הן 0.00.
הספרה 2 הופיעה 10 פעמים לפני ספרה 2 אחרת (הספרה 2 הראשונה הופיעה לפני ארבעה ספרות 2 אחרות, הספרה 2 השניה הופיעה לפני שלושה ספרות 2 אחרות וכו') ולכן ערך המטריצה בעמודה והשורה המתאימות לספרה 2 שווה ל- 10
הספרה 2 הופיעה לפני הספרה 8 שמונה פעמים (הספרה 2 הראשונה, השניה והשלישית הופיעו לפני שתי ספרות 8, הספרה 2 הרביעית והחמישית הופיעו לפני ספרה 8 אחת). ולכן ערך המטריצה בשורה המתאימה לספרה 2 והעמודה המתאימה לספרה 8 שווה ל- 8

```
Enter a sequence of characters:
01234567899876543210
Number of digits: 20
Longest digit subsequence: 20
Frequencies: 0.10 0.10 0.10 0.10 0.10 0.10 0.10 0.10 0.10 0.10
Sequential:
  0 1 2 3 4 5 6 7 8 9
0 1 2 2 2 2 2 2 2 2 2
1 2 1 2 2 2 2 2 2 2 2
2 2 2 1 2 2 2 2 2 2 2
3 2 2 2 1 2 2 2 2 2 2
4 2 2 2 2 1 2 2 2 2 2
5 2 2 2 2 2 1 2 2 2 2
6 2 2 2 2 2 2 1 2 2 2
7 2 2 2 2 2 2 2 1 2 2
8 2 2 2 2 2 2 2 2 1 2
9 2 2 2 2 2 2 2 2 2 1
```

הערות למימוש:

- אין צורך לבדוק את תקינות הקלט בשאלה זו.
- <u>הדרכה לפתרון:</u> ניתן לשמור ולעדכן בכל שלב היסטוגרמה (מערך) המציינת מספר המופעים של כל ספרה עד שלב זה, איך צריך לעדכן את מטריצת Sequential בעת קבלת ספרה חדשה מהקלט?

שאלה 3 – מערכים דו מימדיים

בשאלה זו נממש גרסה שונה של **תקן הצפנה מתקדם (AES – Advanced Encryption Standard)**.

AES הוא צופן בלוקים סמטריים המשומש להצפנת הודעות לצורך האבטחה נגד התקפות. הצופן מקבל הודעה ומפתח להצפנה ומחשב הודעה חדשה מוצפנת שניתנת לפענוח רק אם מפתח ההצפנה ידוע.

בשאלה זו נייצג את ההודעה כמערך דו-מימדי המורכב ממספרים שלמים בין 0-15. גודל ההודעה המקסימלי הוא $M \times N$.

אלגוריתם AES מחלק את ההודעה לבלוקים בגודל 4×4 ויכול להפעיל על כל בלוק בנפרד אחד מארבע הפעולות הבסיסיות הבאות:

1. מיפוי בתים: כל ערך בבלוק ממופה לערך אחר בין 0 ל- 15 באופן חח"ע ועל, נייצג את המיפוי של הערכים בעזרת מערך חד-מימדי בגודל 16 הידוע מראש ונתון לכם בקובץ המצורף לתרגיל הבית (MAX_BYTE_VALUE).
2. הזזת שורות: נסמן את שורות של הבלוק מ- 0 עד 3 (שורה ראשונה - 0, שורה אחרונה - 3), כל שורה i תוזז i פעמים שמאלה ובאופן ציקלי. (ראו דוגמאות)
3. ערבוב עמודות: בתרגיל זה נדלג על שלב ערבוב העמודות.
4. הוספת מפתח הצפנה: מבוצע ב- AES המקורי בעזרת פעולת bitwise-XOR בין ההודעה והמפתח, בגרסה של AES שנממש בתרגיל, נממש את הפעולה בעזרת חיבור מודלו 16 (מודלו MAX_BYTE_VALUE) עם המפתח, כלומר לכל איבר בבלוק נבצע חיבור מודלו 16 עם איבר מהמפתח. החיבור מתבצע איבר-איבר באופן סדרתי. המעבר על הבלוק יהיה משמאל לימין, ומשורה הראשונה עד השורה האחרונה. עבור המפתח, מבצעים החיבור באופן סדרתי ובאופן ציקלי (כלומר מתחילים מאיבר הראשון עד איבר האחרון, וחוזרים לאיבר הראשון עוד פעם עד הוספת המפתח לכל איברי הבלוק).

דוגמה למיפוי בתים:

עבור מערך המיפוי הבא:

10	1	2	4	7	9	14	12	3	5	6	13	15	11	8	0

ובלוק הבא:

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

לפני מיפוי

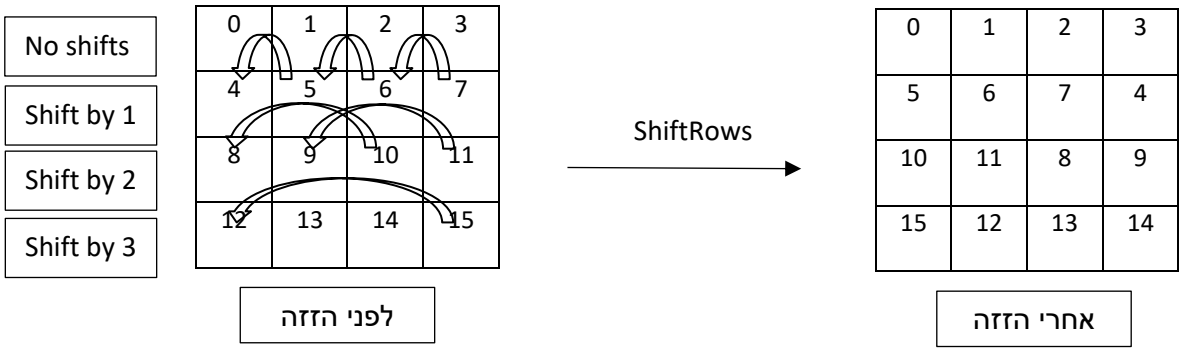
subBytes →

10	1	2	4
7	9	14	12
3	5	6	13
15	11	8	0

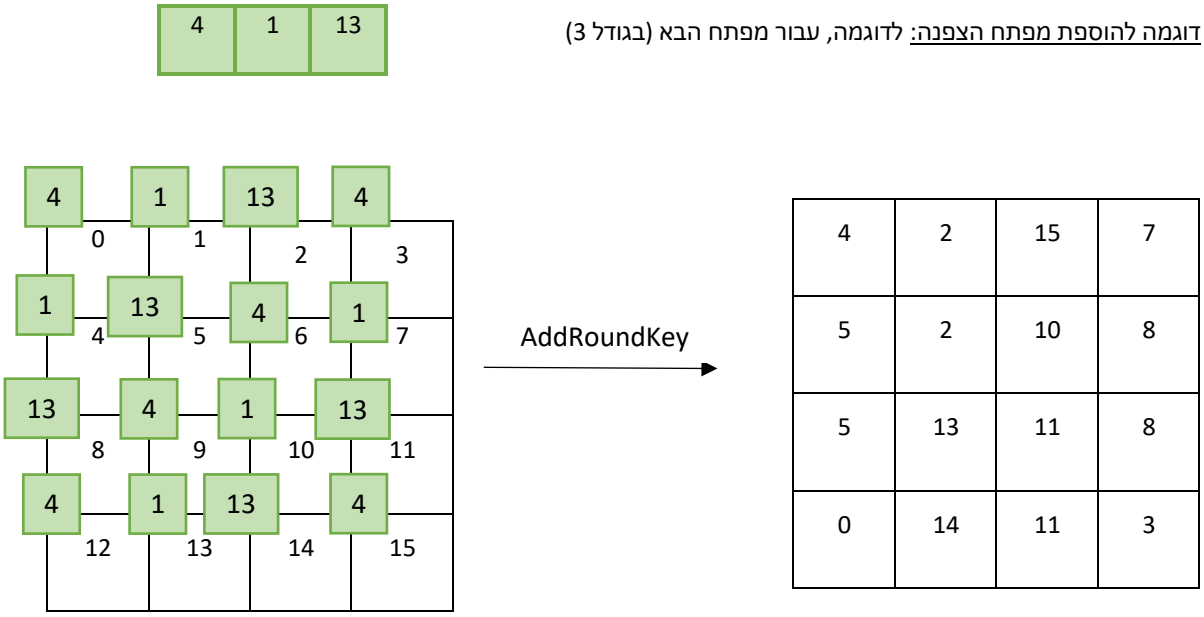
אחרי מיפוי

לשים לב שכל ערך בין 0 ל- 15 מופיע רק פעם אחת במערך המיפוי. כל איבר בבלוק מהווה בעצם את האינדקס של הערך לאחר המיפוי, למשל, הערך בפינה העליונה משמאל לפני המיפוי (0) מהווה האינדקס של הערך בפינה העליונה משמאל (10) במערך המיפוי.

דוגמה להזזת שורות:



דוגמה להוספת מפתח הצפנה: לדוגמה, עבור מפתח הבא (בגודל 3)



כאשר ליד כל איבר בבלוק כתוב משמאל ומעל האיבר עם איזה איבר במפתח התבצעה פעולת החיבור מודלו 16, מתחילים מאינדקס העליון השמאלי של הבלוק ומבצעים חיבור איבר-איבר (מודלו 16) עד שעוברים על כל מפתח. לאחר מכן ממשיכים מתחילת המפתח.

לדוגמה, עבור האיבר האחרון בבלוק נקבל: $15 + 4 = 19$ ושארית החלוקה שלו ב- 16 היא 3.

אלגוריתם AES:

האלגוריתם פועל **בסיבובים** (10 או 12 או 14 אך בתרגיל נפעיל רק 10 סיבובים), בסיבוב הראשון, הצופן מבצע רק את הוספת מפתח הצפנה (פעולה בסיסית מספר 4), בשאר הסיבובים, הצופן במבצע כל ארבע הפעולות הבסיסיות בסדר שהוצגו למעלה (מיפוי בתים, הזזת שורות, ערבוב עמודות והוספת מפתח הצפנה). בכל אחד מהסיבובים, הפעולות מתבצעות על בלוקים בגודל 4×4, כלומר האלגוריתם מחלק את ההודעה לבלוקים בגודל 4×4 והפעולות מופעלות על כל אחד מהבלוקים.

כתבו תוכנית העובדת באופן הבא:

- התוכנית תיבקש מהשתמש להכניס גודל המפתח.
  "Enter size of key:\n"
- לאחר שהמשתמש מכניס גודל המפתח, התוכנית תיבקש מהשתמש להכניס את המפתח "Enter the key:\n"

- לאחר שהמשתמש מכניס כל המפתח, התוכנית תיבקש מהשתמש להכניס את מספר העמודות של ההודעה, מספר השורות של ההודעה יהיה תמיד שווה ל- $\boldsymbol{M = 4}$.

"Enter the number of rows and columns of the message:\n"

- לאחר שהמשתמש מכניס מספר השורות והעמודות, התוכנית תיבקש מהשתמש להכניס את ההודעה שרוצה להצפין.

"Enter the message you want to encrypt\n"

- לאחר שהמשתמש מכניס את ההודעה, התוכנית תפעיל **10** סיבובים של אלגוריתם AES ותדפיס את ההודעה:

"Encrypted message:\n"

ולאחר מכן תדפיס את ההודעה המוצפנת על ידי צופן AES.

<u>דוגמאות:</u>

```
Enter size of key:
3
Enter the key:
4 1 13
Enter the number of columns of the message:
8
Enter the message you want to encrypt:
0 1 2 3 4 5 6 7
4 5 6 7 8 9 10 11
8 9 10 11 12 13 14 15
12 13 14 15 0 1 2 3
Encrypted message:
5 14 8 12 14 12 14 13
6 3 11 10 15 1 12 9
3 12 0 14 4 11 15 15
11 6 6 7 9 9 14 12
```

```
Enter size of key:
1
Enter the key:
0
Enter the number of columns of the message:
4
Enter the message you want to encrypt:
0 1 2 3
4 5 6 7
8 9 10 11
12 13 14 15
Encrypted message:
15 1 2 8
9 10 4 3
0 13 14 5
12 7 11 6
```

הערות למימוש:

- חוץ ממספר הסיבובים, ניתן להניח שהקלט תקין. כלומר המשתמש מכניס רק מספרים שלמים. ומספר העמודות c מקיים $0 < c \leq N$. ושמספר העמודות מתחלק ב- 4. ניתן גם להניח שהערכים המוכנסים הם בין 0 ל- 15. (כולל)
- ניתן להניח שגודל המפתח חיובי. (גדול ממש מ- 0)
- ניתן להניח שהשתמש מכניס מספר איברים המתאים לגודל המפתח וההודעה.
- שימו לב לקבועי התוכנית שהוגדרו בשאלה, צריך להגדיר אותם ב- define.
- ניתן להניח ש- N, M מוגדרים ב- define. (נתונים בקובץ המצורף לתרגיל בית)
- מומלץ ורצוי לחלק את התוכנית לפונקציות, למשל, ניתן לחלק כל פעולה בסיסית (מיפוי, הזזת שורות, ערבוב עמודות, הוספת מפתח הצפנה) לפונקציה אחת שמבצעת את הפעולה.
- בשאלה זו, **כל המערכים שמגדירים חייבים להיות בגודל קבוע**, כלומר עליכם להגדיר מערך בגודל מקסימלי ולהשתמש בחלק ממנו.
- מצורף לתרגיל קובץ עזר המכיל פונקציות שימושיות, ניתן לשנות פונקציות אלה כרצונכם. מצורף בנוסף דוגמה לשימוש בפונקציות אלה.
- מצורף בקובץ מערך המיפוי שישומש לתקן ההצפנה, אל תשנו את הערכים במערך.

דגשים נוספים:

יש להיעזר באתר הבדיקה האוטומטית http://csm.cs.technion.ac.il/~cs234114/ על-מנת לבדוק את הקוד שלכם. האתר מאפשר לכם לשלוח את הקוד שלכם לשאלה מסוימת (קובץ c.) ולבדוק האם הוא עובר בדיקות מסוימות בריצה על הבודק האוטומטי. התוצאה לכל אחת מהבדיקות יכולה להיות אחת משלוש:

א. "עבר"- הבדיקה עברה בהצלחה!

ב. "נכשל"- הפלט עבור הבדיקה לא יצא זהה. במקרה כזה יש להפעיל את התוכנית באמצעות redirection כפי שנלמד בתרגיל בית 0 ולמצוא באמצעות DiffMerge את ההבדלים (את הקלט והפלט המצופה לכל הבדיקות תוכלו למצוא באתר הקורס)

ג. "נתקע"- התכנית נתקעה בלולאה אינסופית או שהיא ממתינה לקלט (יש לחכות 30 שניות עד לקבלת התשובה).

במידה ותהיה בקוד שלכם שגיאת קומפילציה כל הבדיקות יקבלו תוצאת "נכשל" והשגיאה עצמה תהיה רשומה במפורש.

שימו לב: מעבר הבדיקות שבאתר לא מהווה הבטחה לכך שתעברו את כל הבדיקות של הבודק האוטומטי!
האתר מריץ את הקוד שלכם רק על מספר בדיקות מצומצם, בבדיקה האוטומטית הקוד יורץ על בדיקות אלו ומס' בדיקות נוספות. **לכן- כתבו בדיקות משלכם על-מנת לוודא כי הקוד שלכם נותן את הפלט המצופה בכמה שיותר מקרים!**
כאמור, באתר הקורס מסופקים לכם קבצי קלט ופלט מצופה עבור הבדיקות, על-מנת שתוכלו להשתמש בהם לביצוע DiffMerge במקרה שהאתר אומר שאתם לא עוברים בדיקה מסוימת. **פתחו אותם וודאו שאתם מבינים מדוע הפלט הוא הפלט הנכון עבור אותו קלט.**
שאלות ותשובות נפוצות בנוגע לתרגיל יתפרסמו באתר כל כמה זמן תחת סעיף F.A.Q- **חובה להיכנס ולהתעדכן מדי פעם! כל דגש שמפורסם שם הוא מחייב!**

בהצלחה !