

עבודות קיץ לתלמידי כיתה י' ו י' 7

- עבודה בנושא יסודות מדעי המחשב
- עבודת קיץ מערך כאוסף

בשבוע השני של תחילת הלימודים בתום החופשה יתקיים מבחן סיכום לנושאים שלמדנו במהלך השנה.

מצב עבודות קיץ שיש לפתור את השאלות כחזרה למבחן תחילת השנה לתלמידי כיתה י' ו י' 7

במידה ונתקלתם בנושא שהינו חדש או שיש בעיה אני מצפה שתנסו להתמודד עם החומרים שנמצאים ברחבי האינטרנט.

יש משמעות לכישורים בלמידה עצמית למרות שאת החומר למדנו יש גם את כולו בקמפוס il

תמיד אפשר לשלוח אלי שאלות.

מציעה לא להמתין לשבוע האחרון של החופשה

אלה לתכנן את החלוקת השאלות תענו לפי נושאים

1. כל יום תפתרו בין 2-4 שאלות
2. הציון במבחן זה יחושב בציון ההגשה של שאלון זה לקראת בחינת הבגרות על החומר של יא
3. אם צריך עזרה תלמידים יתארגנו לשאלות
4. אשמח לשלוח קישור בזום ולענות על שאלות

שתהיה לכולם חופשה שקטה רגועה ותצאו מהמשחקים סרטים ותעבדו גם עם אתגרים לימודיים זה מסיח וגם נותן לכם תחושה של עשייה משמעותית

והעיקר תקשיבו להורים תעזרו להם לאחים לשכנים שזקוקים לעזרה

תשמרו על הוראת פיקוד העורף

חופשה שקטה וטובה

השאלות

בשאלון זה שלושה פרקים.

יש לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפותרים בשפת Java: בכל שאלה שנדרשת בה קליטה, הניחו שבתוכנית כתובה ההוראה:

`Scanner input = new Scanner (System.in);`

פרק ראשון (25 נקודות)

ענו על שאלה 1 – חובה (10 נקודות).

1. "מערך ממזין בחיוביים" הוא מערך מטיפוס שלם שבו כל המספרים הגדולים מ-0 מופיעים בסדר עולה (סדר עולה – מספר גדול או שווה למספר החיובי הקודם לו).

דוגמה:

המערך `arr` שלפניכם הוא "מערך ממזין בחיוביים".

	0	1	2	3	4	5	6	7	8
arr	5	2	-3	17	0	29	-20	-40	29

הסבר: המספרים החיוביים במערך (5, 9, 17, 29, 29) מופיעים בסדר עולה.

כתבו פעולה חיצונית ששמה `posOrder` בשפת Java או `PosOrder` בשפת C#, המקבלת מערך מטיפוס שלם – `arr`.

הפעולה מחזירה `true` אם `arr` הוא "מערך ממזין בחיוביים", אחרת היא מחזירה `false`.

הניחו שיש לפחות שני מספרים חיוביים במערך.

ענו על אחת מן השאלות 2-3 (15 נקודות).

2. "מערך הפרשים" הוא מערך מטיפוס שלם שבו ההפרש בין המספרים בכל שני תאים סמוכים הוא קבוע.

דוגמה למערך הפרשים:

0	1	2	3	4	5
5	9	13	17	21	25

הסבר: ההפרש בין המספרים בכל שני תאים סמוכים במערך זה הוא 4.

ממשו את הפעולה החיצונית שלפניכם:

Java – public static int missingNum (int [] arr)

C# – public static int MissingNum (int [] arr)

הפעולה מקבלת מערך הפרשים – arr שחסר בו תא אחד, ובעקבות זאת יש במערך שני תאים סמוכים שההפרש בין המספרים בהם שונה מן ההפרש הקבוע. הפעולה תחזיר את המספר שאמור להיות בתא החסר. הניחו שגודל המערך arr שמתקבל הוא לפחות 4.

דוגמה:

עבור המערך arr שלפניכם, הפעולה תחזיר את המספר 10.

	0	1	2	3	4	5
arr	6	8	12	14	16	18

הסבר: בין כל שני תאים סמוכים ההפרש בין המספרים הוא 2, חוץ מן התאים באינדקסים 1 ו-2, שההפרש בין המספרים בהם הוא 4. זאת מפני שחסר ביניהם תא ובו המספר 10.

3. בנתיבי תחבורה ציבורית הוצבו מצלמות תנועה, המקצמות כל כלי רכב שעובר על פניהן.

נתונה המחלקה **CarInfo** – מידע על כלי רכב שצולם, ולה שלוש תכונות:

- id – מספר לוחית זיהוי של כלי הרכב שצולם, מטיפוס מחרוזת.
 - privateCar – אם כלי הרכב שצולם הוא פרטי, התכונה היא true, ואם כלי הרכב הוא ציבורי – false.
 - speed – מהירות הנסיעה של כלי הרכב שצולם, מטיפוס שלם.
- הניחו שקיימות פעולות get/Set ו־ set/Set לכל אחת מן התכונות במחלקה.

לכלי רכב שנוסע בנתיב תחבורה ציבורית תירשם עבירת תנועה אם יתקיים לפחות אחד מן המצבים האלה:

- כלי הרכב הוא פרטי.
 - כלי הרכב נוסע מעל המהירות המותרת.
- א. כתבו פעולה פנימית במחלקה **CarInfo** ששמה illegal בשפת Java או Illegal בשפת C#, המקבלת את מהירות הנסיעה המותרת – maxSpeed מטיפוס שלם.
- הפעולה תחזיר true אם כלי הרכב עבר עבירת תנועה (כלומר, אם כלי הרכב הוא פרטי ו/או נסע מעל המהירות המותרת), אחרת הפעולה תחזיר false.

נתונה המחלקה **CameraInfo** – מידע של מצלמת תנועה, ולה שלוש תכונות:

- city – קוד העיר שבה המצלמה מוצבת, מטיפוס שלם. הקוד הוא מספר בין 0 ל-99 (כולל).
- יש 100 ערים, ולכל עיר יש קוד ייחודי.
- maxSpeed – המהירות המותרת באזור המצלמה, מטיפוס שלם.
- cars – מערך מטיפוס **CarInfo** של כלי הרכב שצולמו במצלמה (המערך ללא ערכי null).

הניחו שקיימות פעולות get/Set ו־ set/Set לכל אחת מן התכונות במחלקה.

ב. (1) כתבו פעולה פנימית במחלקה **CameraInfo** ששמה allGood בשפת Java או AllGood בשפת C#, המחזירה true אם כל כלי הרכב שצולמו במצלמה לא עברו עבירת תנועה, אחרת היא מחזירה false.

אפשר להשתמש בפעולה שכתבתם בסעיף א.

(2) כתבו פעולה חיצונית ששמה legalCities בשפת Java או LegalCities בשפת C#.

הפעולה מקבלת מערך – cameras מטיפוס **CameraInfo** של מצלמות שהוצבו ב-100 הערים השונות.

הפעולה תחזיר את כמות הערים של לא אותרה בהן שום עבירת תנועה.

הערה: תיתכן יותר ממצלמה אחת באותה העיר.

אפשר להשתמש בפעולה שכתבתם בסעיף ב(1).

השאלות

בשאלון זה שלושה פרקים.

יש לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפתורים בשפת Java: בכל שאלה שנדרשת בה קליטה, הניחו שבתוכנית כתובה ההוראה:

Scanner input = new Scanner (System.in);

פרק ראשון (25 נקודות)

ענו על שאלה 1 – חובה (10 נקודות).

1. כתבו פעולה חיצונית ששמה multiply בשפת Java או Multiply בשפת C# המקבלת שני מערכים מטיפוס שלם,

arr1 ו- arr2, ששונים בגודלם. נקרא לגודל המערך הקטן יותר k ולגודל המערך הגדול יותר m. על הפעולה להחזיר

מערך חדש מטיפוס שלם בגודל m (גודל המערך הגדול יותר מבין המערכים arr1 ו- arr2).

על המערך המוחזר להיות לפי הפירוט שלהלן:

ערכו של כל אחד מ- k התאים הראשונים יהיה שווה למכפלה של הערכים בתאים המקבילים לו במערכים arr1

ו- arr2.

שאר ערכי התאים במערך המוחזר יהיו זהים לערכי התאים המקבילים להם במערך הגדול.

דוגמה: בעבור המערכים arr1 ו- arr2 שלהלן:

	0	1	2	3	4
arr1	1	-4	4	9	2

	0	1	2	3	4	5	6
arr2	9	2	0	-1	3	11	23

הפעולה תחזיר את המערך הזה:

	0	1	2	3	4	5	6
	9	-8	0	-9	6	11	23

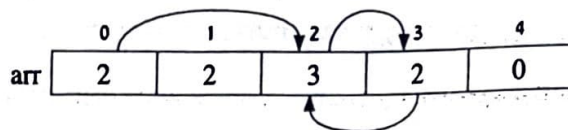
הערה: הניחו שאין תאים ריקים בשני המערכים, ושערכי המערכים תקינים.

/המשך בעמוד 3/

-4-

2. "סריקה לפי ערכי התאים" היא סריקת מערך המתחילה באינדקס 0 ועוברת לאינדקס שמספרו זהה לערך התא

דוגמה:



הניחו שערכי התאים אינם חורגים מגבולות המערך.

1. כל התאים של המעורר נסרקים.

2. הסריקה מסתיימת (מפני שבמערך יש תא שערכו 0 והסריקה מגיעה אליו).

Diagram illustrating the array structure for the sorting process. The array `arr` contains the values `3`, `0`, `1`, `4`, and `2` at indices `0` through `4`. Arrows indicate the mapping from indices to values: `arr[0] = 3`, `arr[1] = 0`, `arr[2] = 1`, `arr[3] = 4`, and `arr[4] = 2`.

המעורר בדוגמה זו הוא מערך מושלם מפני שהסריקה עוברת על כל התאים, ומסתיימת.

arr

0	1	2	3	4	5
1	2	5	1	3	4

מערך זה אינו מערך מושלם מפני שהסריקה אינה מסתיימת.

arr 3 4 1 5 6 0 2

מעריך זה אינו מעריך מושלם מפני שהסריקה מסתיימת בלי שנסרקו התאים 1, 2, 4, 6.

Java – public static boolean isPerfect (int[] arr)

C# – public static bool IsPerfect (int[] arr)

הפעולה תחזיר true אם המערך arr שהתקבל הוא מערך מושלם, אחרת היא תחזיר false.

הערה: אין חובה לשמור על ערכי המערך.

/המשך בעמוד 4/

הניחו שערכי התאים אינם חורגים מגבולות המערך.

3. פיקסל (Pixel) הוא יחידת מידע המתארת נקודה בתמונה דיגיטלית.

כל פיקסל מורכב משילוב של שלושה צבעים: אדום, ירוק וכחול.

כל אחד משלושת הצבעים האלה (אדום, ירוק וכחול) מקבל ערך מספרי מ-0 עד 255 (כולל). המספר מייצג את עוצמת הצבע בנקודה. באמצעות שילוב של שלושת הצבעים בערכים שונים מ-0 עד 255 אפשר לקבל כל גוון בקשת הצבעים.

לקבלת נקודה שהצבע שלה הוא אדום (בין אדום בהיר לאדום כהה), ערכי הצבעים צריכים להיות:

אדום - מ-1 עד 255, ירוק - 0, כחול - 0.

לקבלת נקודה שהצבע שלה הוא ירוק (בין ירוק בהיר לירוק כהה), ערכי הצבעים צריכים להיות:

אדום - 0, ירוק - מ-1 עד 255, כחול - 0.

לקבלת נקודה שהצבע שלה הוא כחול (בין כחול בהיר לכחול כהה), ערכי הצבעים צריכים להיות:

אדום - 0, ירוק - 0, כחול - מ-1 עד 255.

לקבלת נקודות בשאר הצבעים (לא כולל שחור ולבן), הערכים של לפחות שני צבעים מתוך הצבעים אדום, ירוק וכחול צריכים להיות גדולים מ-0.

לקבלת נקודה שהצבע שלה הוא שחור, ערכי הצבעים אדום, ירוק וכחול צריכים להיות 0.

לקבלת נקודה שהצבע שלה הוא לבן, ערכי הצבעים אדום, ירוק וכחול צריכים להיות 255.

נתונה המחלקה Pixel - פיקסל, ולה שלוש תכונות:

- red - צבע אדום, מספר שלם מ-0 עד 255
- green - צבע ירוק, מספר שלם מ-0 עד 255
- blue - צבע כחול, מספר שלם מ-0 עד 255

הניחו שיש פעולות get/Set ו- set/Set בעבור תכונות המחלקה.

נתון ממשק חלקי של המחלקה Pixel:

כותרת הפעולה	תיאור הפעולה
public Pixel (int red, int green, int blue)	פעולה בונה המקבלת ערכים בעבור תכונות המחלקה
Java - public boolean isRed () C# - public bool IsRed ()	פעולה המחזירה true אם הנקודה היא בצבע אדום, אחרת היא מחזירה false
Java - public boolean isGreen () C# - public bool IsGreen ()	פעולה המחזירה true אם הנקודה היא בצבע ירוק, אחרת היא מחזירה false
Java - public boolean isBlue () C# - public bool IsBlue ()	פעולה המחזירה true אם הנקודה היא בצבע כחול, אחרת היא מחזירה false

(שימו לב: המשך השאלה בעמוד הבא.)

/המשך בעמוד 5/

השאלות

בשאלון זה שלושה פרקים.

יש לבחור בחמש שאלות מתוך שלושת הפרקים. לכל שאלה – 20 נקודות.
רשום על הכריכה החיצונית של המחברת את שם המסלול שלמדת, אם תבחר לענות
על שאלות מן הפרק השלישי, ואם לא – רשום "ללא מסלול".

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפתורים בשפת Java: בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה:

`Scanner input = new Scanner (System.in);`

פרק ראשון

כתוב פעולה חיצונית ששמה `filter` בשפת Java או `Filter` בשפת C# המקבלת מערך מטיפוס שלם – `arr` ומספר מטיפוס שלם – `num`. על הפעולה להחזיר מערך חדש מטיפוס שלם, הכולל רק את המספרים המופיעים במערך `arr` שאינם שווים ל-`num`.

לדוגמה: בעבור `num` ששווה ל-9 ומערך `arr` בגודל 7 שלפניך:

	0	1	2	3	4	5	6
arr	6	9	2	2	9	4	-3

הפעולה תחזיר מערך בגודל 5, הנראה כך:

	0	1	2	3	4
	6	2	2	4	-3

הנח שקיים במערך `arr` מספר אחד לפחות שאינו שווה ל-`num` ומספר אחד לפחות ששווה ל-`num`.

שים לב: גודל המערך המוחזר הוא כמספר האיברים במערך `arr` שאינם שווים ל-`num`.

הערה: אין חשיבות לסדר המספרים במערך המוחזר.

/המשך בעמוד 4/

א. ממשו את הפעולה `isRed/IsRed`.

הניחו שערכי התכונות תקינים (מ-0 עד 255).

ב. בסעיף זה אפשר להשתמש בפעולות `isGreen / IsGreen` ו- `isBlue / IsBlue` בלי לממש אותן. אפשר להוסיף

פעולות נוספות למחלקה `Pixel`, אך חובה לממש אותן.

נתונה המחלקה `Structure` – ולה תכונה אחת בלבד:

`arr` – מערך חד-ממדי מטיפוס `Pixel` בגודל `N`.

(1) כתבו פעולה פנימית במחלקה `Structure` ששמה `isBalanced` בשפת `Java` או `IsBalanced` בשפת `C#`.

הפעולה תחזיר `true` אם במערך `arr` מספר הנקודות בצבע האדום זהה למספר הנקודות בצבע הירוק וזהה

למספר הנקודות בצבע הכחול. אחרת הפעולה תחזיר `false`.

שימו לב: הפעולה תחזיר `true` גם אם יש נקודות בצבעים שאינם אדום, ירוק וכחול, כל עוד מספרי הנקודות האדומות, הירוקות והכחולות זהים.

(2) מערך "שחור לבן" הוא מערך שבו כל הנקודות הן רק בצבעים שחור ולבן, ויש בו לפחות נקודה אחת בצבע

לבן ולפחות נקודה אחת בצבע שחור.

כתבו פעולה פנימית במחלקה `Structure` ששמה `isBlackWhite` בשפת `Java` או `IsBlackWhite`

בשפת `C#`. הפעולה תחזיר `true` אם המערך `arr` הוא מערך "שחור לבן", אחרת היא תחזיר `false`.

לפותרים בשפת C#

נתונה המחלקה **Subject**, המייצגת מקצוע בתעודה של תלמיד, ולה שתי תכונות: שם המקצוע – **subName** והציון – **grade** (בין 0 ל-100).

```
public class Subject {
    private string subName;
    private int grade;
}
```

נוסף על כך, נתונה המחלקה **ReportCard**, המייצגת תעודה של תלמיד, ולה שתי תכונות: שם התלמיד – **stuName** ומערך של עצמים – **subArray** מטיפוס **Subject** בגודל מספר המקצועות שלמד התלמיד.

```
public class ReportCard {
    private string stuName;
    private Subject [] subArray;
}
```

הנח שקיימות פעולות **Get** ו-**Set** בעבור כל התכונות בשתי המחלקות.

לפניך ממשק חלקי של המחלקה **ReportCard**.

תיאור הפעולה	כותרת הפעולה
פעולה בונה המקבלת את שם התלמיד ואת מספר המקצועות שלמד התלמיד, ומאתחלת את תכונות המחלקה. המערך subArray מאותחל למערך ריק בגודל num .	public ReportCard (string name, int num)
פעולה המחזירה את ממוצע כל הציונים בתעודה של התלמיד.	public double Average ()
פעולה המחזירה true אם התלמיד נחשב לתלמיד מצטיין, אחרת היא מחזירה false . תלמיד מצטיין: תלמיד שממוצע ציוניו הוא 85 ומעלה, כל ציוניו מעל 54, ולפחות אחד מציוניו הוא 100.	public bool IsExcellent ()

(שים לב: המשך השאלה בעמוד הבא.)

/המשך בעמוד 7/

- א. ממש את הפעולה הבונה של המחלקה **ReportCard**.
- ב. ממש את הפעולה **IsExcellent** במחלקה **ReportCard**.
אפשר להשתמש בפעולה **Average** של המחלקה **ReportCard** בלי לממש אותה.
הנח שאיברי המערך **subArray** שונים מ-**null**.
- ג. כתוב פעולה חיצונית ששמה **PrintExcellent** המקבלת מערך של עצמים – **array** מטיפוס **ReportCard**, המייצג תלמידים בכיתה מסוימת. הפעולה תדפיס את שמות התלמידים המצטיינים.
חובה להשתמש בפעולה **IsExcellent** של המחלקה **ReportCard**.
הנח שאיברי המערך **array** שונים מ-**null**.

לפותרים בשפת C#

"מחרוזת מיוחדת" היא מחרוזת שבה התווים הזהים מופיעים ברצף או מחרוזת ללא תווים זהים (כולל מחרוזת ריקה).

דוגמאות ל"מחרוזת מיוחדת": "1zz**", "&\$**", "a", "aaaa", "ccabbb".

דוגמאות למחרוזות שאינן "מחרוזת מיוחדת": "3%33**", "abcb", "aba".

נתונה המחלקה **MyString**.

```
public class MyString {
    private string str;
}
```

לפניך ממשק המחלקה **MyString**.

יש להשתמש בפעולות הממשק ללא צורך לממש אותן.

שם הפעולה	כותרת הפעולה
פעולה הבונה מחרוזת ריקה.	<code>public MyString ()</code>
פעולה המחזירה את מספר המופעים של התו <code>ch</code> במחרוזת <code>str</code> .	<code>public int CountChar (char ch)</code>
פעולה המקבלת תו <code>ch</code> ומוחקת מן המחרוזת <code>str</code> את כל המופעים שלו.	<code>public void RemoveChar (char ch)</code>
פעולה המוסיפה לסוף המחרוזת <code>str</code> את התו <code>ch</code> .	<code>public void AppendChar (char ch)</code>
פעולה המחזירה את התו הראשון במחרוזת <code>str</code> .	<code>public char FirstChar ()</code>
פעולה המחזירה <code>true</code> אם המחרוזת <code>str</code> ריקה, אחרת היא מחזירה <code>false</code> .	<code>public bool IsEmpty ()</code>

(שים לב: המשך השאלה בעמוד הבא.)

לפניך כותרת של הפעולה החיצונית Special :

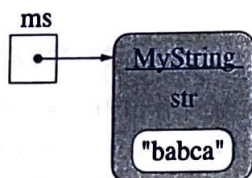
public static MyString Special (MyString ms)

הפעולה Special מחזירה עצם חדש מטיפוס MyString שהתכונה שלו, str, היא "מחרוזת מיוחדת" המורכבת מכל התווים שבתכונה str של העצם ms.

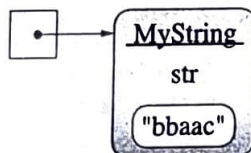
ממש את הפעולה Special באמצעות פעולות ממשק המחלקה MyString בלבד. אין להוסיף פעולות, גם לא פעולות Set ו-Get, למחלקה MyString.

לדוגמה:

בעבור העצם ms שלפניך, שבו מחרוזת רגילה:



הפעולה Special תחזיר עצם חדש, שבו "מחרוזת מיוחדת":



הערות:

- אין צורך לשמור על המחרוזת str בעצם ms.
- אין חשיבות לסדר התווים בעצם החדש, כל עוד נשמרת ההגדרה של "מחרוזת מיוחדת".

השאלות

בשאלון זה שלושה פרקים.

יש לבחור בחמש שאלות מתוך שלושת הפרקים. לכל שאלה – 20 נקודות.
רשום על הכריכה החיצונית של המחברת את שם המסלול שלמדת, אם תבחר לענות על שאלות מן הפרק השלישי, ואם לא – רשום "ללא מסלול".

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפתורים בשפת Java: בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה:

`Scanner input = new Scanner (System.in);`

פרק ראשון

1. "מחרוזת כפולה" היא מחרוזת שאינה ריקה המורכבת משתי תתי-מחרוזות זהות שמופיעות זו אחר זו.

דוגמאות ל"מחרוזת כפולה": "abcabc", "a@a@", "***".

דוגמאות למחרוזות שאינן "מחרוזת כפולה": "232323", "ab&ab", "abba", "a@a", "ab", "%".

כתוב פעולה חיצונית ששמה `isDouble` בשפת `Java` או `IsDouble` בשפת `C#` המקבלת מחרוזת `str` –

ומחזירה `true` אם `str` היא "מחרוזת כפולה", אחרת היא מחזירה `false`.

הערה: הנח שהמחרוזת `str` אינה ריקה.

2. זוג "סוף התחלה" הוא זוג של מספרים חיוביים ושלמים שבו הספרה הראשונה באחד מן המספרים זהה לספרה האחרונה במספר האחר (אין חשיבות לסדר המספרים).

מספר הקטן מ-10 (כלומר, שמורכב מספרה אחת בלבד) נחשב גם לספרה הראשונה וגם לספרה האחרונה.

דוגמאות לזוג "סוף התחלה":

מספר I	מספר II	מספר I	מספר II
3090	6783	25	534

נתונה המחלקה **PairOfNums** – זוג מספרים חיוביים, ולה שתי תכונות:

```
public class PairOfNums {
    private int num1 ;
    private int num2 ;
}
```

הנח שקיימות פעולות `get` ו-`set` בשפת **Java** ופעולות `Get` ו-`Set` בשפת **C#** בעבור תכונות המחלקה.

לפניך ממשק חלקי של המחלקה **PairOfNums** בשפות **Java** ו-**C#**.

אפשר להשתמש בפעולות הממשק ללא צורך לממש אותן.

תיאור הפעולה	כותרת הפעולה
פעולה בונה המקבלת ערכים בעבור תכונות המחלקה.	<code>public PairOfNums (int num1, int num2)</code>
פעולה המחזירה <code>true</code> אם העצם הוא זוג "סוף התחלה", אחרת היא מחזירה <code>false</code> .	בשפת Java <code>public boolean endStart ()</code> בשפת C# <code>public bool EndStart ()</code>

לפניך כותרת פעולה חיצונית בשפת **Java**:

```
public static PairOfNums[] generate (int n)
```

לפניך כותרת פעולה חיצונית בשפת **C#**:

```
public static PairOfNums[] Generate (int n)
```

הפעולה `generate / Generate` בונה מערך ריק של עצמים מטיפוס **PairOfNums** בגודל `n`. בעבור כל תא במערך, הפעולה מגרילה שני מספרים בין 1 ל-1,000 (כולל), ובודקת אם הם זוג "סוף התחלה". אם כן, מוכנס לתא עצם שערכי התכונות שלו הם זוג המספרים שהוגרלו. אחרת, הפעולה ממשיכה להגריל מספרים עד שמתקבל זוג "סוף התחלה", ואז מוכנס לתא עצם שערכי התכונות שלו הם זוג המספרים שהוגרלו.

לאחר שהמערך מתמלא, הפעולה מחזירה את המערך.

ממש את הפעולה.

הערה: הנח ש-`n` גדול מאפס.

/המשך בעמוד 4/

השאלות

בשאלון זה שלושה פרקים.

עליך לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפתורים בשפת Java: בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה:

Scanner input = new Scanner (System.in);

פרק ראשון (25 נקודות)

ענה על שאלה 1 – חובה (10 נקודות).

1. כתוב פעולה חיצונית `above` בשפת Java או `Above` בשפת C#, המקבלת מערך `arr` מטיפוס שלם

ומספר `num` מטיפוס שלם. הפעולה תחזיר את האינדקס הראשון של התא שסכום המספרים מתחילת המערך

עד אליו (כולל) גדול מן המספר `num`. אם אין תא כזה, תחזיר הפעולה `-1`.

דוגמה: במערך `arr` שלפניך, בעבור `num` שווה ל-9, תחזיר הפעולה את המספר 4, משום שהוא האינדקס הראשון

של התא שסכום האיברים מתחילת המערך עד אליו (כולל) גדול מן המספר 9.

0	1	2	3	4	5
3	-2	6	2	1	3

דוגמה נוספת: במערך `arr` שלפניך, בעבור `num` שווה ל-11, תחזיר הפעולה את המספר `-1`, משום שאין תא שסכום

המספרים מתחילת המערך עד אליו (כולל) גדול מן המספר 11.

0	1	2	3	4	5
1	2	-2	0	7	3

/המשך בעמוד 4/

ענה על אחת מן השאלות 2-3 (15 נקודות).

2. נתונה המחלקה **Weight** – משקל, שיש לה שתי תכונות:

kilo – מספר קילוגרמים, המקבלת ערכים מ-0 ומעלה, מטיפוס שלם.

gram – מספר גרמים, המקבלת ערכים מ-0 עד 999 (כולל), מטיפוס שלם.

הנח שלכל תכונה במחלקה הוגדרו בשפת Java פעולות get ו-set ובשפת C# פעולות Get ו-Set.

לפניך ממשק המחלקה **Weight** הכתוב בשפת Java ובשפת C#:

כותרת הפעולה	תיאור הפעולה
public Weight ()	פעולה בונה המאתחלת את התכונות kilo ו-gram לאפס.
public Weight (int kilo, int gram)	פעולה בונה המקבלת ערכים בקילוגרמים וגרמים. הנח שמספר הקילוגרמים במשתנה kilo גדול או שווה ל-0, ומספר הגרמים במשתנה gram הוא בין 0 ל-999 (כולל).
public Weight (int totalGram)	פעולה בונה המקבלת ערך בגרמים. מספר הגרמים במשתנה totalGram יכול להיות מעל 999. הפעולה תחשב ותכניס את מספר הקילוגרמים ל-kilo ואת מספר הגרמים ל-gram. <u>דוגמה:</u> אם מספר הגרמים במשתנה totalGram הוא 3,215 התכונה kilo תקבל 3 והתכונה gram תקבל 215. הנח שמספר הגרמים במשתנה totalGram גדול או שווה ל-0.
בשפת Java – public void add (Weight other) בשפת C# – public void Add (Weight other)	פעולה המקבלת עצם אחר מטיפוס Weight ומוסיפה את ערכיו למשקל של העצם הנוכחי. הנח ש-other אינו null.
בשפת Java – public boolean less (Weight other) בשפת C# – public bool Less (Weight other)	פעולה המקבלת עצם אחר מטיפוס Weight ומחזירה true אם משקל העצם הנוכחי קטן מן המשקל האחר. הנח ש-other אינו null.

(שים לב: המשך השאלה בעמוד הבא.)

/המשך בעמוד 5/

א. ממש את שתי הפעולות הבונות של המחלקה **Weight** : **public Weight (int kilo, int gram)**

ו- **public Weight (int totalGram)** (אין צורך לממש את הפעולה הבונה ללא פרמטרים).

ב. ממש את הפעולות **add** ו- **less** בשפת **Java**, או את הפעולות **Add** ו- **Less** בשפת **C#**.

ג. נתונה המחלקה **AllWeights** שיש לה תכונה אחת: **arr** – מערך מטיפוס **Weight**.

לפניך פעולה של ממשק המחלקה **AllWeights** הכתובה בשפת **Java** ובשפת **C#**:

כותרת הפעולה	תיאור הפעולה
public Weight sum ()	פעולה המחזירה עצם מטיפוס Weight השווה
public Weight Sum ()	לסכום המשקלים של איברי המערך.
	<u>הערה:</u> אין לשנות את איברי המערך.

ממש את הפעולה **sum** בשפת **Java** או את הפעולה **Sum** בשפת **C#**.

הנח שהתאים במערך אינם **null**.

הערה: תוכל להשתמש בכל אחת מן הפעולות של ממשק המחלקה **Weight**.

3. באכסניית הנוער "חלומות טובים" יש 3 קומות, 1-3. באכסניה 200 חדרים סך הכול. באכסניה שני סוגי חדרים: חדר זוגי וחדר יחיד. המחיר ללילה בחדר יחיד הוא 50 שקלים ובחדר זוגי 100 שקלים. במערכת המחשוב של האכסניה מוגדרת המחלקה **Room** – חדר, שיש לה שלוש תכונות:
- **roomNum** – מספר חדר בן שלוש ספרות, מטיפוס שלם.
 - הספרה הראשונה (משמאל) מייצגת את מספר הקומה שבה החדר נמצא.
 - לדוגמה: חדר מספר 231 נמצא בקומה מספר 2, וחדר 145 נמצא בקומה מספר 1.
 - **roomType** – סוג חדר: 1 הוא חדר יחיד, 2 הוא חדר זוגי, מטיפוס שלם.
 - **nightsReserved** – מספר הלילות שהחדר מוזמן מעכשיו, מטיפוס שלם.
 - אם **nightsReserved** שווה ל-0 – החדר פנוי, ואם הוא מעל 0 – החדר תפוס.
- הנח שלכל תכונה במחלקה הוגדרו בשפת Java פעולות **get** ו-**set** ובשפת C# פעולות **Get** ו-**Set**.
- א. כתוב פעולה **income** בשפת Java או **Income** בשפת C#, במחלקה **Room**, המחזירה את ההכנסה המתקבלת מן החדר. חישוב ההכנסה הוא מספר הלילות המוזמנים כפול מחיר החדר.
- ב. במערכת המחשוב של האכסניה מוגדרת מחלקה נוספת, **Hostel** – אכסניה, שיש לה תכונה אחת:
- allRooms** – מערך בגודל 200 מטיפוס **Room**. (החדרים במערך אינם ממוינים לפי סדר כלשהו). הנח שהתאים במערך אינם **null**.

לפניך ממשק חלקי של המחלקה **Hostel** הכתוב בשפת Java ובשפת C#:

תיאור הפעולה	כותרת הפעולה
הפעולה מוצאת את החדר הראשון במערך allRooms שעונה על התנאים האלה: פנוי ומן הסוג type . הפעולה מעדכנת בחדר את מספר הלילות המבוקש המופיע במשתנה nights ומחזירה את מספר החדר. אם אין במערך חדר פנוי מן הסוג המבוקש, הפעולה מחזירה 1 –.	בשפת Java – public int orderRoom (int type, int nights) בשפת C# – public int OrderRoom (int type, int nights)
הפעולה מחזירה מערך בגודל 3 שבו מופיעים סך כל סכומי ההכנסה מכל החדרים בכל אחת מן הקומות באכסניה. באינדקס 0 מופיע סכום ההכנסה מקומה 1, באינדקס 1 מופיע סכום ההכנסה מקומה 2, ובאינדקס 2 מופיע סכום ההכנסה מקומה 3.	בשפת Java – public int [] floorIncome () בשפת C# – public int [] FloorIncome ()

/המשך בעמוד 7/

(שים לב: המשך השאלה בעמוד הבא.)

(1) ממש את הפעולה orderRoom בשפת Java או את הפעולה OrderRoom בשפת C#, במחלקה Hostel.

(2) ממש את הפעולה floorIncome בשפת Java או את הפעולה FloorIncome בשפת C#, במחלקה Hostel.

חובה להשתמש בפעולה שכתבת בסעיף א.

השאלות

בשאלון זה שלושה פרקים.

עליך לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפותרים בשפת Java: בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה:

Scanner input = new Scanner (System.in);

פרק ראשון (35 נקודות)

ענה על שאלה 1 – חובה. (15 נקודות)

שים לב: לשאלה זו שני נוסחים: בשפת Java (עמודים 2-3) ובשפת C# (עמוד 4).

1. לפותרים בשפת Java

לפניך המחלקה עיפרון Pencil.

Pencil	
private int length	תכונה של אורך העיפרון
private boolean sharpened	תכונה שהערך שלה true אם העיפרון מחודד, אחרת הערך שלה false.
public Pencil (int length, boolean sharpened)	פעולה בונה המקבלת ערכים בעבור כל התכונות של העיפרון.
public boolean isLonger (Pencil other)	פעולה המקבלת עצם אחר מטיפוס Pencil ומחזירה true אם העיפרון הנוכחי ארוך יותר מן העיפרון האחר – other, אחרת היא מחזירה false. הנח ש- other אינו null.
public boolean isSharpened ()	פעולה המחזירה true אם העיפרון מחודד, אחרת היא מחזירה false.

(שים לב: המשך השאלה בעמוד הבא.)

/המשך בעמוד 3/

א. ממש את הפעולות isLonger ו isSharpened.

ב. לפניך קטע תוכנית הכתוב בפעולה main של המחלקה הראשית.

```
Pencil pencil1 = new Pencil (12, true);
Pencil pencil2 = new Pencil (13, false);
if (pencil1.isLonger (pencil2) && pencil1.isSharpened ())
    System.out.println ("I choose pencil1");
else {
    if (pencil2.isSharpened ())
        System.out.println ("I choose pencil2");
    else
        System.out.println ("pencil2 needs to be sharpened");
}
```

עקוב אחר ביצוע קטע התוכנית, הצג את העצמים שנוצרו, ורשום את הפלט.

/המשך בעמוד 4/

לפותרים בשפת C#

לפניך המחלקה עיפרון Pencil.

Pencil	
private int length	תכונה של אורך העיפרון
private bool sharpened.	תכונה שהערך שלה true אם העיפרון מחודד, אחרת הערך שלה false.
public Pencil (int length, bool sharpened)	פעולה בונה המקבלת ערכים בעבור כל התכונות של העיפרון.
public bool IsLonger (Pencil other)	פעולה המקבלת עצם אחר מטיפוס Pencil ומחזירה true אם העיפרון הנוכחי ארוך יותר מן העיפרון האחר – other, אחרת היא מחזירה false. הנח ש- other אינו null.
public bool IsSharpened ()	פעולה המחזירה true אם העיפרון מחודד, אחרת היא מחזירה false.

א. ממש את הפעולות IsLonger ו- IsSharpened.

ב. לפניך קטע תוכנית הכתוב בפעולה Main של המחלקה הראשית.

```
Pencil pencil1 = new Pencil (12, true);
Pencil pencil2 = new Pencil (13, false);
if (pencil1.IsLonger (pencil2) && pencil1.IsSharpened ())
    Console.WriteLine ("I choose pencil1");
else {
    if (pencil2.IsSharpened ())
        Console.WriteLine ("I choose pencil2");
    else
        Console.WriteLine ("pencil2 need to be sharpened");
}
```

עקוב אחר ביצוע קטע התוכנית, הצג את העצמים שנוצרו, ורשום את הפלט.

/המשך בעמוד 5/

ענה על אחת מן השאלות 2-3. (20 נקודות)

2. נתונה המחלקה **Doctor** – רופא, ולה שלוש תכונות:

- name – שם הרופא, מטיפוס מחרוזת.
- specialization – תחום ההתמחות, מטיפוס מחרוזת.
- rate – דירוג הרופא, מטיפוס מספר ממשי, שמקבל ערכים מ-0 עד 10.

הנח שלכל תכונה במחלקה **Doctor** הוגדרו בשפת Java הפעולות `get` ו-`set` ובשפת C# הפעולות `Get` ו-`Set`.

א. כתוב פעולה בונה המקבלת ערכים בעבור שם הרופא ותחום ההתמחות ומאתחלת את התכונה `rate` ל-0.

ב. דירוג (`rate`) של רופא מחושב לפי ממוצע ההערכות של המטופלים שלו. הערכה היא מספר שלם בין 0 ל-10 (כולל).

כתוב פעולה `void` חיצונית בשפת Java בשם `doctorRating`, או בשפת C# בשם `DoctorRating`, שתקבל

מערך בשם `doctors` מטיפוס **Doctor**. הנח שערך הדירוג – `rate` של איברי המערך הוא 0.

הפעולה תקלוט, לפי סדר האיברים במערך, את כל ההערכות שנתנו המטופלים לכל רופא.

הקלט בעבור כל רופא מסתיים בקליטת הערך (-1), ולאחר הערך (-1) מתחיל הקלט של הרופא הבא במערך,

עד אחרון הרופאים במערך (המסתיים גם הוא בקליטת הערך (-1)).

הפעולה תעדכן את הדירוג של כל רופא לפי חישוב ממוצע ההערכות של המטופלים שלו.

לדוגמה: בעבור מערך – `doctors` בגודל 3 התקבל הקלט הזה:

-1, 6, 7, -1, 2, -1, 8, 9, 7, 10 (משמאל לימין).

הפעולה תעדכן את הדירוג של הרופא הראשון במערך (באינדקס 0) ל-8.5, את הדירוג של הרופא השני ל-2

ואת הדירוג של הרופא השלישי ל-6.5.

הנח שיש לפחות הערכה אחת בעבור כל רופא במערך.

הנח שכל האיברים במערך `doctors` שונים מ-`null`.

אין צורך לבדוק את תקינות הקלט.

3. איבר במערך חד-ממדי ייקרא איבר שוויון אם סכום האיברים במערך מצד ימין שלו שווה לסכום האיברים במערך מצד שמאל שלו.

לאיבר שנמצא בקצה המערך אין איברים מצד ימין או מצד שמאל שלו, ובמקרה זה הסכום שבצד שבו אין איברים נחשב ל-0.

לדוגמה: במערך החד-ממדי בגודל 6 שלפניו האיבר המסומן הוא איבר שוויון, כי סכום האיברים משני צדדיו שווה ל-6.

-1	7	4	2	3	1
----	---	---	---	---	---

דוגמה נוספת: במערך החד-ממדי בגודל 7 שלפניו האיבר המסומן הוא איבר שוויון, כי סכום האיברים משני צדדיו שווה ל-0.

6	-3	4	2	0	-6	3
---	----	---	---	---	----	---

א. כתוב בשפת Java או בשפת C# פעולה חיצונית שתקבל:

— מערך חד-ממדי מטיפוס שלם.

— מספר המציין מקום של איבר במערך (אינדקס), מטיפוס שלם.

הנח שהמספר המתקבל תקין.

הפעולה תבדוק אם איבר זה הוא איבר שוויון.

אם כן — תחזיר הפעולה true, אחרת היא תחזיר false.

ב. מערך חד-ממדי ייקרא מערך שוויוני אם במערך יש לפחות איבר אחד שהוא איבר שוויון.

כתוב בשפת Java או בשפת C# פעולה חיצונית שתקבל מערך חד-ממדי מטיפוס שלם.

הפעולה תבדוק אם מערך זה הוא מערך שוויוני.

אם כן — תחזיר הפעולה true, אחרת היא תחזיר false.

הערה: אפשר להשתמש בפעולה שכתבת בסעיף א.

השאלות

בשאלון זה שלושה פרקים.

עליך לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

פרק ראשון (25 נקודות)

זערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפתרים ב-Java: בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה:

Scanner input = new Scanner (System.in);

ונה על שאלה 1 – חובה (10 נקודות)

1. כתוב פעולה חיצונית exact ב-Java או Exact ב-C#, המקבלת מערך – arr מטיפוס מחרוזות, ומספר שלם – num. הפעולה תחזיר את מספר המחרוזות במערך arr, שאורכן שווה ל- num.

ונה על אחת מן השאלות 2-3 (15 נקודות).

במערכת מחשוב של חנות למוצרי מחנות יש מחלקה בשם Flashlight שיש לה שתי תכונות:

model – שם דגם של פנס מטיפוס מחרוזות.

price – מחיר הפנס, מספר מטיפוס ממשי.

א. כתוב ב-Java או ב-C# את כותרת המחלקה Flashlight ואת התכונות שלה, וכתוב פעולה בונה המקבלת

ערכים עבור דגם הפנס ומחיר הפנס.

ב. (1) לקראת טיול עליך להצטייד בשלושה פנסים שונים. כתוב פעולה חיצונית threeFlashlights ב-Java או

ThreeFlashlights ב-C#, המקבלת מערך – arr של עצמים שונים מטיפוס Flashlight,

ומספר – total מטיפוס ממשי.

על הפעולה למצוא ולהדפיס פעם אחת בלבד את הדגמים של שלושת הפנסים שסכום המחירים שלהם הוא המספר total.

הנח שלכל תכונה במחלקה Flashlight הוגדרו ב-Java פעולות get ו-set וב-C# פעולות Get ו-Set.

הנח שיש שלשה אחת בלבד של פנסים שסכום המחירים שלהם הוא המספר total.

(2) מהי סיבוכיות זמן הריצה של הפעולה שכתבת בתת-סעיף ב (1)? נמק.

3. במערכת מחשוב של חברה למכירת מכוניות משומשות הוגדרה מחלקה בשם `Car` שיש לה שלוש תכונות:
- `licenseNum` – מספר רישוי מטיפוס מחרוזת.
 - `hadAccident` – משתנה מטיפוס בוליאני המקבל `true` אם המכונית עברה תאונה, אחרת – מקבל `false`.
 - `price` – מחיר המכונית הוא מספר מטיפוס שלם.
- הנח שהוגדרה למחלקה פעולה בונה המקבלת ערכים עבור כל אחת מן התכונות של המחלקה, ולכל תכונה הוגדרו ב-`Java` פעולות `get` ו-`set` וב-`C#` פעולות `Get` ו-`Set`.
- א. כתוב במחלקה `Car` פעולה פנימית בוליאנית בשם `range` ב-`Java` או `Range` ב-`C#`, המקבלת שני מספרים מטיפוס שלם `min` ו-`max`. הפעולה תחזיר `true` אם מחיר המכונית הוא בין `min` ל-`max` (כולל), אחרת תחזיר הפעולה `false`.
- הנח ש-`max` גדול מ-`min`.
- ב. נתונה המחלקה `AllCars` שיש לה שתי תכונות:
- `cars` – מערך חד ממדי מטיפוס `Car`.
 - `num` – מספר המכוניות הנוכחי שיש בחברה, מטיפוס שלם.
- לפניך הפעולה הבונה של המחלקה:

```
public AllCars (int max)
{
    this.cars = new Car [max];
    this.num = 0;
}
```

- (1) כתוב במחלקה `AllCars` פעולה פנימית בוליאנית בשם `addCar` ב-`Java` או `AddCar` ב-`C#`, המקבלת עצם מטיפוס `Car`. הפעולה מוסיפה אותו למערך המכוניות במקום הראשון הפנוי ומחזירה `true`, אם המערך מלא תחזיר הפעולה `false`.
- הנח שאין מקומות פנויים מהתחלת המערך ועד למספר `num` (לא כולל `num`).
- (2) כתוב במחלקה `AllCars` פעולה פנימית בשם `print` ב-`Java` או `Print` ב-`C#`, המקבלת שני מספרים `min` ו-`max` מטיפוס שלם (`max` גדול מ-`min`). הפעולה תדפיס את מספר הרישוי של כל מכונית במערך `cars` שלא עברה תאונה ושמחירה נמצא בין שני המספרים `min` ו-`max` (כולל).
- חובה להשתמש בפעולה שהוגדרה בסעיף א.

פרק שני (50 נקודות)

שים לב: בכל שאלה שנדרש בה מימוש אתה יכול להשתמש בפעולות של המחלקות: תור, מחסנית, עץ בינרי וחוליה, בלי לממש אותן. אם אתה משתמש בפעולות נוספות, עליך לממש אותן.

ענה על שתיים מן השאלות 4-6 (לכל שאלה – 25 נקודות).

שים לב: לשאלה זו שני נוסחים: ב-Java (עמודים 4-5) וב-C# (עמודים 6-7). פתור לפי מה שלמדת.

לפתורים בשפת Java

לפניך הפעולה secret1:

```
public static boolean secret1 (int num, int digit)
```

```
{
    if (num < 10)
        return ( num % 2 == digit % 2 );
    if ( num % 2 != digit % 2 )
        return false;
    return secret1 ( num / 10, digit );
}
```

- א. (1) כתוב מה יחזיר זימון הפעולה secret1 (937, 5). עליך להראות מעקב.
- (2) הבא דוגמה למספר num בעל 3 ספרות, שעבורו זימון הפעולה secret1(num, 5) יחזיר ערך שונה ממה שהתקבל בתת-סעיף א (1). עליך להראות מעקב.
- (3) כתוב במשפט אחד מה מבצעת הפעולה הבוליאנית secret1, כלומר מהי השאלה שעליה הפעולה מחזירה true או false.

(שים לב: המשך השאלה בעמוד הבא.)

/המשך בעמוד 5/

השאלות

בשאלון זה שלושה פרקים.

עליך לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

פרק ראשון (25 נקודות)

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפתור ב-Java: בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה:

Scanner input = new Scanner (System.in);

ענה על שאלה 1 – חובה (10 נקודות)

1. נתונה המחלקה **AllNumbers** שיש לה תכונה אחת: מערך חד-ממדי **arrayNum**, מטיפוס שלם.

במערך יש מספרים חיוביים הגדולים מאפס, שחלקם מספרים זוגיים וחלקם מספרים אי-זוגיים.

כתוב פעולה פנימית **lastOddValue** ב-Java או **LastOddValue** ב-C# שתחזיר את הערך של המספר האי-זוגי האחרון במערך.

לדוגמה: בעבור המערך **arrayNum** בגודל 6 שלפניך, תחזיר הפעולה את המספר 3, שהוא הערך של המספר האי-זוגי האחרון במערך.

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

הנח שבמערך יש לפחות מספר אי-זוגי אחד.

/המשך בעמוד 3/

ענה על אחת מן השאלות 2-3 (15 נקודות).

2.

במצעד הפזמונים מתחרים 40 שירים שמספרם מ-1 ועד 40. 25 שופטים מדרגים את השירים.

כל שופט בוחר את שלשת השירים הטובים ביותר לדעתו. לשיר במקום הראשון (הטוב ביותר) הוא מעניק 7 נקודות,

לשיר במקום השני הוא מעניק 5 נקודות, ולשיר שבמקום השלישי הוא מעניק נקודה אחת.

השיר שיצבור את מספר הנקודות הגבוה ביותר מכל השופטים יזכה במקום הראשון.

לקראת מצעד הפזמונים הוגדרה מחלקה `Vote`.

למחלקה שלוש תכונות:

first — מספר השיר שהשופט דירג במקום הראשון, מטיפוס שלם.

second — מספר השיר שהשופט דירג במקום השני, מטיפוס שלם.

third — מספר השיר שהשופט דירג במקום השלישי, מטיפוס שלם.

הנח לשכל תכונה הוגדרו ב-Java פעולות `get` ו-`set` וב-C# פעולות `Get` ו-`Set`.

כתוב פעולה חיצונית `theWinner` ב-Java או `TheWinner` ב-C#, המקבלת מערך מטיפוס `Vote` של

דירוגי השופטים, ומדפיסה את מספרו של השיר הזוכה במקום הראשון.

הנח שקיים שיר זוכה אחד.

הנח שערכי המערך תקינים.

3. נתונה המחלקה **Time** שיש לה שתי תכונות:

hour – מייצגת שעה בין 0 ל- 23 כולל, מטיפוס שלם.

minute – מייצגת דקה בין 0 ל- 59 כולל, מטיפוס שלם.

הנח שלכל תכונה הוגדרו ב- Java פעולות get ו- set וב- C# פעולות Get ו- Set.

א. כתוב ב- Java או ב- C# במחלקה **Time** פעולה בונה שתקבל ערכים בעבור כל תכונה.

עבור כל תכונה – אם הערך המתקבל אינו אפשרי לפי הגדרות התכונה, הוא יוחלף ב- 0.

ב. נתונה המחלקה **Flight** שיש לה ארבע תכונות: שם של חברת תעופה – name מטיפוס מחרוזת,

שם של ארץ יעד – destination מטיפוס מחרוזת, קוד טיסה – flightCode מטיפוס מחרוזת,

זמן טיסה – flightTime מטיפוס **Time**.

כתוב ב- Java או ב- C# את כותרת המחלקה ואת תכונות המחלקה.

ג. נתונה המחלקה **Airport** שיש לה תכונה אחת: מערך חד-ממדי – flights מטיפוס **Flight**.

הנח שלכל תכונה במחלקה **Flight** הוגדרו ב- Java פעולות get ו- set וב- C# פעולות Get ו- Set.

במחלקה **Airport** נתונה פעולה פנימית בוליאנית, isFly ב- Java ו- IsFly ב- C#, שמטרתה לבדוק אם

במערך הטיסות flights יש טיסה של חברת התעופה "Sky". הפעולה מחזירה true אם במערך flights יש

טיסה של חברת התעופה "Sky". אחרת – הפעולה מחזירה false.

הנח שכל התאים במערך שונים מ- null.

הפעולה הנתונה שלפניך (isFly ב- Java ו- IsFly ב- C#) שגויה.

```

Java
public boolean isFly()
{
    for (int i = 0; i < this.flights.length; i++)
    {
        if (this.flights[i].getName().equals("Sky"))
            return true;
        return false;
    }
}

```

```

C#
public bool IsFly()
{
    for (int i = 0; i < this.flights.Length; i++)
    {
        if (this.flights[i].GetName() == "Sky")
            return true;
        return false;
    }
}

```

(שים לב: התתי-סעיפים של סעיף ג בעמוד הבא.)

/המשך בעמוד 5/

(1) כתוב מה תחזיר הפעולה השניה הנתונה עבור מערך flights בגודל 4, שערכי תכונת name של העצמים במערך

הם:

במקום הראשון: "Cloud",

במקום השני: "Air",

במקום השלישי: "Sky",

במקום הרביעי: "Travel".

(2) הסבר מהי השגיאה בפעולה.

(3) תקן את הפעולה כדי שהיא תבצע את הנדרש, והעתק למחברתך את הפעולה המתוקנת.

/המשך בעמוד 6/

השאלות

בשאלון זה שלושה פרקים.

עליך לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

פרק ראשון (25 נקודות)

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפתורים ב-Java: בכל שאלה שנדרשת בה קליטה, הנח שבתכנית כתובה ההוראה:

`Scanner input = new Scanner (System.in);`

ענה על שאלה 1 – חובה (10 נקודות)

1. כתוב פעולה חיצונית `big` ב-Java או `Big` ב-C# שתקבל מערך של מספרים שלמים. הפעולה תחזיר את המציין (האינדקס) של האיבר שהערך שלו הוא הגדול ביותר במערך. אם המספר מופיע יותר מפעם אחת במערך, הפעולה תחזיר את המציין (האינדקס) הקטן ביותר מבין המופעים.

ענה על אחת מן השאלות 2-3 (לשאלה – 15 נקודות).

2. בהיערכות לקראת משחקי ספורט בין-לאומיים הגדירו מחלקה `Game` שיש לה שלוש תכונות:

שם המשחק – `gameName` מטיפוס מחרוזת, מספר השחקנים – `numPlayers` מטיפוס

שלם, אם המשחק מתקיים במים – `isWater` מטיפוס בוליאני.

הנח שלכל תכונה הוגדרו ב-Java פעולות `get` ו-`set` וב-C# פעולות `Get` ו-`Set`.

א. כתוב ב-Java או ב-C# במחלקה `Game` פעולה בונה שתקבל ערכים בעבור כל תכונה.

ב. הגדירו מחלקה מדינה – `Country` שיש לה שתי תכונות:

שם המדינה – `countryName` מטיפוס מחרוזת, מערך חד-ממדי – `games` בגודל 43

מטיפוס `Game`. המערך מכיל את המשחקים שהמדינה משתתפת בהם.

הנח שלכל תכונה הוגדרו ב-Java פעולות `get` ו-`set` וב-C# פעולות `Get` ו-`Set`.

(1) כתוב ב-Java או ב-C# את כותרת המחלקה `Country` ואת התכונות שלה.

(2) כתוב ב-Java או ב-C# במחלקה `Country` פעולה בונה שתקבל את שם המדינה.

הפעולה תבנה עצם בעבור מדינה שאינה משתתפת באף לא משחק אחד.

(3) כתוב ב-Java או ב-C# במחלקה `Country` פעולה בוליאנית שתקבל את

שם המשחק ותחזיר `true` – אם המדינה משתתפת במשחק, אחרת – הפעולה

תחזיר `false`.

/המשך בעמוד 3/

השאלות

בשאלון זה שלושה פרקים.

עליך לענות על שאלות משלושת הפרקים, לפי ההוראות בכל פרק.

פרק ראשון (25 נקודות)

הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.

לפותרים ב-Java: בכל שאלה שנדרשת בה קליטה, הנח שבתכנית כתובה ההוראה:

`Scanner input = new Scanner (System.in);`

ענה על שאלה 1 – חובה (10 נקודות)

1. נתונה מחלקה תלמיד – `Student` שיש לה שתי תכונות: שם התלמיד – `name` מטיפוס מחרוזת, ומערך חד-ממדי `arrTest` בגודל 3 מטיפוס שלם. בכל תא במערך מאוחסן ציון של התלמיד במבחן.
כתוב ב-Java או ב-C# פעולה במחלקה `Student` שתחזיר את הציון הממוצע של התלמיד בשלושת המבחנים.
הערה: אין צורך לבדוק את תקינות הנתונים במערך.

/המשך בעמוד 3/

ענה על אחת מהשאלות 2 - 3 (15 נקודות)

2. נתונה מחלקה שחקן – Actor שיש לה שלוש תכונות: מספר תעודת זהות – id מטיפוס מחרוזת, מין השחקן – gender מטיפוס מחרוזת ("F" מייצג נקבה, "M" מייצג זכר), מספר הסרטים שהשחקן השתתף בהם – numFilms מטיפוס שלם. לפיכך מממש המחלקה Actor הכתוב ב-Java וב-C#.

כותרת הפעולה ב-Java	תיאור הפעולה
<code>public Actor(String id, String gender, int numFilms)</code>	פעולה הבונה שחקן שמספר תעודת הזהות שלו id, מינו gender, מספר הסרטים שהשתתף בהם numFilms.
<code>public void addFilm()</code>	פעולה המוסיפה 1 למספר הסרטים שהשחקן השתתף בהם.
<code>public int compare(Actor other)</code>	פעולה שמחזירה: 1 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם גדול ממספר הסרטים שהשחקן other השתתף בהם. 2 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם קטן ממספר הסרטים שהשחקן other השתתף בהם. 3 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם זהה למספר הסרטים שהשחקן other השתתף בהם.

(שים לב: המשך השאלה בעמוד הבא.)

/המשך בעמוד 4/

תיאור הפעולה	כותרת הפעולה ב-C#
פעולה הבונה שחקן שמספר תעודת הזהות שלו id, מינו gender, מספר הסרטים שהשתתף בהם numFilms.	public Actor(string id, string gender, int numFilms)
פעולה המוסיפה 1 למספר הסרטים שהשחקן השתתף בהם.	public void AddFilm()
פעולה שמחזירה: 1 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם גדול ממספר הסרטים שהשחקן other השתתף בהם. 2 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם קטן ממספר הסרטים שהשחקן other השתתף בהם. 3 – אם מספר הסרטים שהשחקן הנוכחי השתתף בהם זהה למספר הסרטים שהשחקן other השתתף בהם.	public int Compare(Actor other)

הנח שלכל תכונה הוגדרו ב-Java פעולות get ו-set וב-C# פעולות Get ו-Set.

א. ממש את הפעולה הבונה של המחלקה Actor.

ב-Java את הפעולה הבונה:

public Actor(String id, String gender, int numFilms)

ב-C# את הפעולה הבונה:

public Actor(string id, string gender, int numFilms)

ב. ממש ב-Java את הפעולה compare או ב-C# את הפעולה Compare.

ג. בפעולה הראשית במחלקה Program הוגדר מערך חד-ממדי actArr בגודל 37

מטיפוס Actor.

כתוב ב-Java או ב-C# פעולה במחלקה Program שתקבל מערך חד-ממדי מטיפוס

Actor ומספר num שלם וגדול מ-0.

הפעולה תדפיס את מספר השחקנים במערך שהשתתפו ביותב מ-num סרטים.

/המשך בעמוד 5/

3.

נתונה מחלקה תכנית טלוויזיה – **TvProgram** שיש לה שלוש תכונות:

קוד התכנית – **code** מטיפוס שלם, היום בשבוע שבו משודרת התכנית – **day** מטיפוס שלם, המקבל ערכים מ-1 עד 7 (כולל), האם זו תכנית ספורט – **isSport** מטיפוס בוליאני, שערכה **true** אם זו תכנית ספורט, אחרת – **false**.

הנח שלכל תכונה הוגדרו ב-**Java** פעולות **get** ו-**set** וב-**C#** פעולות **Get** ו-**Set**.

א. כתוב ב-**Java** או ב-**C#** פעולה בונה במחלקה **TvProgram** שתקבל ערכים לכל תכונה.

ב. נתונה מחלקה תכניות השבוע – **TvWeek** שיש לה שתי תכונות:

מערך חד-ממדי – **artProg** בגודל 100 מטיפוס **TvProgram**,

מספר התכניות הנוכחי בשבוע – **current** מטיפוס שלם, שערכו קטן מגודל המערך.

הנח שלכל תכונה הוגדרו ב-**Java** פעולות **get** ו-**set** וב-**C#** פעולות **Get** ו-**Set**.

(1) כתוב ב-**Java** או ב-**C#** את כותרת המחלקה **TvWeek** ואת התכונות שלה.

(2) כתוב ב-**Java** או ב-**C#** במחלקה **TvWeek** פעולה שתקבל תכנית **TvProgram**,

תוסיף אותה לתכניות השבוע ותעדכן את מספר התכניות הנוכחי בשבוע.

הנח שיש מקום להוסיף את התכנית.

(3) כתוב ב-**Java** או ב-**C#**, במחלקה **TvWeek** פעולה שתחזיר את

מספר תכניות הספורט שיש בשבוע בטלוויזיה.

/המשך בעמוד 6/

עבודת קיץ חזרה בנושא – מערך כאוסף

יש לפתור את כל השאלות במחשב.

לפניכם 5 בעיות עליכם לענות על כולם.

כל השאלות חייבות לעבור קומפילציה ובדיקת פלטים .

בשבוע הראשון של תחילת בשנה יערך מבחן בנושא זה.

הערות חשובות:

**** במקום שיש מערך כמאפיין במחלקה, יש להוסיף מאפיין שהוא
LastPosition**

**** בכל בעיה חובה ליצור תכנית ראשית .**

ליצור אובייקטים לכל המחלקות ולהריץ את התוכנית

**** יש לשמור את הפרויקטים ב diskonkey כל פרויקט בנפרד בשם
משמעותי**

**** יש להדפיס את קבצי הפתרון של כל בעיה וכן את הבעיה עצמה
ולשים זאת בקלסר.**

**** חובה לכתוב תיעוד ליד כל מאפיין וכן הסבר ליד כל פעולה פנימית
מה היא מבצעת**

וכן טענת כניסה וטענת יציאה ליד כל פעולה חיצונית.

**** חשוב לציין כי חלק מהשאלות נפתרו במהלך השעורים**

**בחודש האחרון ולכן עליכם לעבור על השאלות ולראות אם יש סעיפים
נוספים שלא בצעתם ולהוסיף הערות בהתאם לדרישות .**

עבודה מהנה

וחופשה טובה

יונה סעדיה טליה גיל-עד טייר

בעיה מספר 1

הנהלת המשקים

המטרה לבנות מערכת ממוחשבת שתנהל מידע אודות עצי פרי במשקים במושב.

לניהול המידע הוחלט לבנות מחלקה **עץ** , **משק מושב**.

להלן הגדרות מחלקות

מחלקה עץ

שם עץ

קוד עץ

כמות הפירות שנקטפו מעץ זה בשנה האחרונה.

מחלקה משק

שם בעל המשק

קוד בעל המשק

אוסף של עצי הפרי ממשק זה. (מקסימום 500 עצים במשק אחד)

LastPosition - מספר העצים במשק

מחלקה מושב

שם מושב

אוסף של משקים. (מקסימום 1000 משקים)

LastPosition - מספר המשקים במושב

משימות

1. כתוב את המחלקות **עץ** , **משק** , **מושב**
כל מחלקה כוללת את הפעולות בונה , בונה מעתיקה, עדכון , אחזור, הדפס.
בנוסף כתוב את הפעולות : לכל פעולה בחר באיזה מחלקה **נכון** להוסיף אותה:
הוספת עץ למשק – פעולה המקבלת **עץ** ומוסיפה אותו לעצי המשק.
מחיקת עץ למשק – פעולה המקבלת **עץ** ומוחקת אותו מהמשק.
סה"כ הפירות שנקטפו במשק – פעולה המחזירה את מספר הפירות שנקטפו מהמשק.
איתור עץ – פעולה המקבלת **עץ** בודקת אם העץ קיים באוסף העצים מחזירה את מיקומו אחרת הפעולה מחזירה 1-
More100-כתוב פעולה במחלקת משק המחזירה **את העצים** שקטפו מהם יותר מ100 פירות.

הוספת משק במושב – פעולה המקבלת משק ומוסיפה אותו למושב.
איתור משק במושב – פעולה המקבלת **משק** ומחזירה **את** מיקומו באוסף המשק אחרת הפעולה מחזירה 1-

בתוכנית הראשית

2. בנה ואתחל מושב המכיל 2 משקים שכל אחד מהם מכיל 3 עצים.
3. הדפס את פרטי כל העצים בכל המשקים במושב.
4. כתוב פעולה חיצונית המקבלת **מושב ומדפיסה את מספר המשק ושם המשק שאין להם עצי פרי**.
5. כתוב פעולה חיצונית המקבלת **מערך של 10 מושבים ועץ ומחזירה את סהכ הפירות שנקטפו המעץ מכל המשקים ביחד**.

בעיה מספר 2

מערכת בית ספרית

משרד החינוך החליט לנהל את מערכת המידע על הציונים של תלמידים בכל בתי הספר בארץ. לשם כך הוגדרו את המחלקות הבאות:

מקצוע מיוצג ע"י

- שם המקצוע
- ציון
- מס' יחידות הלימוד

תלמיד מיוצג ע"י Student

- שם התלמיד
- מספר הזיהוי של התלמיד
- אוסף מקצועות הלימוד של התלמיד (מקסימום 12 מקצועות לתלמיד)
- מספר מקצועות בפועל - LastPosition

כיתת לימוד מיוצגת ע"י StudentsClass

- שם הכיתה
- אוסף תלמידים (מקסימום 35 תלמידים בכיתה)
- מספר תלמידים בפועל - lLastPosition

מה עליכם לעשות?

ייצגו את המחלקות: כותרת ופעולות: בונה, אחזור ועדכון לפי הצורך, מתארת

- מקצוע Subject
- תלמיד Student
- כיתת לימוד StudentsClass

ממשו את הפעולות הבאות במחלקה Student:

double <u>GetAverage()</u>	הפעולה מחזירה את ממוצע ציוני התעודה של התלמיד הנחות: 5 יחידות לימוד מזכות את התלמיד ב-20 נקודות בונס ו- 4 יחידות מזכות את התלמיד ב- 10 נקודות בונס מקצועות מתמטיקה ואנגלית מזכים בבונס 25 נקודות עבור 5 יחידות ו 12.5 נקודות עבור 4 יחידות
int <u>SumPoints()</u>	הפעולה מחזירה סה"כ את מס יחידות הלימוד שהתלמיד לומד
void <u>Add Subject(Subject Sub1)</u>	הפעולה מוסיפה מקצוע לתלמיד
bool <u>IsInnocent()</u>	הפעולה מחזירה אמת אם כל הציונים שלו מעל 55 אחרת תחזיר הפעולה שקר

ממשו את הפעולות הבאות במחלקה StudentsClass

void <u>AddStudent (Student s)</u>	הפעולה מוסיפה תלמיד s לכיתה במידה שיש מקום בכיתה
void <u>RemoveStudent (Student s)</u>	הפעולה מוחקת תלמיד s מהכיתה, במידה והתלמיד לא נמצא הפעולה לא תבצע כלום
int <u>StudentExists(string id)</u>	הפעולה מקבלת תלמיד ומחזירה את מיקומו של התלמיד במערך אחרת הפעולה מחזירה -1
void <u>SortStudentsNames()</u>	הפעולה ממיינת את שמות התלמידים לפי סדר א"ב
double <u>GetAverageClass()</u>	הפעולה מחזירה את הממוצע הכיתתי
void <u>PrintBestStudent()</u>	הפעולה מדפיסה כל הפרטים של התלמידים המצטיינים בכיתה. תלמיד מצטיין - ממוצע תעודת הבגרות שלו מעל 110

int NumberInnocentInClass()

הפעולה מחזירה את מספר התלמידים
הזכאים לתעודת בגרות.
תלמיד יהיה זכאי לתעודה אם כל הציונים
שלו מעל 55

בתוכנית הראשית

1. כתבו תכנית ראשית המבצעת ומדפיסה את המשימות הבאות :

- יוצרת כיתה המכילה 3 תלמידים מאותחלים.
 - בעבור כל תלמיד סך כל מספר היחידות ואת ממוצע ציוניו.
 - 2 כתוב פעולה חיצונית המקבלת כיתת לימוד ומחזירה את ממוצע הציונים של תלמידי הכיתה.
 - 3 כתוב פעולה חיצונית המקבלת כיתת לימוד ומחזירה את מספר ת"ז של התלמיד שיש לו את הממוצע הציונים הגבוה ביותר בכיתה . אם יש יותר מאחד הפעולה תחזיר את ת"ז של התלמיד הראשון במערך .
 - 4 כתוב פעולה חיצונית המקבלת מערך של 10 כיתות לימוד.
- הפעולה תדפיס את רשימת השמות של התלמידים שלומדים מתמטיקה 5 יח"ל והציון שלהם במתמטיקה מעל 90.

בעיה מספר 3

חברת הכבלים "תקשורת לכל" מספקת שרותי לוין וכבלים לתושבים ברחבי הארץ.

לקוח בחברה Client מיוצג על ידי :

- ❖ מספר תעודת זהות
- ❖ שם לקוח
- ❖ מספר טלפון
- ❖ Channels - רשימה של הערוצים השייכים לחבילת הצפייה שרכש מקסימום לכל לקוח יש אפשרות לבחור 30 ערוצים לדוגמא :
["Disney", "Music", "EuroSport", "BBC", "Sky News"]
- ❖ LastPosition – מספר הערוצים בפועל ללקוח.

חברת הכבלים - Cabels מיוצגת על ידי :

- ❖ שם חברת הכבלים
- ❖ ArrCliens - רשימת הלקוחות - מקסימום לקוחות שהחברה יכולה לקבל הוא 1000
- ❖ LastPosition – מספר לקוחות בפועל

כתוב את המשימות הבאות:

א. כתוב את כותרת המחלקה, תכונותיה, פעולה בונה בעבור כל אחת מהמחלקות הבאות : (חובה לתעד)

- i. לקוח – **Client**
- ii. חברת כבלים – **Cabels**

ב. כתוב פעולה המקבלת שם של ערוץ **ChannelName**. הפעולה בודקת האם ערוץ בשם זה קיים באוסף של הערוצים השייכים לחבילת הצפייה שרכש לקוח מסוים. אם כן הפעולה תחזיר אמת אחרת הפעולה תחזיר שקר. תחילה רשום באיזה מחלקה תממש פעולה ולאחר מכן ממש את הפעולה.

ג. כתוב פעולה המקבלת שם של ערוץ **ChannelName** שאתו יצאה חברת הכבלים במבצע חסר תקדים. הפעולה תדפיס מבין כל לקוחות החברה רק את מספרי הטלפון של הלקוחות שהערוץ אינו נמצא בחבילת הצפייה שלהם. תחילה רשום באיזה מחלקה תממש פעולה ולאחר מכן ממש את הפעולה.

ד. בשל סכסוך בין שני מנהלים החברה התפצלה לשתי חברות והלקוחות התחלקו בין שתי החברות וכל חברה בנתה רשימת לקוחות חדשה משלה. החברות החדשות מעוניינת לדעת האם נפלה טעות וקיימים לקוחות שמנויים אצל שתי החברות החדשות. כתוב פעולה פנימית במחלקת **Cabels** המקבלת חברה **cab1** והמחזירה מערך חדש המכיל את שמות הלקוחות המופיעים אצל שתי החברות החדשות במידה ולא קיימים לקוחות שמופיעים אצל שתי החברות אז הפעולה תחזיר מערך ריק.

ה. כתוב פעולה חיצונית המקבלת חברת כבלים **Cabels** מדפיסה את ת"ז של הלקוחות שיש להם חבילה של לפחות 10 ערוצים(ניתן ורצוי לכתוב פעולת עזר פנימית במחלקה המתאימה המחשבת לכל לקוח את מספר הערוצים שיש לו בחבילה)

בעיה מספר - 4

בחברת תקשורת קיימות לכל היותר 12 מחלקות, כל מחלקה אחראית על תחום מסוים.

חברה Company

- שם החברה

- ArrDeps - מערך בגודל 12 של המחלקות.

- LastPosition – מספר המחלקות הקיימות בחברה

מחלקה Department

- Specialty – תחום ההתמחות

- ArrWorkers - אוסף העובדים השייכים למחלקה. (במחלקה מקסימום 30 עובדים)

- LastPosition – מספר העובדים במחלקה

עובד Worker

- תעודת זהות

- שם העובד

- מספר שנות לימוד

בכל הנוגע לפיטורי עובדים, מונהגת במשרד מדיניות של

"אחרון נכנס ראשון יוצא", ובהתאם למדיניות זו נשמרים נתוני העובדים בכל מחלקה.

לפניך מספר משימות :

בחר באיזה מחלקה יש לממש את הפעולות הבאות

1. רשמו כותרת מחלקה ותכונות לכל אחת מהמחלקות: Department , Worker ו-

Company.

2. כתבו פעולה במחלקה המתאימה המוסיפה עובד חדש למחלקה לפי תחום ההתמחות,

הפעולה תקבל עובד ותחום התמחות ותוסיף את העובד למחלקה המתאימה. לכל

פעולה ציינו את המחלקה המתאימה.

3. במגמה לעלות את רמת העובדים בחברה, הוחלט לפטר את כל העובדים בחברה אשר

מספר שנות הלימוד שלהם קטן מ 15. פיטורי העובדים יתבצעו בהתאם למדיניות

החברה.

" אחרון נכנס ראשון יוצא " .

4. כתבו פעולה חיצונית אשר מקבלת את החברה ועובד ומבצעת את שתי המשימות

לעיל: הוספת עובד ופיטורי עובדים.

בעיה מספר - 5

ספרייה עירונית

המטרה לבנות מערכת ממוחשבת שתנהל מידע אודות ספרים בספרייה

לניהול המידע הוחלט לבנות את המחלקות הבאות:

מחלקה ספר

שם ספר

קוד ספר

מספר עותקים מספר בספרייה

מחלקה מנוי

שם מנוי

קוד מנוי

מספר ספרייה שלמנוי מותר להשאיל

מחלקה ספרייה

שם ספרייה עירונית

אוסף ספרים (מקסימום 10000)

אוסף מנויים (מקסימום 1000)

NumbersOfMembers – מספר שלם

NumbersOfBooks - מספר שלם

1. כתוב את המחלקות מנוי, ספר, ספרייה

כל מחלקה כוללת את הפעולות בונה, בונה מעתיקה, עדכון, אחזור, הדפס.
בנוסף כתוב את הפעולות: לכל פעולה בחר באיזה מחלקה נכון להוסיף אותה:

הוספת מנוי – פעולה המקבלת מנוי ומוסיפה אותו לספרייה.

מחיקת מנוי – פעולה המקבלת קוד מנוי ומוחקת אותו מהספרייה.

הוספת ספר לספרייה – פעולה המקבלת ספר ומוסיפה אותו לספרייה.

איתור מנוי בספרייה – פעולה המקבלת קוד מנוי ומחזירה את מיקומו

במערך צאחרת הפעולה מחזירה 1-.

איתור ספר בספרייה – פעולה המקבלת שם ספר ומחזירה אמת אם הספר

קיים בספרייה אחרת היא מחזירה שקר

2. **בנה ספרייה ובה 3 מנויים 3 ספרים והדפס את שם הספרייה ואת כל המנויים הספרים.**

3. **כתוב פעולה חיצונית המשאילה ספר למנויי**

הפעולה מקבלת קוד ספר וקוד מנויי הפעולה תדפיס הודעה האם ניתן להשאיל את הספר למנויי בהתאם להגבלות.

4. **הפעולה מקבלת ספרייה ומדפיסה את מספר הספרים בספרייה.**