

מבוא למדעי המחשב בשפת ++C - חוברת תרגילים

הגשת התרגילים היא ביחידים, יש להגיש במערכת ה- moodle את כל הפרוייקט כקובץ ZIP.

חובה להגיש תרגיל שמתקמפל + דוגמאות פלט של ההרצה.

יש להקפיד על הדברים הבאים:

- שמות משתנים ברורים
- שימוש בקבועים והגדרתם מעל ה- main
- קוד מרווח ומאוורר
- הפרדת חלקי השאלה השונים ע"י שורת רווח: הגדרת משתנים, קלט, חישוב ולבסוף הפלט.

תוכן עניינים

1. אבני הבניין של שפת ++C 2
2. ביטויים לוגיים ומשפטי תנאי 3
3. לולאות 5
4. מערכים ומטריצות 12
5. פונקציות 18
6. מחרוזות 21
7. מצביעים 23
8. אריתמטיקה של מצביעים 25
9. רקורסיות 27
10. מיונים וחיפושים 31
11. מבנים 34
12. הקצאות דינאמיות 34
13. הקצאות דינאמיות ומבנים 38

1. אבני הבניין של שפת ++C

הקפידו להשתמש בקבועים!

1. כתבו תוכנית הקולטת מהמשתמש רדיו של מעגל ומדפיסה את היקף ואת שטח המעגל.
2. בקפידה ניתן להזמין טוסט שמחירו 12 ש"ח.
ניתן לבקש להוסיף תוספות "זולות" (כגון פטריות, זיתים, תירס וכו'), ומחיר כל תוספת הינו 2 ש"ח נוספים.
ניתן לבקש להוסיף תוספות "יקרות" (כגון גבינה בולגרית, אקסטרה גבינה-צהובה וכו'), ומחיר כל תוספת הינו 3 ש"ח נוספים.
כמו כן, יתכנו סוגי תוספות זולות או יקרות נוספים שאינם בשאלה.

דוגמאות:

- עבור טוסט בתוספת זיתים יש לשלם 14 ש"ח
- עבור טוסט בתוספת זיתים ופטריות יש לשלם 16 ש"ח
- עבור טוסט בתוספת זיתים וגבינה בולגרית יש לשלם 17 ש"ח
- עבור טוסט עם 2 תוספות יקרות ותוספת זולה יש לשלם 20 ש"ח

החליטו מהם נתוני התוכנית וכן האם כל נתון הוא קבוע, קלט, חישוב או פלט.
כתבו תוכנית המבקשת מהמשתמש נתונים עבור הטוסט שברצונו להזמין והציגו את המחיר שעליו לשלם.

3. בחנות "רהיטים שווים" מספקים את שירות ההובלה של הרהיט בתשלום נוסף:
עבור כל ק"מ הובלה יש לשלם 5 ש"ח. עבור כל קומה יש לשלם 1 ש"ח נוסף בעבור כל ק"ג של הרהיט.
יש להציג ללקוח מהי הוצאתו הסופית בעבור הרכישה, כאשר ההוצאה כוללת את מחיר הרהיט, מחיר ההובלה ו- 10% טיפ ממחיר הרהיט למובילים.

דוגמא:

עבור ספה שעולה 5000 ש"ח ומשקלה 80 ק"ג, עבור לקוח שגר בקומה 3 במרחק 10 ק"מ מהחנות, סכום הוצאתו הוא:

$$5790 = 5000 + 5 \cdot 10 + 3 \cdot 80 + 5000 \cdot 0.1 \text{ ש"ח}$$

החליטו מהם נתוני התוכנית וכן האם כל נתון הוא קבוע, קלט, חישוב או פלט.
כתבו תוכנית המבקשת מהמשתמש נתונים עבור הרכישה והציגו את המחיר שעליו לשלם.

2. ביטויים לוגיים ומשפטי תנאי

יש לייצר solution ובו פרוייקט לכל אחת מהשאלות הבאות.
יש להקפיד בנוסף בדיקת מקרים זרים על קריטריון זהה ולא לשלב בין קריטריונים שונים. אם יש צורך אז בתוך if פנימי.

יש להקפיד על בדיקת מקרים זרים על קריטריון זהה ולא לשלב בין קריטריונים שונים. אם יש צורך אז בתוך if פנימי.

1. פלוני רוצה לקנות טוסטוס. הוא עובד ומרוויח כסף, אך יחד עם זאת יש לו הוצאות שוטפות.
כתוב תוכנית המבקשת מפלוני את הנתונים הדרושים ומציגה לו אם יוכל לקנות מתישהו את הטוסטוס, ואם כן, כמה חודשים יהיה עליו לעבוד עד אשר יוכל לקנות אותו.
 2. כתבו תוכנית הפותרת את המשוואה $Ax+B=0$ הבאה. עליכם לקלוט מהמשתמש את מקדמי המשוואה A ו- B , ולהציג את פתרון המשוואה: ערכו של X , אין פתרון או אינסוף פתרונות.
 3. כתוב תוכנית המחשבת כמה כסף יש לתת בצ'ק לחתונה עפ"י הקריטריונים הבאים:
 - אם החתן או הכלה מוגדרים כחברים קרובים סכום הבסיס הינו 500 ₪
 - אם החתן או הכלה הינם בני-משפחה הבסיס הוא 1000 ₪
 - בכל מקרה אחר סכום הבסיס הינו 250 ₪
 - אם ההיכרות עם אחד מבני הזוג היא מעל 3 שנים, יש לשים 50 ₪ נוספים, אלא אם אתה בן-משפחה
 - אם זמן הנסיעה לחתונה מעל שעה יש להוריד 50 ₪ מהסכום, אלא אם אתה בן-משפחה
- עליכם להחליט מהם הנתונים שיש לקלוט ולבסוף להציג את ההמלצה לגובה הצ'ק.
4. בספריה יש מנויים, כך שכל מנוי יכול להשאיל ספרים עפ"י ההגבלות הבאות:
מנוי מבוגר יכול להשאיל עד 5 ספרים, ומנוי ילד יכול להשאיל עד 3 ספרים. כאשר מנוי משאיל ספר, במקרה ויש לו בבית ספר שמוחזק מעל חודש, לא ניתן להשאיל לו ספר נוסף (ללא תלות במספר הספרים שעדיין יכול להשאיל).
עליכם לכתוב תוכנית אשר קולטת נתוני מנוי, ומציגה האם יכול להשאיל כעת ספר נוסף.
עליכם להחליט אילו נתונים יש לקלוט.
 5. על מנת להחליט האם סטודנט מתקבל ללימודים במכללת ליליפוט יש לבדוק האם הוא עונה לפחות על אחד מן הקריטריונים הבאים:
- מממוצע בגרות לפחות 102.

- ציון פסיכומטרי לפחות 700, וגם שהציון על החלק הכמותי הינו לפחות 145 וגם שהציון על חלק האנגלית הינו לפחות 120.

- שיקול ממוצע הבגרות עם ציון הפסיכומטרי הוא לפחות 600, כאשר נוסחאת השיקול הינה:

$$1.2 / (\text{ציון בגרות}) + 0.8 * (\text{ציון פסיכומטרי})$$

דוגמאות:

גוליבר קיבל פסיכומטרי 650 ויש לו ציון בגרות 95. מאחר ואינו מתקבל על בסיס אחד משני הקריטריונים הראשונים, נבדוק מהו ציונו המשוקלל: $650 * 0.8 + 95 / 1.2 = 599.1666$. מאחר וציונו המשוקלל של גוליבר קטן מ-600 הוא אינו מתקבל ללימודים.

לעומת זאת, עבור פסיכומטרי 650 וציון בגרות 97 היה גוליבר מתקבל ללימודים מאחר וציונו המשוקלל היה מעל 600: $650 * 0.8 + 97 / 1.2 = 600.8333$.

כתבו תוכנית הקולטת מהמשתמש את הנתונים הדרושים ומציגה הודעה האם הסטודנט יכול להתקבל ללימודים במכללת ליליפוט.

6. כתבו תוכנית המציגה הצעת אימון לאדם הרוצה להיכנס לכושר ריצה.

ההצעה מתבססת על כושרו של האדם, המבוסס על דופק המנוחה, ועל מספר השבועות שכבר התאמן.

להלן טבלה עם נתונים:

	דופק מנוחה עד 60	דופק מנוחה בין 60 ל-70	דופק מנוחה מעל 70
שבוע 1-2	3 ק"מ	3 ק"מ	3 ק"מ
שבוע 3-4	5 ק"מ	5 ק"מ	3 ק"מ
שבוע 5 והילך	10 ק"מ	8 ק"מ	3 ק"מ

יש להחליט מה הנתונים שיש לקלוט ולהציג לבסוף המלצה כמה ק"מ על האדם לרוץ באימון הקרוב.

7. כתבו תוכנית אשר מקבלת מהמשתמש 3 מספרים שלמים המיצגים: ציון בחינה (ציון מתוך מאה), ממוצע תרגילי בית (ציון מתוך מאה), ומספר תרגילים שהוגשו (עד 8 תרגילים). התוכנית תחשב את הציון הסופי על-פי הנוסחה הבאה:

- אם הוגשו עד 4 תרגילי בית כולל, הציון הסופי הוא 0 (נכשל).
- אם הוגשו 5 או 6 תרגילי בית: ממוצע תרגילי הבית יהווה 20% תקף בציון הסופי (ראו בהמשך הסבר מהו "ציון תקף") במידה והסטודנט עבר את הבחינה בציון 55 לפחות. במידה והסטודנט קיבל בבחינה ציון של 54 ומטה, הציון הסופי יהיה זהה לציון הבחינה.

- אם הוגשו 7 או 8 תרגילי בית:

○ אם ציון הבחינה הוא 54 ומטה:

- אם ממוצע תרגילי הבית הוא 80 ומעלה, אז תרגילי הבית מהווים 25% מגן.
- אם ממוצע תרגילי הבית הוא פחות מ-80, אז תרגילי הבית מהווים 20% מגן.
- הציון בבחינה נע בין 55 ל-100: תרגילי הבית מהווים 30% מגן.

הערה: כאשר כתוב תקף, הכוונה היא שהציון של תרגילי הבית ישוקלל בכל מקרה, גם אם הוא מוריד את הממוצע. כאשר כתוב מגן, הכוונה היא שהציון של תרגילי הבית ישוקלל **אם ורק אם הוא מעלה את הממוצע**.

התכנית שתכתבו צריכה לקרוא מהקלט 3 מספרים ולהדפיס את הציון הסופי בקורס לפי החישוב שתואר לעיל.

3. לולאות

כתוב תפריט המכיל את האפשרויות הבאות: (שימו לב שתפריט אמור לחזור על עצמו עד אשר המשתמש מבקש לצאת!):

טיפ: לעבור על דוגמאות יצירת מספרים ע"י הוספת ספרות משמאל / ימין מהמצגת "הרצאה: ניסוח ופתרון בעיות עם לולאות" שנמצאת תחת שבוע 3 ב-moodle.

1. קלוט מהמשתמש מספר וספרה, והצג כמה פעמים הספרה מופיעה במספר.
דוגמא: עבור המספק 123532 והספרה 3 יוצג הערך 2, מאחר והספרה 3 מופיעה פעמיים.
 2. יש לקלוט מהמשתמש מספר ולייצר מספר חדש המכיל רק את הספרות שבמיקומים הזוגיים. מיקומה של הספרה הימנית ביותר הוא 0.
דוגמא: עבור המספר 12345 יש לייצר את המספר 135.
 3. כתוב תוכנית הקולטת מהמשתמש 2 מספרים בעלי מספר ספרות זהה. יש להציג את מספר הספרות במיקומים זהים שבעלות ערך זהה.
דוגמא: עבור המספרים 12345 ו-12855 יוצג 3 כי הספרות 1, 2 ו-5 נמצאות באותם מיקומים בשני המספרים.
 4. ספרת הביקורת היא הספרה התשיעית, הימנית ביותר, של מספר הזהות והיא מחושבת על ידי אלגוריתם Luhn, שאותו פיתח איש מעבדות IBM בשנות ה-50, המשמש לחישוב ספרות ביקורת לא רק למספרי הזהות הישראליים.
- הנוסחה לחישוב ספרת הביקורת מורכבת מסדרת משקל זהה שניתנת לכל אחת משמונה הספרות שמשמאל בכל מספר זהות. סדרת המשקל שניתנת לכל אחת משמונה הספרות משמאל: 1 (לספרה הראשונה), 2, 1, 2, 1, 2, 1, 2.
- כל ספרה מוכפלת במשקל שניתן לה. בהמשך מחברים את כל הספרות שנוצרו מהכפלה. אם למשל בעקבות ההכפלה נוצר מספר דו ספרתי, שתי ספרותיו מחוברות זו לזו בפעולת חיבור הספרות. למשל, אם

הספרה השישית של מספר הזהות היא 8, המשקל שלה הוא 2 לפי סדרת המשקל הזהה. תוצאת ההכפלה היא 16. בפעולת החיבור של כל הספרות, מחברים בין היתר את 1 ו-6, ולא מוסיפים את 16 לכל החיבור.

את הסכום שמתקבל בחיבור כל הספרות שנוצרו בהכפלה משלימים לכפולה הקרובה של 10 כלפי מעלה. למשל, אם הסכום שהתקבל הוא 48, מעגלים ל-50. ההפרש שנדרש לצורך ההשלמה הוא ספרת הביקורת. במקרה זה 2.

חישוב לדוגמה (מספר זהות פיקטיבי): 78962134-9

$$7 \times 1 = 7, 8 \times 2 = 16, 9 \times 1 = 9, 6 \times 2 = 12, 2 \times 1 = 2, 1 \times 2 = 2, 3 \times 1 = 3, 4 \times 2 = 8$$

מחברים את כל הספרות שהתקבלו: $7+1+6+9+1+2+2+2+3+8$

הסכום שהתקבל: 41.

כדי להגיע לכפולה הקרובה של 10 מעגלים ל-50. ההפרש שנדרש להשלמה (ההפרש בין 50 ל-41, שהוא סכום החיבור) הוא 9. לכן 9 היא ספרת הביקורת.

מקור התיאור: <http://www.ynet.co.il/articles/0,7340,L-4483052,00.html>

קלוט מהמשתמש מספר תעודת זהות הכולל ספרת ביקורת (סה"כ 8 או 9 ספרות), והצג למשתמש הודעה האם המספר תקין עפ"י בדיקת האלגוריתם הנ"ל.

5. כתוב תוכנית הקולטת מהמשתמש מספר ומייצרת מספר חדש כך שכל זוג ספרות יוחלף באופן הבא:

- ספרת האחדות תהפוך להיות ספרת העשרות וספרת העשרות תהפוך להיות ספרת האחדות
- ספרת המאות תהפוך להיות ספרת האלפים וספרת האלפים תהפוך להיות ספרת המאות
- וכו'

במידה ומספר הספרות אי-זוגי, הספרה השמאלית תשאר במקומה.

דוגמאות:

עבור המספר 1234567 התוכנית תייצר ותציג את המספר 7654321

עבור המספר 9728 התוכנית תייצר ותציג את המספר 8279

6. יש לקלוט מהמשתמש מספר ולייצר מספר חדש כך שספרותיו הן כמו המספר המקורי, ומימין הספרות בסדר הפוך.

דוגמא: עבור המספר 123 יש לייצר את המספר 321.

7. כתוב תוכנית הקולטת מהמשתמש ביטויים חשבוניים הכוללים את הסימן + עד אשר יוקלד ביטוי עם הסימן -.

עבור כל ביטוי התוכנית תייצר מספר חדש ותציגו באופן הבא: הספרות השמאליות הינן תוצאת חיבור המספרים והספרות שמיימין הן תוצאת חיבור המספרים.

דוגמאות:

6	+	4	=	210
9	+	2	=	711
8	+	5	=	313
5	+	2	=	37
7	+	6	=	113
9	+	8	=	117
10	+	6	=	416
15	+	3	=	1218

ניתן להניח שהמספר הראשון גדול מהמספר השני.

התמונה לקוחה מתוך: https://fbcdn-sphotos-f-a.akamaihd.net/hphotos-ak-prn2/t1/1888550_10152193586557878_1560289186_n.jpg

8. כתוב תוכנית הקולטת מהמשתמש שני מספרים בעלי מספר ספרות זהה (ניתן להניח כי הקלט תקין). התוכנית תבנה מספר חדש מטיפוס int באופן הבא:

- ספרות האחדות במספר הראשון תעיד כמה פעמים תופיע ספרות האחדות מהמספר השני במספר החדש
- ספרות העשרות במספר הראשון תעיד כמה פעמים תופיע ספרות העשרות מהמספר השני במספר החדש
- וכו'
- סדר הספרות במספר החדש יהיה עפ"י סדרן במספר המקורי.
- המספר החדש מוגבל ל-9 ספרות בלבד.

דוגמה:

Enter 2 numbers with the same number of digits:

123 456

res = 455666

כלומר, הספרה 6 תופיע 3 פעמים במספר החדש, הספרה 5 תופיע פעמיים במספר החדש והספרה 4 תופיע פעם אחד בלבד.

דוגמה:

Enter 2 numbers with the same number of digits:

2323 6789

res = 677788999

כלומר, הספרה 9 תופיע 3 פעמים במספר החדש, הספרה 8 תופיע פעמיים במספר החדש, הספרה 7 תופיע 3 פעמים במספר החדש והספרה 6 תופיע פעם אחת בלבד מאחר ולא יתכן שיהיו יותר מ-9 ספרות במספר החדש.

דוגמה:

Enter 2 numbers with the same number of digits:

333 678

res = 666777888

9. כתוב תוכנית הקולטת מהמשתמש מספר וספרה. בתוכנית זו נחלק את המספר לחלקים (מימין לשמאל) כך שאורכו של כל חלק יהיה כערך הספרה שנקלטה. במידה ולחלק האחרון אין מספיק ספרות, הוא ישאר באורך הנותר.

למשל:

עבור המספר 123456 והספרה 2 נחלק את המספר לחלקים הבאים: 12|34|56

עבור המספר 1234567 והספרה 3 נחלק את המספר לחלקים הבאים: 1|234|567

התוכנית תייצר מספר חדש כך שסדר הספרות בכל חלק הפוך.

דוגמה:

Enter a number and a digit:

12345678 3

res = 21543876

חלקי המספר המקורי: 12|345|678 ולאחר שהופכים את סדר הספרות בכל חלק המספר החדש שמתקבל הוא 21|543|876

דוגמה:

Enter a number and a digit:

123456 2

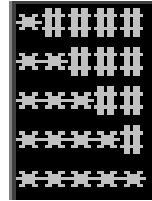
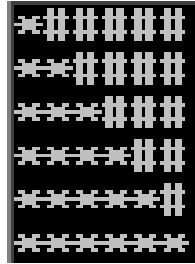
res = 214365

10. כתוב תוכנית הקולטת מהמשתמש מספר.

התוכנית תציג למסך ריבוע באופן הבא: המשולש התחתון שמאלי של הריבוע יהיה עם * והמשולש העליון הימני יהיה עם #.

דוגמאות:

עבור המספר 6 יוצג הריבוע הבא:



עבור המספר 5 יוצג הריבוע הבא:

11. כתוב תוכנית הקולטת מהמשתמש מספר ומציירת משולש שווה שוקיים שבסיסו למעלה.
דוגמא: עבור המספר 7 יצויר המשולש הבא (הקווים הם רק כדי שיהיה לכם יותר נוח לנתח):

```
*_*_*_*_*_*_*
*_**_*_*_*_*
_**_*_*_*_*_*
__**_*_*_*_*
___**_*_*_*
____**_*_*
_____**
_____*
```

12. קלוט מהמשתמש מספר אי-זוגי המייצג בסיס של משולש (ניתן להניח כי הקלט תקין). צייר שעון חול כך שבסיסו העליון והתחתון הם ברוחב הבסיס שנקלט.

דוגמאות:

עבור $\text{base}=3$ יצויר שעון החול הבא:

```
* * *
 * *
  *
   *
  * *
 * * *
```

עבור $\text{base}=5$ יצויר שעון החול הבא:



13. קלוט מהמשתמש מספר והדפס "שטיח" של ריבועים: אורכו ורוחבו של כל ריבוע יהיה כערך המספר שהוקלד, וכן בכל שורה ועמודה יהיו ריבועים כערך המספר שהוקלד.

דוגמאות:

```

Enter size: 2

** **
** **

** **
** **

```

```

Enter size: 3

*** *** ***
*** *** ***
*** *** ***

*** *** ***
*** *** ***
*** *** ***

*** *** ***
*** *** ***
*** *** ***

```

```

Enter size: 4

***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****

***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****

***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****

***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****
***** ***** ***** *****

```

14. כתוב תוכנית שתדפיס עץ ברוש. עץ ברוש מורכב מרצף של משולשים אחד מתחת לשני ולבסוף גזע. קלוט מהמתמש מספר אי-זוגי שיהווה את בסיס המשולש. יש לצייר ברצף אחד-מתחת לשני 3 משולשים, ובסוף לצייר את גזע העץ שהוא עמוד בגובה המספר האי-זוגי שהוקלד מתחת למרכז המשולש האחרון.

לדוגמא, עבור הערך 3:

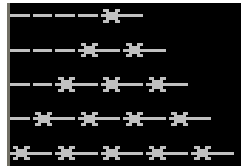
```

  *
 * *
* * *
 *
 * *
* * *
  *
 * *
* * *
  *
 *
  *
  *

```

כדי להקל את העבודה תתחילו בוכנית הקוראת מהמשתמש מספר X. התוכנית תדפיס משולש שווה שוקיים שבסיסו X.

למשל, עבור $X=5$ הפלט יראה כך (בדוגמא יש מקפים במקום רווחים, כדי שיהיה לכם יותר נוח לנתח את הפלט):



4. מערכים ומטריצות

כתוב תפריט המכיל את האפשרויות הבאות: (שימו לב שתפריט אמור לחזור על עצמו עד אשר המשתמש מבקש לצאת!):

1. כתוב תוכנית הקולטת מספרים לתוך 2 מערכים בגודל 5 כל אחד. יש להציג העם ערכי המערך הראשון זהים לערכי המערך השני, אבל בסדר הפוך.

דוגמאות:

- עבור האוסף 1 2 3 4 5 והאוסף 5 4 3 2 1 תוצג ההודעה שהאוספים הפוכים
- עבור האוסף 1 2 3 4 5 והאוסף 1 2 3 5 4 תוצג ההודעה שהאוספים אינם הפוכים

2. כתוב תוכנית הקולטת מספרים לתוך מערך בגודל 5. התוכנית תציג האם הערך של כל איבר המערך גדול מהערך שמשמאלו.

דוגמאות:

- עבור האוסף 20 21 32 65 98 תוצג ההודעה שכל ערך גדול מהערך שמשמאלו.
- עבור האוסף 20 21 70 65 98 תוצג ההודעה שלא כל ערך גדול מהערך שמשמאלו.

3. כתוב תוכנית המגדירה מערך בגודל 10 וקלוט ערכים ל-2 האיברים הראשונים. עליך למלא את שאר איברי המערך כך שערכו של כל איבר יהיה סכום שני האיברים שלפניו. לבסוף יש להציג את ערכי המערך.

דוגמא:

אם קלטנו כמספר הראשון את 2 וכמספר השני את 6, המערך לבסוף יראה כך:
2,6,8,14,22,36,58,94,152,246

4. כתוב תוכנית הקולטת מהמשתמש מערך בגודל 5 ומספר שלם. התוכנית תציג האם כל זוג ערכים שווה למספר שהתקבל.

דוגמא:

עבור המערך [4, 3, 4, 3, 4] והמספר 7 יוצג "אמת" מאחר וסכום האיבר הראשון והשני הוא 7, סכום האיבר השני עם השלישי הוא 7 וכו'.

5. כתוב תוכנית הקולטת ערכים לשני מערכים בגודל זהה.

התוכנית תציג האם ערכי איברי האוסף השני זהים בסדר שלהם לערכי האוסף הראשון, אבל עם סטייה של איבר אחד ימינה.

כלומר:

- האם ערכו של האיבר ה-2 במערך השני שווה לערכו של האיבר ה-1 במערך הראשון
- וגם האם ערכו של האיבר ה-3 במערך השני שווה לערכו של האיבר ה-2 במערך הראשון
- וכן הלאה. ערכו של האיבר הראשון במערך השני יבדק למול האיבר האחרון במערך הראשון

דוגמאות:

עבור שני המערכים הבאים יוצג true:

1	2	3	4	5
9	3	4	6	8

1	2	3	4	5
8	9	3	4	6

עבור שני המערכים הבאים יוצג false:

1	2	3	4	5
9	3	4	6	8

1	2	3	4	5
9	3	4	5	8

6. הגדר מערך של 10 תווים וקלוט לתוכו נתונים. ידוע כי במערך זה יש 2 אותיות גדולות בדיוק, כל השאר קטנות (ניתן להניח שהמשתמש ידידותי). עליך להחליף את כל האותיות בטווח זה לאותיות גדולות.

למשל: עבור המערך: afgKertMh הוא ישתנה להיות afgKERTMh

7. הגדר 3 מערכים בגודל 5 וקרא לתוך שניים מהם ערכים מהמשתמש, על הערכים להיות ממוינים מהקטן לגדול (ניתן להניח כי המשתמש ידידותי). העתק למערך השלישי כל ערך שמופיע גם במערך הראשון וגם במערך השני. לבסוף הדפס את המערך השלישי וכן את מספר האיברים שבו.

דוגמא: עבור המערך 1,2,3,4,5 והמערך 2,4,7,8,9 במערך השלישי יהיו הערכים 2 ו-4, ויוצג בנוסף הערך 2.

8. צמד מספרים יקרא "צמד מראה" אם סדר הספרות שלהם הפוך.

דוגמאות:

- 123 ו- 321 הינם "צמד מראה"
- 121 ו- 121 הינם "מצד מראה"

מערך נקרא "מערך מראה" אם כל זוג ערכים מההתחלה והסוף הינם "צמד מראה".

דוגמאות:

האוסף הבא הינו "אוסף מראה" מאחר והערכים במיקומים 1 ו-6 הינם "צמד מראה" הערכים במיקומים 2 ו-5 הינם "צמד מראה" והערכים במיקומים 3 ו-4 הינם "צמד מראה".

123	121	45	54	121	321
-----	-----	----	----	-----	-----

האוסף הבא אינו "אוסף מראה" מאחר והערכים במיקומים 2 ו-5 אינם "צמד מראה".

123	123	45	54	121	321
-----	------------	----	----	------------	-----

כתוב תוכנית המגדירה מערך בגודל SIZE זוגי (למשל 6) וקולטת לתוכו מספרים. התוכנית תציג הודעה מתאימה האם המערך הינו "אוסף מראה".

ניתן להניח כי גודל המערך שנקלט זוגי.

9. הגדר 2 אוספים בגודל זהה וקלוט לתוכם מספרים.

יש לבדוק ולהציג הודעה האם סכום הספרות זהה באיברים במיקומים תואמים.

דוגמא:

עבור שני האוספים הבאים התוכנית תדפיס שהאוספים תואמים.

14	61	227
----	----	-----

23	43	92
----	----	----

הסבר:

- סכום הספרות באיבר הראשון באוסף הראשון הוא 5 וגם סכום הספרות באיבר הראשון אוסף השני הוא 5
- סכום הספרות באיבר השני באוסף הראשון הוא 7 וגם סכום הספרות באיבר השני באוסף השני הוא 7

- סכום הספרות באיבר השלישי באוסף הראשון הוא 11 וגם סכום הספרות באיבר השלישי באוסף השני הוא 11

10. הגדר מטריצה בגודל 5X5 של תווים ואתחל אותה. קלוט מהמשתמש תו והדפס את האינדקס של העמודה בה התו שהוקלד מופיע הכי הרבה פעמים. אם התו כלל לא מופיע במטריצה יש לתת הודעה מתאימה (שימו לב: אין לעבור על המטריצה בהתחלה כדי לבדוק זאת!).

11. הגדר מטריצה של מספרים בגודל 10X10. שאל את המשתמש בכמה שורות ובכמה עמודות מהמטריצה ברצונו להשתמש, יש לוודא כי הקלט אכן תקין (כלומר בין 1 ל-10). קלוט ערכים למטריצה לפי המימד שהמשתמש ביקש. הצג למסך את המטריצה כפי שהוקלדה ע"י המשתמש (במימד שהמשתמש ביקש). יש למצוא את הערך המקסימלי הנמצא על המסגרת החיצונית של המטריצה ולהציגו למסך.

דוגמאת פלט:

```
Enter rows --> 12
Value should be between 1-10, Try again --> 5
Enter cols --> 11
Value should be between 1-10, Try again --> 4
Enter 20 numbers: 8 5 2 77 44 65 98 54 100 2 5 33 6 55 77 54 1 22 -8 5
The matrix is:
8      5      2      77
44     65     98     54
100    2      5      33
6      55     77     54
1      22     -8      5
The max is 100
```

12. הגדר מערך עם 4 מספרים וקלוט לתוכו ערכים בטווח 0-10. הדפס למסך את הפלט הבא, בהתאם להוראות הבאות:
יש להדפיס למסך פלט כך שעבור כל ערך במערך תהייה עמודה שבתחתיתה הערך ומעליה כוכביות כערך. לדוגמא עבור המערך {4 2 6 5} יודפס הפלט הבא:

```

      *
      * *
*      * *
*      * *
* *   * *
* *   * *
-----
4  2  6  5
Press any key to continue . . .
```

13. בשאלה זו עליכם להגדיר מטריצה בגודל NxM (שיוגדרו כקבועים) ולמלא אותה בצורת נחש כפי שראינו במצגת, אבל הפעם הערך 1 יתחיל בעמודה הימנית למעלה, ירד ויעלה בעמודה ליד וכו' (ראו דוגמאות פלט).

שימו לב: הקוד צריך להיות כללי גם לכמות עמודות זוגית וגם אי-זוגית.

רמז בקרה: בפתרון הטוב ביותר יש רק if אחד והעקרון מאוד דומה לפתרון שראינו בהרצאה.

```

29      28      15      14      1
30      27      16      13      2
31      26      17      12      3
32      25      18      11      4
33      24      19      10      5
34      23      20      9       6
35      22      21      8       7
```

28	15	14	1
27	16	13	2
26	17	12	3
25	18	11	4
24	19	10	5
23	20	9	6
22	21	8	7

14. הגדרה: מטריצה תקרא יהלומית אם יש בה 0-ים במקומות היוצרים צורת יהלום, אין חשיבות לערכים שיש בשאר המקומות.

דוגמאות:

		0		
	0		0	
0				0
	0		0	
		0		

		0	0		
	0			0	
0					0
0					0
	0			0	
		0	0		

כתוב תוכנית הקוראת מהמשתמש מספר N ומקצה מטריצה ריבועית $N \times N$ של מספרים בהתאם.

בקש מהמשתמש להכניס לתוך כל איבר את הערך 0 או 1.

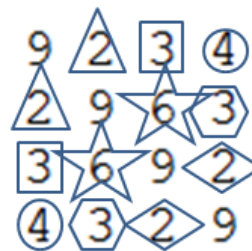
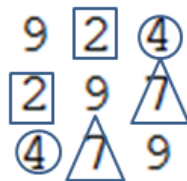
הדפס את המטריצה עם הנתונים.

בדוק האם המטריצה שהוכנסה היא יהלומית והדפס את התשובה.

15. נגדיר "מטריצה מקופלת ראשית" אם כאשר "מקפלים" את הפינה הימנית העליונה לעבר הפינה השמאלית התחתונה הערכים שיושבים אחד על השני זהים. נשים לב שאיברי האלכסון הראשי (מהפינה השמאלית העליונה לפינה הימנית התחתונה) אינם רלוונטים להגדרה.

דוגמאות:

בשתי הדוגמאות הבאות כל זוג מספרים שיושב אחד על השני בקיפול מוקף בצורה זהה:



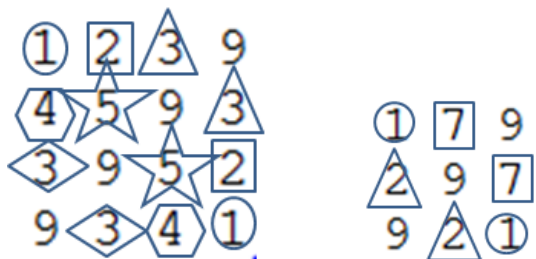
כתוב תוכנית הקולטת מספרים למטריצה ריבועית של מספרים (מספר השורות שווה למספר העמודות) והדפס את המטריצה.

התוכנית תציג הודעה האם המטריצה שהתקבלה היא "מקופלת ראשית".

16. נגדיר "מטריצה מקופלת משנית" אם כאשר "מקפלים" את הפינה השמאלית העליונה לעבר הפינה הימנית התחתונה הערכים שיושבים אחד על השני זהים. נשים לב שאיברי האלכסון המשוני (מהפינה הימנית העליונה לפינה השמאלית התחתונה) אינם רלוונטיים להגדרה.

דוגמאות:

בשתי הדוגמאות הבאות כל זוג מספרים שיושב אחד על השני בקיפול מוקף בצורה זהה:



כתוב תוכנית הקולטת מספרים למטריצה ריבועית של מספרים (מספר השורות שווה למספר העמודות) והדפס את המטריצה.

התוכנית תציג הודעה האם המטריצה שהתקבלה היא "מקופלת משנית".

17. בשאלה זו נדון במטריצה המורכבת מתווים, ובפרט התווים 'י' ו- 'י'.

מסלול במטריצה הוא רצף של איברים צמודים משמאל או מלמטה, המתחיל באיבר הימני העליון של המטריצה ומסתיים באיבר כלשהו בשורה התחתונה. כיוון התנועה ברצף זה הינו שמאלה או מטה בלבד. רצף התווים מכיל את התווים 'י' ו/או 'י' בלבד. התווים הללו מעידים על מיקום התו הבא ברצף.

למשל אם בתא במסלול יש את התו 'י' משמע התו הבא במסלול צריך להיות מתחתיו, ואם התו במסלול הוא 'י' התו הבא במסלול צריך להיות משמאלו.

התו בשורה התחתונה חייב להיות התו 'י'.

הגרל את מימדי המטריצה וקלט לתוכה תווים מהמשתמש והצג את המטריצה. בדוק האם קיים מסלול של קווים מהפינה הימנית העליונה ועד לאיבר כלשהו בשורה התחתונה והצג הודעה מתאימה.

דוגמאות:

עבור המטריצות הבאות יוצג true מאחר ויש מסלול עפ"י ההגדרה הנ"ל.

a	a	a	a	a	!
a	a	a	a	a	!
a	a	a	!	-	-
a	a	!	-	a	a
a	!	-	a	a	a

a	a	a	a	a	!
a	a	a	a	a	!
a	a	a	!	-	-
a	a	!	-	a	a
a	a	!	a	a	a

עבור המטריצה הבאה יוצג false מאחר שאין מסלול, כי התו בשורה התחתונה הוא 'י' ולא 'י'.

```

a a a a a !
a a a a a !
a a a ! - -
a a ! - a a
a a - a a a

```

עבור המטריצה הבאה יוצג false מאחר שאין מסלול, כי באיבר הימני ביותר בשורה השלישית יש '|', ואז האיבר הבא במסלול היה צריך להיות מתחתיו ולא לידו.

```

a a a a a !
a a a a a !
a a a ! - !
a a ! - a a
a a ! a a a

```

בכל הדוגמאות האלה, במקום התו 'a' יכול להופיע כל תו אחר, כולל התווים '|' ו-'-' כאשר אינם חלק מהמסלול.

5. פונקציות

כתוב תפריט המכיל את האפשרויות הבאות: (שימו לב שתפריט אמור לחזור על עצמו עד אשר המשתמש מבקש לצאת!):

1. כתוב פונקציה המקבלת מספר ומחזירה true במידה וכל ספרותיו זהות ו- false אחרת. במידה וכל הספרות זהות, יש להחזיר גם את מספר הספרות.

2. כתוב פונקציה המקבלת מספר ותו, ומדפיסה למסך מסגרת של מעוין. למשל עבור 4 ו-# יודפס (ציירתי פה _ במקום רווחים):

```

#####
(8 סולמיות (כלומר 4 סולמיות, 0 רווחים ו- 4 סולמיות))
###_###
(3 סולמיות, 2 רווחים, 3 סולמיות)
##_##_
(2 סולמיות, 4 רווחים, 2 סולמיות)
#_#_#
(1 סולמיות, 6 רווחים, 1 סולמיות)
##_##_
(2 סולמיות, 4 רווחים, 2 סולמיות)
###_###
(3 סולמיות, 2 רווחים, 3 סולמיות)
#####
(8 סולמיות (כלומר 4 סולמיות, 0 רווחים ו- 4 סולמיות))

```

3. כתוב פונקציה המקבלת מטריצה ריבועית (משמע, מספר השורות שווה למספר העמודות). הפונקציה תחזיר TRUE אם סכום האיברים בכל שורה שווה לסכום האיברים בעמודה המתאימה, ותחזיר FALSE אחרת.

כלומר, יש לבדוק אם סכום איברי השורה הראשונה שווה לסכום העמודה הראשונה וכו'.

דוגמא: עבור המטריצה הבאה הפונקציה תחזיר TRUE, מאחר וסכום איברי השורה הראשונה הוא 8 וכן"ל סכום איברי העמודה הראשונה. סכום איברי השורה השנייה הוא 16 וכן"ל סכום איברי העמודה השנייה וסכום איברי השורה השלישית הוא 8 וכן"ל סכום איברי העמודה השלישית:

2	1	5
6	7	3
0	8	0

4. מטריצת מראה הינה מטריצה שבה איברי הטור הראשון זהים בערכם ובמיקומם לאיברי הטור האחרון, איברי הטור השני זהים בערכם ובמיקומם לאיברי הטור לפני האחרון וכו'.

דוגמאות:

2 המטריצות הבאות הינן מטריצות "מראה":

2	1	5	1	2
6	7	3	7	6
0	8	0	8	0

2	1	5	5	1	2
6	7	3	3	7	6
0	8	0	0	8	0

המטריצה הבאה אינה מטריצת "מראה":

2	1	5	1	2
6	7	3	8	6
0	8	0	8	0

כתוב פונקציה המקבלת מטריצה של מספרים ומימדיה. הפונקציה תחזיר TRUE אם המטריצה היא "מטריצת מראה", FALSE אחרת.

5. הגדרה: מטריצת אפסים שמאלית-עליונה בגודל X היא תת-מטריצה בתוך מטריצה אשר כל ערכיה בטווח $[0][0]$ ועד האיבר $[X-1][X-1]$ הם 0.

דוגמאות:

```

0 0 0 0 0
0 0 0 0 0
0 0 1 0 0
0 0 0 0 0
0 0 0 0 0
max left-top rectangle is 2

```

```

0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 1 0
0 0 0 0 0
max left-top rectangle is 3

0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
max left-top rectangle is 5

```

כתבו את הפונקציה `max0Rectangle` המקבלת מטריצה ריבועית, ומחזירה את גודלה X כפי שמתואר לעיל.

הגבלה: בכל איטרציה אין לבדוק תאים שנבדקו באיטרציה הקודמת (כלומר, היעילות תהייה $O(SIZE \times SIZE)$).

הבהרה 1: הערך שאינו 0 יכול להופיע בכל מיקום במטריצה, ולא בהכרח רק על האלכסון.

הבהרה 2: באיברי המטריצה שאינם חלק ממטריצת האפסים השמאלית עליונה יכול להיות כל ערך. כלומר, לא ספציפית רק הערך 1 ולא ספציפית רק באיבר יחיד במטריצה.

6. משחק איקס-עיגול (איקס-מיקס-דריקס)

בקובץ `XO_starter.cpp` המצורף נתונה תוכנית המדמה את המשחק איקס-עיגול. עליכם להשלים את הקוד כך שהתוכנית תעבוד בצורה מושלמת. שימו לב להגדרת הקבועים ולשימוש בהם. מומלץ מאוד לקרוא את הקוד המצורף ולהבין את הלוגיקה הכללית, ורק אז "לצלול" לתוך המימושים.

מצ"ב קובץ EXE שתוכלו להריץ לראות כיצד התוכנית לבסוף צריכה לעבוד.



הגדרת המשחק:

- עם התחלת התוכנית יתחיל משחק, ויש לבצע בו את השלבים הבאים עד לקבלת תוצאה סופית של המשחק:

א. ציור לוח המשחק במצבו הנוכחי למסך
 ב. בקשת מהלך עבור משתמש: כלומר לבקש שורה ועמודה בגבולות הלוח. יש לוודא כי הקלט תקין וכן כי המשבצת המבוקשת עדיין אינה מסומנת. אין להמשיך לשלב הבא לפני קבלת נתונים בשלב זה.

ג. בדיקה האם יש תוצאה סופית למשחק: ניצחון של אחד השחקנים או תיקו
 ד. בכל מקרה אחר יש לדאוג שבסיבוב הבא יתקבלו נתונים עבור המשתמש השני

- עם סיום המשחק יש להציג את אחת מתוצאות המשחק:

- השחקן הראשון (X) ניצח
 - השחקן השני (O) ניצח
 - תיקו (הלוח מלא, ואף שחקן לא מנצח).
- לאחר הצגת תוצאת המשחק יש לשאול את המשתמש האם ברצונו לשחק שוב, כך עד אשר המשתמש יבחר לסיים.
דוגמאת פלט באיטרציה מסוימת:

```
The current board:
X |  |  | 
---|---|---
  | O |  | 
---|---|---
  |  | X | 
---|---|---

Please enter cordination for player #2: 2 2
The cell is already taken!
Press any key to continue . . .
```

6. מחרוזות

1. כתוב פונקציה המקבלת מערך של מחרוזות (מטריצה של תווים), את מספר השורות במטריצה ומחרוזת נוספת. הפונקציה תחזיר כמה שורות במטריצה זהות למחרוזת שהתקבלה.

דוגמא:

השורות במטריצה:

Hi
Good
Good morning
Hi
This is nice!

והמחרוזת היא Hi.

הפונקציה תחזיר 2 משום שהמחרוזת Hi מופיעה פעמיים במטריצה.

2. כתוב פונקציה המקבלת 3 מערכים של מחרוזות (מטריצות של תווים) (text1, text2, text3) וכן את מספר השורות במטריצות (המטריצות בעלות מימדים זהים). ב- 2 המטריצות הראשונות יש טקסט ובמטריצה השלישית זבל.

הפונקציה תשים מחרוזות ב-text3 באופן הבא:

במידה והאורך הכולל של השורה המתאימה ב-text1 וב-text2 קטן מאורך שורה אפשרית, נעתיק את השורה המתאימה מ-text1 ל-text3 ואח"כ נשרשר את השורה המתאימה מ-text2. אחרת נשים ב-text3 שורה ריקה.

כתוב main המגדיר 3 מטריצות שכאלו וקרא נותנים לתוך 2 המטריצות הראשונות. הפעל את הפונקציה, ולבסוף הדפס את המטריצה השלישית.

דוגמא לפלט לתוכנית:

```
Please fill text1 with 3 lines, max 10 letters in each line:
hi
good
abc
Please fill text2 with 3 lines, max 10 letters in each line:
hihi
morning
def
text3 is:
hihihi

abcdef
Press any key to continue . . . _
```

3. כתוב פונקציה המקבלת מערך של מחרוזות (מטריצה של תווים), מספר השורות המטריצה, מספר שלם ומערך בוליאני שאורכו כמספר השורות במטריצה. הפונקציה תחזיר מספר שלם. הפונקציה תחזיר כמה שורות במטריצה יותר קצרות מהמספר שהתקבל וכן עבור כל שורה תעדכן במיקום המתאים במערך הבוליאני האם השורה קצרה מהמספר שהתקבל או לא (כלומר, אם השורה הראשונה קצרה מהמספר שהתקבל, ערכו של המערך הבוליאני במקום ה-0 יהיה true, אחרת false). לבסוף הפונקציה תחזיר את מספר השורות שאורכן קצר מהמספר שהתקבל.
- ב- main קלוט נתונים לתוך המטריצה, והגרל מספר בין 5-20 שישלח כמספר המבוקש לפונקציה. הפעל את הפונקציה, הדפס את מספר השורות שאורכן קצר מהמספר שהוגרל וכן את האינדקסים של שורות אלו.
4. כתוב פונקציה המקבלת מערך של מחרוזות (מטריצה של תווים) ואת מספר השורות במטריצה. הפונקציה תחזיר true בכל שורה המחרוזת ארוכה מהמחרוזת שבשורה שלפניה.
5. כתוב פונקציה המקבלת מערך של מחרוזות (מטריצה של תווים) ואת מספר השורות במטריצה. הפונקציה תחזיר true האם המחרוזות מסודרות לקסיקוגרפית.
6. כתוב פונקציה המקבלת 2 מחרוזות. הפונקציה תבדוק האם המחרוזת השניה היא תת-מחרוזת במחרוזת הראשונה. אם כן, הפונקציה תחזיר את אינדקס ההתחלה של המחרוזת השניה בראשונה, אחרת הפונקציה תחזיר -1.

דוגמאות:

- עבור המחרוזת abcdef והמחרוזת cde יוחזר 2 מאחר והמחרוזת cde מופיעה ברצף במחרוזת הראשונה החל מאינדקס 2.

- עבור המחרוזת abcdef והמחרוזת cdt יוחזר -1 מאחר ורצף האותיות cdt אינו מופיע ברצף במחרוזת הראשונה.

7. כתוב פונקציה המקבלת 2 מחרוזות. הפונקציה תבדוק האם אותיות המחרוזת השניה מופיעים בסדר זהה במחרוזת הראשונה. אם כן, הפונקציה תחזיר true, אחרת הפונקציה תחזיר false.

דוגמאות:

- עבור המחזורות abcdef והמחזורות bde יוחזר true מאחר והתווים d, b, e מופיעים במחזורות הראשונה בסדר זה (אך לא ברצף).
 - עבור המחזורות abcdef והמחזורות bdc יוחזר false מאחר והתווים d, b, c אכן מופיעים במחזורות הראשונה, אך לא בסדר זה.
8. כתוב תוכנית המגדירה מערך של מחזורות. יש להזיז את איבר המערך כך שבשורה השניה תהיה המחזורות הראשונה, בשורה השלישית המחזורות השניה וכו'. השורה האחרונה תועתק במקום המחזורות הראשונה. לבסוף התוכנית תדפיס את המערך המעודכן.

שימו לב: במידת הצורך יש להשתמש בפונקציות שלמדנו, ולא "להמציא את הגלגל מחדש"!!

דוגמא לפלט לתוכנית:

```
Please enter 5 lines <maximum 10 chars>
aaa
bb
ccccc
dd
ee
text after rotation:
aaa
bb
ccccc
dd
Press any key to continue . . . =
```

7. מצביעים

1. בשאלה זו 4 סעיפים. בכל סעיף יש קטע קוד ועליכם לקבוע האם הקוד מתקמפל או לא. אם לא, יש לציין מדוע, ואם כן – יש לכתוב מה הפלט. את התשובה יש להגיש בכתב ברור/ מודפס.
- יש להניח שהכתובת ב- main מתחילים בכתובת 1000 ובפונקציה מ- 2000. מומלץ מאוד לצייר את תמונת הזכרון כפי שאני עושה במצגת.
- אין להריץ תרגילים אלו על מחשב!

(a)

```
#include <iostream>
using namespace std;

void foo(int& x)
{
    x = x%2 + 10;
}

int* goo(int& y)
{
    y = y/100 + y%10;
```

```

        return &y;
    }

void main()
{
    int temp = 1234;
    int* p = goo(temp);
    cout << temp << endl;
    foo(*p);
    cout << temp << endl;
}

```

(b)

```

#include <iostream>
using namespace std;

int f1(int& x, int* y)
{
    x*= *y;
    *y += x;

    return *y / x;
}

void main()
{
    int num1=4, num2=5;

    int res = f1(num1, &num2);
    cout << "num1=" << num1 << " num2=" << num2
        << " res=" << res << endl;
}

```

(c)

```

#include <iostream>
using namespace std;

void func(int** ptr)
{
    **ptr = 99;
}

void main()
{
    int x = 30;
    int* pX = &x;
    func(&pX);
    cout << *pX << endl;
}

```

(d)

```

#include <iostream>
using namespace std;

void myFunc(int** x, int* y, int z)
{

```



```

    y = &z;
    x = &y;
}

void main()
{
    int x=3, *y, **z;
    myFunc(x, y, z);
    cout << "x=" << x << " *y=" << *y << " **z=" << **z << endl;
}

```

2. מחוברת התרגילים, עמ' 41 שאלה 1 (אין להריץ על המחשב).
3. מחוברת התרגילים, עמ' 43 שאלה 3 (א+ב). שימו לב שכאשר כתוב שהפונקציה מקבלת `char* s` היא מקבלת את כתובת תחילת המערך (שקול ל- `char str[]`).
4. כתוב פונקציה המקבלת מערך וגודלו. הפונקציה תשים בכל איבר את ריבוע האינדקס. כלומר, באיבר הראשון את הערך 0, באיבר 1 את הערך 1, באיבר 2 את הערך 4 וכו'.
- הגבלה: אין להשתמש בגוף הפונקציה עם [] אלא רק עם פוינטר מטייל.
5. `void fibArray(int A[], int n);`

בפונקציה זו, הינך מתבקש למלא את האיברים של המערך A באיברים של סדרת פיבונאצ'י (ניתן לקרוא על סדרת פיבונאצ'י בויקיפדיה), באופן כזה שרק תאים שהאינדקס שלהם הוא מספר בסדרה, יקבלו את הערך של הסדרה. שאר האיברים במערך, שהאינדקס שלהם אינו איבר בסדרת פיבונאצ'י, ישארו ללא שינוי. לדוגמא, עבור מערך בגודל 20 שאיבריו בהתחלה היו 1-1, לאחר הקריאה לפונקציה הוא יראה כך:

0, 1, 2, 3, -1, 5, -1, -1, 8, -1, -1, -1, -1, 13, -1, -1, -1, -1, -1, -1

אילוצים רק לפונקציה זו:

- בפונקציה לא יופיע סוגר מרובע כלשהו, פרט לשורת ההצהרה.
- יש לבצע ע"י מצביע מטייל, כלומר מצביע שהוא בעצמו יצביע כל פעם על תא אחר במערך, ולתוך תא זה יוצב הערך המתאים.
- בנוסף 5 נקודות: המצביע לא יעבור על כל התאים, אלא יקפוץ ישירות רק על התאים שהאינדקס שלהם הוא איבר בסדרת פיבונאצ'י.
- ניתן להניח שגודל המערך הוא לפחות שני איברים ואין צורך לבדוק זאת, כלומר ניתן להציב את שני האיברים הראשונים בסדרה במערך ללא בדיקה.

8. אריתמטיקה של מצביעים

1. הגדר 4 מערכים של מספרים באורך 5 כל אחד. הגדר מערך של כתובות A, כך שבאיבר הראשון תהיה כתובת המערך הראשון שהגדרת, באיבר השני כתובת המערך השני וכו'.
- בעזרת המערך A בלבד, יש להדפיס את הערכים הזוגיים שבמערכים.

2.

א. כתוב פונקציה המקבלת מערך של מספרים שלמים, גודלו, גודל קפיצה ומספר. הפונקציה תמלא את כל האיברים שהאינדקס שלהם הוא מחלק שלם של גודל הקפיצה במספר שהתקבל.
דוגמא, עבור מערך בגודל 10, קפיצה בגודל 3 והמספר 2, הפונקציה תשים את הערך 2 באיברים שהאינדקס שלהם הוא 0, 3, 6 ו-9 (האינדקסים המתחלקים ב-3, גודל הקפיצה שהתקבל).
הגבלה: מעבר על איברי המערך יהיה באמצעות פוינטר מטייל בלבד.
ב. כתוב main:

1. הגדר מטריצה בגודל 10x10 של מספרים.
2. באמצעות הפונקציה מסעיף א', יש למלא את הערכים בכל עמודה במטריצה עם האינדקס של העמודה. כלומר:
 - ערכי כל האיברים בעמודה הראשונה יהיה 0
 - ערכי כל האיברים בעמודה השניה יהיה 1
 - וכד'
3. הדפס את ערכי המטריצה

פלט התוכנית יהיה:

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

3.

א. כתוב פונקציה המקבלת מערך של מספרים שלמים, גודלו ומספר. הפונקציה תמלא את כל איברי המערך במספר שהתקבל.
הגבלה: יש להשתמש בפוינטר מטייל (כלומר, טיול על כתובות המערך). אין בשום מקרה להשתמש ב- [].

ב. כתוב את main – הבא:

1. הגדר מטריצה בגודל 10x10 של מספרים.
2. באמצעות קריאה אחת בלבד לפונקציה מהסעיף הראשון יש למלא את כל המטריצה באפסים.
3. יש למלא את המשולש העליון הימני במטריצה עם מספר השורה. כלומר:
 - בשורה הראשונה כל הערכים יהיו 1
 - בשורה השניה, החל מהאיבר השני כל הערכים יהיו 2
 - וכד'
4. הדפס את ערכי המטריצה

פלט התוכנית יהיה:

1	1	1	1	1	1	1	1	1	1
0	2	2	2	2	2	2	2	2	2
0	0	3	3	3	3	3	3	3	3
0	0	0	4	4	4	4	4	4	4
0	0	0	0	5	5	5	5	5	5
0	0	0	0	0	6	6	6	6	6
0	0	0	0	0	0	7	7	7	7
0	0	0	0	0	0	0	8	8	8
0	0	0	0	0	0	0	0	9	9
0	0	0	0	0	0	0	0	0	10

4.

מה עושה הפונקציה הבאה?

אין להריץ פונקציה זו, אלא לבצע הרצה יבשה ולתאר בקצרה מה התוצר הסופי. ניתן להניח כי $rows < 10$.

```
void f(int M[][10], int rows)
{
    int count[100] = {0}, i, *ptr = (int*)M;

    for ( i = 0; i < rows*10; i++ )
        count[ptr[i]]++;

    for ( i = 0; i < 100; i++ )
        while ( count[i]-- )
            *ptr++ = i;
}
```

9. רקורסיות

1. להלן 4 פונקציות.

עבור כל פונקציה יש לצרף עץ המתאר הרצה יבשה של הקוד עבור הערכים שמצויינים בהערה בשורת חתימת הפונקציה, וכן לתאר במשפט בודד מה הפונקציה עושה (לציין מה הפונקציה מבצעת ולא איך).

שימו לב: אין להריץ שאלה זו, אלא לפתור ולהגישה על נייר.

```
int foo(int x)    // run with x=529
{
    if (x < 10)
        return x;

    return foo(x/10) + x%10;
}

int goo(int x[], int s) // run with x={12, 91, 28}, s=3
{
    if (s == 1)
        return x[s-1];
}
```

```

        return goo(x, s-1) + x[0];
    }

void moo(int n) // run with n=4
{
    if (n == 0)
        return;

    moo(n-1);
    for (int i=0 ; i < n ; i++)
        cout << " * ";
    cout << endl;
}

int koo(int x) // run with x=529
{
    if (x < 10)
        return 1;

    return 1 + koo(x/10);
}

```

כתוב תפריט המכיל את האפשרויות הבאות: (שימו לב שתפריט אמור לחזור על עצמו עד אשר המשתמש מבקש לצאת!):

2. כתוב פונקציה רקורסיבית המקבלת מספר ומחזירה את מספר הספרות הזוגיות.
 3. כתוב פונקציה רקורסיבית המקבלת מספר ומחזירה true אם כל ספרותיו זוגיות, false אחרת.
 4. כתוב פונקציה רקורסיבית המקבלת מערך של מספרים וגודלו ומחזירה את מספר הערכים הזוגיים.
 5. כתוב פונקציה רקורסיבית המקבלת מערך של מספרים וגודלו ומחזירה true אם כל הערכים בו זוגיים, false אחרת.
 6. כתוב פונקציה רקורסיבית המקבלת מחרוזת ומחזירה true אם כל התווים בו הם אותיות גדולות, false אחרת.
 7. נגדיר "מספר מתחלף" כמספר בו כל זוג ספרות שכנות (או צמודות) הינו בעל זוגיות שונה.
- כתוב פונקציה רקורסיבית המקבלת מספר שלם חיובי n ותחזיר true אם הוא "מספר מתחלף", אחרת תחזיר false.

דוגמאות:

- עבור המספר 163458 יוחזר true כי ליד כל ספרה זוגית יש ספרה אי זוגית.
- עבור המספר 1634589 יוחזר true כי ליד כל ספרה זוגית יש ספרה אי זוגית.
- עבור המספר 163789 יוחזר false כי הספרות 3 ו-7 צמודות ושניהן אי-זוגיות.

8. כתוב פונקציה רקורסיבית המקבלת מספר חיובי שלם ותחזיר true אם קיים לפחות זוג אחד של ספרות מתחלפות כפי שהוגדר בשאלה הקודמת, אחרת הפונקציה תחזיר false.

9. כתוב פונקציה רקורסיבית המקבלת שני מספרים חיוביים שלמים a ו-b. הפונקציה תחזיר את שארית החלוקה של a ב-b (כלומר את תוצאת a%b) באמצעות שימוש בפעולות חיבור וחסור בלבד.

10. כתבו את הפונקציה הרקורסיבית switch_letters: void switch_letters(char *beg, char *end). הפונקציה מקבלת מצביע לתו הראשון במחרוזת, beg, ומצביע לתו האחרון במחרוזת, end. הפונקציה תהפוך המחרוזת מהסוף להתחלה. דוגמא: עבור המחרוזת: "abcde" יועברו לפונקציה כתובת התו 'a' בתור beg וכתובת התו 'e' בתור end. בסיום הפונקציה, המחרוזת המקורית תהפוך להיות "edcba".

11. כתוב פונקציה רקורסיבית המקבלת 2 מספרים שלמים. הפונקציה תחזיר true אם סכום ספרות המספר הראשון שווה לערך המספר השני, אחרת הפונקציה תחזיר false.

דוגמאות:

- עבור המספרים 123 ו-6 יוחזר true
- עבור המספרים 124 ו-6 יוחזר false
- עבור המספרים 4362 ו-15 יוחזר true

12. כתוב פונקציה רקורסיבית המקבלת 2 מחרוזות. הפונקציה תחזיר את מספר התווים הזהים הנמצאים במיקום זהה בשתי המחרוזות. הפונקציה אינה מבדילה בין אותיות קטנות לגדולות.

דוגמה:

עבור המחרוזות aBdRe ו-ASdRle יוחזר 3, מאחר שבשתי המחרוזות התו במקום ה-0 הוא A, התו במקום ה-2 הוא D, והתו במקום ה-3 הוא R.

שימו לב: ייתכן ואורכי המחרוזות אינם שווים.

13. כתוב פונקציה רקורסיבית המקבלת מערך של תווים ואורכו ומחזירה true אם כל התווים בו רק אותיות גדולות או רק אותות קטנות, false אחרת.

למשל: עבור ABCD ו-bacd יוחזר true, ועבור AkJL יוחזר false.

14. כתוב פונקציה רקורסיבית המקבלת מספר בבסיס 10 ומחזירה מספר חדש המייצג את המספר שהתקבל בבסיס 2.

15. כתוב פונקציה רקורסיבית המקבלת מספר בבסיס 10 ומספר נוסף המייצג בסיס (שיכול להיות 2 או 8 בלבד). הפונקציה מחזירה מספר חדש המייצג את המספר שהתקבל בבסיס שהתקבל.

16. כתוב פונקציה רקורסיבית המקבלת מערך של מספרים וגודלו, וכן ספרה בין 1-9. הפונקציה תחזיר true אם יש רצף של מספרים במערך מ-1 ועד הספרה, אחרת תחזיר false.

דוגמאות:

- עבור המערך 3,1,2,3,4,6,3 גודלו 7 והספרה 4 יוחזר true
- עבור המערך 3,1,2,1,2,3,4 גודלו 7 והספרה 4 יוחזר true
- עבור המערך 1,2,3,4,1,5,1 גודלו 7 והספרה 4 יוחזר true
- עבור המערך 3,1,2,1,2,3,5 גודלו 7 והספרה 4 יוחזר false
- עבור המערך 3,1,2,3,5,6,3 גודלו 7 והספרה 4 יוחזר false

17. כתוב פונקציה רקורסיבית המקבלת מערך של מחרוזות, מספר האיברים במערך ותו. הפונקציה תחזיר את מספר המופעים של התו במערך המחרוזות.

דוגמא:

- עבור מערך המחרוזות {"gogo", "momo", "yoyo"} שאורכו 3 והתו o הפונקציה תחזיר 6 משום שתו זה מופיע 6 פעמים בכל המחרוזות.

18. נתונה הסדרה (משמאל לימין): 1, 2, 3, 6, 4, 13, 7, 24, 11, 42

- נתון כי תמיד שלושת האיברים הראשונים בסדרה הם: 3, 2, 1
- כל איבר שנמצא במקום זוגי בסדרה, החל מהמקום ה-4, הוא סכום שלושת האיברים הקודמים לו
- כל איבר הנמצא במקום אי-זוגי בסדרה, החל מהמקום ה-5, הוא ההפרש המוחלט שבין שני האיברים במקומות הזוגיים שלפניו.

כתוב פונקציה רקורסיבית לחישוב האיבר ה-h בסדרה זו (הקלט לפעולה יהיה המקום ה-h והערך שיוחזר יהיה ערכו של האיבר במקום זה).

רמז: היזכרו בפתרון של סדרת פיבונאצ'י

19. בהינתן סולם, ניתן לטפס על שלביו ע"י טיפוס של שלב בודד או ע"י טיפוס שני שלבים בו-זמנית (לא ניתן לדלג על 3 שלבים או יותר).

כתוב פונקציה רקורסיבית המקבלת כפרמטר מספר שלבים בסולם וחזירה את מספר האפשרויות השונות לטיפוס עליו.

דוגמה:

עבור סולם עם 4 שלבים, ניתן לטפס עליו בכל אחת מהאפשרויות הבאות:

- $1 \leftarrow 1 \leftarrow 1 \leftarrow 1$
- $2 \leftarrow 1 \leftarrow 1$
- $1 \leftarrow 2 \leftarrow 1$

$$\begin{aligned} &1 \leftarrow 1 \leftarrow 2 \quad - \\ &2 \leftarrow 2 \quad - \end{aligned}$$

ולכן הפונקציה תחזיר את הערך 5 כי יש 5 אפשרויות שונות לטפס על שלבי הסולם.

20. כתבו פונקציה המקבלת מספר שלם ומציירת שנתות של סרגל כך שהמספר שהתקבל הוא מספר השנתות במרכז הסרגל.

דוגמא עבור $n=3$:

```

-
--
-
---
-
--
-

```

דוגמא עבור $n=4$:

```

-
--
-
---
-
--
-
----
-
--
-
---
-
--
-

```

10. מיונים וחיפושים

1. שאלה 1:

נניח שאברי המטריצה ממוינים בסדר לא יורד, כלומר בכל שורה הערכים ממוינים מקטן לגדול וגם לכל $1 \leq i \leq R-1$ הערך האחרון בשורה $i-1$ קטן או שווה לערך הראשון בשורה i .

ברצוננו לממש פונקציה אשר בהנתן המטריצה וערך מסוים x מחזירה את מספר השורה בה נמצא הערך בתוך המטריצה. ניתן להניח כי בוודאות x נמצא במטריצה. יש להשלים את הקוד להלן במלבנים הלבנים כך שתתקבל הפונקציה המבוקשת:

```
const int R = 10;
const int C = 20;
```

```
int SearchRow(int M[][C], int x)
```

```
{
```

```

int min = 0, max = R-1, mid;

while ( min <= max ) {

    mid = (min + max)/2;

    if 

        return mid;

    if ( x < M[mid][0] )

    else

        min = mid + 1;

}



}

```

2. ראינו בכיתה מימוש למיון Bubble-Sort כך שבסיום האיטרציה הראשונה הערך הגדול ביותר יהיה במיקום האחרון במערך, בסוף האיטרציה השניה הערך השני בגודלו נמצא באיבר לפני הסוף וכו'.

כתבו את הפונקציה מחדש כך שבסיום האיטרציה הראשונה הערך הקטן ביותר יהיה באיבר הראשון במערך, בסוף האיטרציה השני האיבר הקטן השני יהיה במיקום השני וכו'.

3. כתבו פונקציה המקבלת מטריצה ריבועית של מספרים. הפונקציה תמיין את ערכי האלכסון הראשי מהקטן לגדול.

דוגמה:
עבור המטריצה:

9	3	6	5
2	4	8	2
7	2	2	1
9	8	3	6

יש לעדכנה להיות:

2	3	6	5
2	4	8	2
7	2	6	1
9	8	3	9

4. כתבו פונקציה המקבלת מחרוזת הכוללת אותיות קטנות וגדולות. הפונקציה תמיין את אותיות המחרוזת מהאות הקטנה לגדולה, אך לא תבדיל בין אותיות גדולות לקטנות

דוגמאות:

- עבור המחרוזת zBa הפונקציה תמיין אותה להיות aBz
- עבור המחרוזת zZaAC הפונקציה תמיין אותה להיות aACzZ

הנחה: כאשר יש אות גדולה וקטנה זהה, לא משנה מי תופיע קודם

5. כתוב פונקציה המקבלת מטריצה של מספרים, שעמודותיה ממוינות מהערך הקטן לגדול, תקבל את מספר השורות במטריצה וערך לחיפוש. הפונקציה תחזיר בפרמטרי פלט את השורה והעמודה בה נמצא הערך. במידה ואינו קיים הפונקציה תשים בפרמטרי הפלט את הערך 1-.

למשל, עבור המטריצה הבאה והערך 7:

2	1	5	1
3	6	6	2
8	7	7	3
9	8	9	4

הפונקציה תחזיר שורה 2 ועמודה 1 כי במיקום זה קיים הערך 7. הפונקציה יכלה להחזיר גם את שורה 2 ועמודה 2.

6. כתוב פונקציה המקבלת מטריצה של מספרים. יש למיין את ערכי המטריצה תוך שימוש באלגוריתם Bubble Sort שנלמד בכיתה כך שהערך הנמוך ביותר יהיה בפניה השמאלית העליונה והערכים יסודרו בערך עולה לפי שורותיהם.

שימו לב: יש להשתמש במיין ולא רק לשים ערכים חדשים במטריצה (כלומר אין להשתמש במערך עזר).

דוגמה לאיך המטריצה צריכה להיראות לאחר המיין:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

11. מבנים

1. ניהול אנשי קשר:

א. הגדירו את המבנה `phone_number_t` ששדותיו הם מחרוזת באורך 4 עבור הקידומת, ומספר שלם עבור המספר.

ב. כתבו פונקציות קלט ופלט למבנה `phone_number_t`.

ג. כתבו את המבנה `contact_info_t` אשר שדותיו הם שם איש הקשר (מחרוזת בגודל 20), מערך של 3 משתנים מסוג `phone_t`, מספר מספרי הטלפון השמורים בפרטי הקשר (מקסימום 3) וכתובת האימייל של הקשר (מחרוזת בגודל 30).

ד. כתבו פונקציות קלט ופלט למבנה `contact_info_t`. יש להקפיד על מודולריות.

ה. הגדירו ב- `main` מערך של 3 משתנים מטיפוס `contact_info_t`. קראו לתוכם נתונים והדפיסו את הנתונים. יש להשתמש בפונקציות שהוגדרו בסעיף הקודם.

2. מחוברת התרגילים של מדעי המחשב, עמוד 52 שאלה 1.

- בסעיף ג' יש להשתמש במיון בועות כפי שנלמד בכיתה, ולא ב- `merge sort` כמבוקש בשאלה.
- בסעיף ד' ניתן להתעלם מדרישת היעילות.

3. מחוברת התרגילים של מדעי המחשב, עמוד 52 שאלה 2. הפונקציה אינה חייבת להיות רקורסיבית.

12. הקצאות דינאמיות

כתוב תפריט המכיל את האפשרויות הבאות: (שימו לב שתפריט אמור לחזור על עצמו עד אשר המשתמש מבקש לצאת!):

1. כתוב פונקציה המקבלת מחרוזת ומחזירה מחרוזת חדשה המכילה את האות הראשונה מכל מילה במחרוזת שהתקבלה, וכל אות יכולה להופיע במחרוזת החדשה מקסימום פעם אחת בלבד.

דוגמאות:

- עבור המחזורת Hello World תוחזר המחזורת HW
- עבור המחזורת Good morning Sunshine Moshe תוחזר המחזורת GmS
- עבור המחזורת num of student תוחזר המחזורת nos

2. כתוב פונקציה המקבלת מחזורת ומחזירה מערך של מספרים שמספר איבריו הוא כמספר המשפטים במחזורת וערכי איבריו הם מספר התווים בכל משפט במחזורת.

אם במחזורת אין משפטים יוחזר NULL.

שימו לב:

- משפט נגמר בתו נקודה (.) או בתו סימן קריאה (!)
- ניתן להניח שיש לפחות רווח אחד בין משפט למשפט (אך יתכן גם יותר מרווח אחד)
- רווחים בתחילת המשפט אינם נספרים באורך המשפט (כלומר, אורך המשפט הוא מהתו הראשון שאינו רווח ועד הנקודה או סימן הקריאה (לא כולל))

דוגמאות:

- עבור המחזורת:
"Hi! This is a lovely day. Please join me for breakfast."
יוחזר מערך באורך 3 עם הערכים הבאים: [2, 20, 28].
- עבור המחזורת:
"Hi! This is a lovely day. Please join me for breakfast"
יוחזר מערך באורך 2 עם הערכים הבאים: [2, 20] (מאחר אין נקודה בסוף המחזורת, המשפט האחרון אינו נספר).
- עבור המחזורת:
" Hi ! This is a lovely day. Please join me for breakfast."
יוחזר מערך באורך 3 עם הערכים הבאים: [3, 20, 28] (המשפט הראשון באורך 3, מאחר והרווח לפני '!' נספר באורך המשפט).
- עבור המחזורת "" יוחזר NULL ופרמטר הפלט יעודכן ל-0.
- עבור המחזורת "ab" יוחזר NULL ופרמטר הפלט יעודכן ל-0.

3. הטיפוסים long ו-int מוגבלים בטווח הערכים אותם הם יכולים להכיל. לכן בתרגיל זה נבחר לייצג מספר בתוך מערך, כך שבכל איבר במערך תהייה ספרה אחת.

דוגמא:

המספר 1234 יכול להיות מיוצג במערך באופן הבא:

1	2	3	4
---	---	---	---

או למשל במערך הבא (אין משמעות לאפסים מובילים במספר):

0	1	2	3	4
---	---	---	---	---

כתבו את הפונקציה add אשר מקבלת 2 מערכים שכל אחד מהם מייצג מספר כפי שהוגדר לעיל, ואת שני גדלי המערך. הפונקציה תייצר ותחזיר מערך חדש שייצג את המספר שהוא סכום ערכי 2 המספרים שהתקבלו. כמו כן, הפונקציה תחזיר בפרמטר פלט את אורך המערך המוחזר.

דוגמא 1:

עבור המערך

1	2	3	4
---	---	---	---

והמערך

9	7
---	---

המייצגים את המספרים 1234 ו- 97 בהתאמה, יש לייצר ולהחזיר את המערך:

1	3	3	1
---	---	---	---

$$(1234+97=1331)$$

דוגמא 2:

עבור המערך

9	9	9	9
---	---	---	---

והמערך

9	7
---	---

המייצגים את המספרים 9999 ו- 97 בהתאמה, יש לייצר ולהחזיר את המערך:

1	0	0	9	6
---	---	---	---	---

או למשל את המערך:

0	1	0	0	9	6
---	---	---	---	---	---

$$(9999+97=10096)$$

לתשומת ליבכם, בדוגמא 2 בתוצאה יש ספרה אחת יותר מאשר במספר הארוך יותר.

הגבלה: אין בשום שלב להמיר את איברי המספר ממערך לטיפוס אחר.

4. הטיפוסים long ו- int מוגבלים בטווח המספרים אותם הם יכולים להכיל. לכן בתרגיל זה נבחר לייצג מספר בתוך מערך, כך שבכל איבר במערך תהייה ספרה אחת בלבד.

דוגמא:

המספר 1234 יכול להיות מיוצג במערך:

1	2	3	4
---	---	---	---

או למשל במערך (אין משמעות לאפסים מובילים במספר):

0	1	2	3	4
---	---	---	---	---

כתבו את הפונקציה multiply אשר מקבלת 2 מערכים שכל אחד מהם מייצג מספר כפי שהוגדר לעיל, ואת שני גדלי המערך. הפונקציה תייצר ותחזיר מערך חדש שייצג את המספר שהוא מכפלת ערכי 2 המספרים שהתקבלו. כמו כן, הפונקציה תחזיר בפרמטר פלט את אורך המערך המוחזר. פעולת המכפלה צריכה להתבצע על-פי האלגוריתם של חישוב כפל ארוך (ראו תזכורת לכפל ארוך בסוף השאלה).

דוגמא 1:

עבור המערך

1	2	3	4
---	---	---	---

והמערך

9	7
---	---

המייצגים את המספרים 1234 ו- 97 בהתאמה, יש לייצר ולהחזיר את המערך:

1	1	9	6	9	8
---	---	---	---	---	---

$$(1234*97=119,698)$$

הגבלה: אין בשום שלב להמיר את איברי המספר ממערך לטיפוס אחר.

שימו לב: הפתרון צריך להיות כללי כך שיתאים לכל אורך של מספרים.

תזכורת לכפל ארוך ניתן למצוא בקישור:

<http://he.wikibooks.org/wiki/%D7%97%D7%A9%D7%91%D7%95%D7%9F/%D7%9B%D7%A4%D7%9C>

13. הקצאות דינאמיות ומבנים

1. (א) הגדר את המבנה `person_t` המכילה נתונים של אדם: שם פרטי ות.ז. ו-2 מצביעים ל-`person_t*` עבור האבא והאמא, אם קיימים (ערכם יהיה NULL אם אינם קיימים).

(ב) הגדר את המבנה `family_t` המכילה את הנתונים הבאים:

- שם משפחה
- מצביע לאבא מטיפוס `person_t*`
- מצביע לאמא מטיפוס `person_t*`
- מספר הילדים במשפחה
- מערך של מבנים ל-`Person` שגודלו כמספר הילדים

(ג) כתבו פונקציה קלט ל-`person_t` שתקלוט רק את שמו הפרטי ות.ז.. המצביעים לאבא ולאמא יאותחלו ב-NULL.

(ד) כתבו פונקציה פלט ל-`person_t` המציגה את כל נתוניו: שם ות.ז., ואם יש אבא אז רק את שמו, ואם יש אמא אז רק את שמה.

(ה) כתבו פונקציות קלט למבנה `family_t` שתקבל כפרמטרים מצביעים לאבא ולאמא ותצביע אליהם מהמבנה. הפונקציה תקלוט את שם המשפחה ומספר הילדים, תקצה מערך עבור הילדים, ותקרא לתוכם נתונים באמצעות הפונקציה מסעיף ג'. שימו לב: האבא והאמא של ילדים אלו צריכים להיות האבא והאמא של המשפחה.

(ו) כתבו פונקציה פלט למבנה `family_t` (הקפידו על למודלריות).

(ז) הגדירו ב-`main` 2 משתנים מטיפוס `person_t` לתוכו נתונים באמצעות הפונקציה מסעיף ג'.

(ח) הגדירו ב-`main` משתנה מטיפוס `family_t` וקראו לתוכו נתונים באמצעות הפונקציה מסעיף ה'. יש להעביר לפונקציה את המשתנים שהוגדרו בסעיף ז'.

(ט) הדפיסו ב-`main` את נתוני המשפחה.

(י) שחררו את כל הזכרון שהוקצה.

2. בתרגיל זה הנכם מתבקשים לצור "תמונה", להציג אותה ולדחוס אותה. לשם כך נגדיר מבנה הבא:

```
struct Picture
{
```

$$\};$$

למשל אם `width=5` ,`height=4` ו-`data=[*,*,*,*,*,^,*,*,^,^,^,*,^,^,^,^,^]` התמונה תראה כך:

כתבו תוכנית המפעילה את האפשרויות בהמשך, כך שכל סעיף יהיה פונקציה נפרדת.

להדפיס את התמונה לאחר שלב זה.

2. דחיסת תמונה. נגדיר עבור רצף תווים זהים מבנה הבא:

{

במבנה הזה נשמור את התו ואת אורך הרצף. כמו כן נגדיר עבור תמונה דחוסה את המבנה:

 $\{$

כתבו פונקציה:

```
CompressedPic compressThePicture(Picture p);
```

הפונקציה מקבלת תמונה Picture, דוחסת אותה כך שבמקום רצף תווים יישמר מידע על הרצף (התו ואורכו) ושומרת את התמונה הדחוסה במבנה CompressedPic אותו היא מחזירה.

3. פתיחת תמונה דחוסה.

Picture unCompressThePicture(CompressedPic p);

הפונקציה מקבלת תמונה מכווצת CompressedPic, ומפענחת אותה בצורה ההפוכה לפונקציה compressThePicture. יש להדפיס את התמונה לאחר שלב זה.