

פרק 7

מערך של עצמים

מערך לל 38 מ'ט - ללט מה?

מעריך משמש לשמירת מספר ערכים מאותו טיפוס בדרך המאפשרת פנייה אחידה אל כל אחד מן הערכים. ניתן להשתמש במערך גם לייצוג אוסף של עצמים, באותה דרך בה משתמשים במערך לייצוג אוסף של מספרים שלמים או אוסף של מספרים ממשיים. למשל: מערך המחברות של רוני, מערך הכבשים בדיר, או מערך המחשבים בחברה. מערך של מחרוזות אף הוא מערך של עצמים, אלא שבנייה של מחרוזת נעשית גם ללא פניה מפורשת לפעולה הבונה. הדבר אפשרי כי הטיפוס string בנוי בשפה ואינו מוגדר על ידי המשתמש.



מולג'ים חדל'ים

מעריך של עצמים – מעריך שכל איבר בו הוא עצם מטיפוס שהוגדר על ידי המשתמש.

בפרק זה נלמד:

- ✓ הצהרה על מעריך של עצמים – עמוד 146 ;
- ✓ אתחול תא במעריך של עצמים – עמוד 147 ;
- ✓ פנייה אל עצם במעריך של עצמים – עמוד 147 ;
- ✓ הדגמת אופני שימוש במעריך של עצמים – עמוד 148 ;
- ✓ סיכום שלבי הצהרה ואתחול של מעריך עצמים – עמוד 148 ;
- ✓ דגשים לסיכום הפרק – עמוד 155.



מערכת לֵא 3X N'ים

בטיפוס הנתונים מערך אנחנו משתמשים כדי לייצג כמות גדולה של נתונים שהם מאותו טיפוס. השימוש במערך מאפשר מחד, שימוש בשם כולל לאוסף הנתונים, ומאידך גישה אל כל ערך באופן נפרד על ידי שימוש במצוין. בתהליך פתרון בעיות אנו נדרשים גם לשימוש במערכים שהערכים שלהם הם מטיפוס שהוגדר על ידי המשתמש ולא רק מטיפוסים בסיסיים (int, char, ...) כפי שהכרנו עד כה. מערך שומר על דרך הייצוג שלו, התכונות שלו, ודרך הפעולה איתו גם כאשר כל נתון המאוחסן בו הוא עצם.

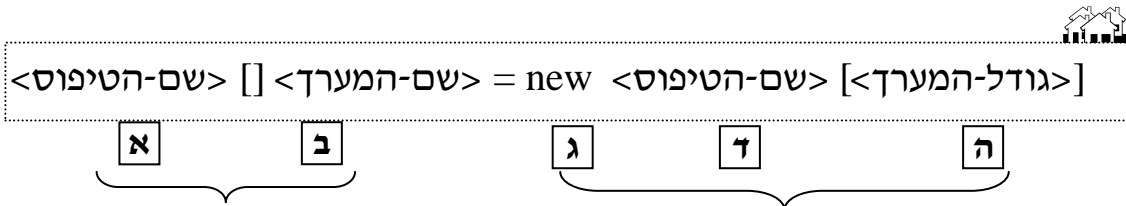
למשל: לכל כבשה בדיר יש את אותן תכונות וניתן להפעיל עליה את אותן פעולות. לכן אנחנו מגדירים את הטיפוס *כבשה*, ונוכל לבנות עצמים מן הטיפוס כך שכל אחד מהם מייצג כבשה מסויימת. בדיר יש 500 כבשים, ברור שלא נשתמש בתכנית שלנו ב- 500 משתנים מטיפוס כבשה כדי להתייחס אל כל הכבשים. כלומר אנו זקוקים למערך כדי לייצג את הכבשים.

בעת שימוש במערך של עצמים יש להתייחס אל כללי ההגדרה והפעולה של מערך, יחד עם כללי ההגדרה והפעולה של עצמים. ערך של משתנה מטיפוס שהוגדר על ידי המשתמש הוא עצם ויש להתייחס אליו בהתאם: לבנות אותו, להשים אותו במשתנה, ואז ניתן להפעיל עליו את הפעולות המוגדרות בטיפוס. אותו תהליך צריך להתבצע גם עבור כל אחד מתאי מערך של עצמים. בעבור כל תא במערך יש לבנות את העצם שאמור להיות בו, להשים את העצם במקומו במערך ואז ניתן להפעיל עליו את הפעולות המוגדרות בטיפוס שלו.

פרק זה ידגים תחילה את השימוש במערך של עצמים המבוסס על הטיפוס Student שהוצג בפרק 6 עמודים 129-135 – מערך של תלמידים. בהמשך תוצג דוגמה פתורה מלאה המתייחסת אל הטיפוס Song שהוצג בפרק 6 עמודים 141-143 – מערך של שירים.

הצהרה על מערכת לֵא 3X N'ים

הצהרה על מערך של עצמים זהה לחלוטין להצהרה על כל מערך מטיפוס בסיסי.



הסבר על מבנה ההצהרה על מערך של עצמים

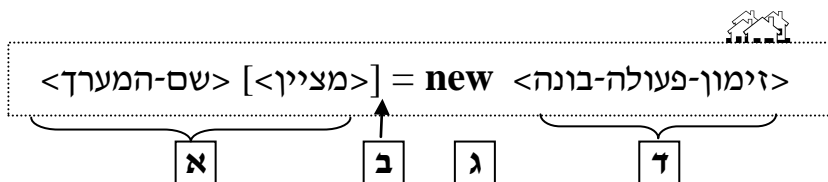
- א - מערך שערכיו הם מטיפוס <שם-הטיפוס>.
- ב - שם מערך העצמים בו יהיה שימוש במהלך התכנית.
- ג - המילה השמורה new – כמו בהצהרה על כל מערך. מביאה לבניית תאי המערך אך לא להשמת ערכים אליהם.
- ד - קביעת גודל המערך כדי שיבנו מספר תאים כ <גודל-המערך> מן הטיפוס <שם-הטיפוס>.

דוגמאות:

Student[] students = new Student[30];	בהנחה שהוגדר הטיפוס Student המתאר תלמיד, הצהרה על מערך של 30 תלמידים בשם students
Car[] cars = new Car [50];	בהנחה שהוגדר הטיפוס Car המתאר מכונית, הצהרה על מערך של 50 מכוניות בשם cars
Car[] cars; cars = new Car [50];	כמו בהצהרה על כל מערך ניתן לחלק את הצהרת המערך ובניית המערך לשני חלקים:

אתחול תא במערך לז'מנ'ס

כדי לאתחל מערך של עצמים, יש לאתחל את התאים שלו. מאחר וכל תא הוא מטיפוס שהוגדר על ידי המשתמש, אזי הערך שלו הוא עצם. כלומר בעבור כל תא של מערך העצמים, יש לבנות עצם ולהשים אותו בתוך המערך.

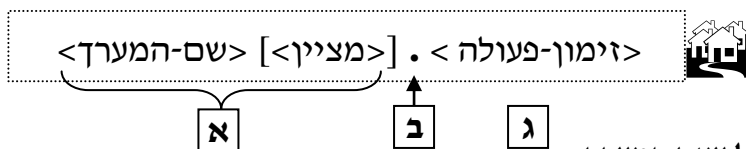


הסבר על מבנה אתחול תא במערך של עצמים

- א - פנייה לתא המערך על פי המציין הנדרש.
 - ב - סימן השמה.
 - ג - המילה השמורה new – עבור יצירת עצם חדש.
 - ד - זימון פעולה בונה עבור עצם מן הטיפוס של אברי המערך.
- בהנחה שהוגדר הטיפוס Student המתאר תלמיד, ומוגדרת בו פעולה בונה המקבלת שם ות.ז., אתחול תא המערך שמספרו 10 יכול להתבצע על ידי ההוראה:
- ```
students [10] = new Student ("Mayan", "123123123");
```
- בהנחה שהוגדר הטיפוס Car המתאר מכונית, ומוגדרת בו פעולה בונה המקבלת את סוג המכונית, את דגם המכונית ואת מספר הזיהוי שלה, אתחול תא המערך שמספרו n יכול להתבצע על ידי ההוראה:
- ```
cars [n] = new Car ("Opel", "Corsa", "9611119");
```

כני'ה אל ז'מנ'ס במערך לז'מנ'ס

כל ערך במערך הוא עצם, לכן אפשר להפעיל עליו פעולות המוגדרות בטיפוס שלו.



הסבר על מבנה הפעלת פעולה על עצם במערך

- א - פנייה לתא המערך על פי המציין הנדרש.
- ב - סימן הפעלת הפעולה.
- ג - זימון הפעולה המוגדרת בטיפוס של העצם.

דוגמה 1: בטיפוס Student מוגדרת הפעולה SetMath(int m) הקובעת את הציון במתמטיקה. ניתן להפעיל פעולה זו על עצם במערך שכל איבר בו הוא מטיפוס Student. כדי להפעיל את הפעולה על התלמיד שנמצא במקום 17 במערך, נרשום:

דוגמה 2: בטיפוס Car המתאר מכונית, מוגדרת הפעולה string GetKind() המחזירה את סוג הרכב, אזי אפשר להפעיל אותה על עצם במערך. כדי להדפיס את סוג המכונית הנמצאת במקום p במערך נרשום:

הדגמת אופן לימוד האצטרק של צ'פ'ס

אחרי ביצוע שלושת השלבים של הצהרה ואתחול, ניתן לבצע על כל אחד מן העצמים את כל אחת מן הפעולות המוגדרת בטיפוס שלו. להלן מוצגות מספר דוגמאות לשימוש במערך של עצמים המודגמות על מערך תלמידים:

קוד התכנית	הסבר קטע התכנית
<pre>int grade; for (int i=0 ; i<students.Length ; i++) { Console.WriteLine("Enter student " +(i+1) +"grade in Math:"); grade = int.Parse(Console.ReadLine()); students[i].SetMath(grade); }</pre>	קליטה ועדכון של ציוני התלמידים במתמטיקה.
<pre>for (int i=0 ; i<students.Length ; i++) { students[i].SetEng(students[i].GetEng()+10); }</pre>	העלאת הציון באנגלית של כל התלמידים ב 10 נקודות.
<pre>for (int i=0 ; i<students.Length ; i++) { if (students[i].computeAverage(>80) students[i].printCertificate(); }</pre>	הדפסת תעודות לכל תלמידי הכיתה שהממוצע שלהם מעל 80.

סיכום שלבי הצהרה ואתחול של אצטרק צ'פ'ס

ניתן להתייחס לשלושה שלבים בהצהרה ואתחול של מערך ביחס לניהול הזכרון בעת ביצוע תכנית. שלושת השלבים ומשמעותם מתוארים ומודגמים בטבלה הבאה בעבור מערך של תלמידים:

קוד לדוגמה	השלב ומשמעותו
<pre>Student[] students;</pre> <p>הסבר: students הוא מערך שהתאים שלו הם מטיפוס Students.</p>	שלב 1 - הצהרה על מערך מטיפוס של המשתמש.
<pre>students = new Student[33];</pre> <p>הסבר: במערך students נבנים 33 תאים (עם מציינים מ-0 עד 32).</p>	שלב 2 - בניית תאי המערך.

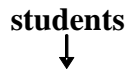


<pre>for (int i=0 ; i<students.Length ; i++) { students[i] = new Student(); } </pre> <p>הסבר : נוצרים 33 עצמים מטיפוס Student וכל אחד מהם מושם בהתאמה בתא מערך לפי המצויין.</p>	<p>שלב 3 - בניית העצמים מהטיפוס והשמתם בתאי המערך. בניית כל עצם תעשה על ידי הפעלת פעולה בונה המוגדרת בטיפוס המתאים. לא כל העצמים צריכים להבנות על ידי אותה פעולה בונה.</p>
<pre>for (int i=0 ; i<students.Length ; i++) { Console.WriteLine("Enter student name:"); name = Console.ReadLine(); Console.WriteLine("Enter student ID:"); id = Console.ReadLine(); students[i] = new Student(name, id); } </pre> <p>הסבר : יצירת עצמים תוך שימוש בפעולה בונה המקבלת כפרמטרים את שם התלמיד ואת תעודת הזהות שלו.</p>	



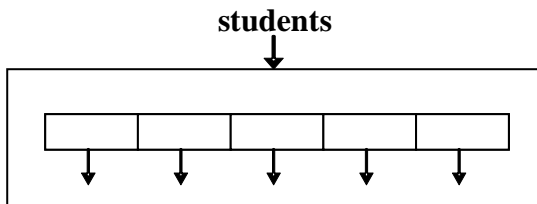
הדגמת תמונת הזיכרון עבור השלבים השונים:

שלב 1 - Student[] students;



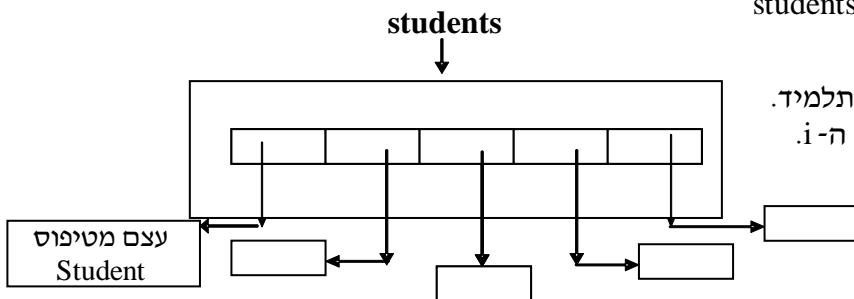
מוקצה תא אחד שיכיל בהמשך הפנייה אל המקום בזיכרון בו יוקצו תאי המערך. לפני הקצאת תאי המערך יש בו הפנייה ריקה – הקבוע null.

שלב 2 - students = new Student[33];



מוקצים תאי הזיכרון של המערך בהתאם לגודל הנדרש. ההפניה הראשית אליהם הושמה במשתנה המערך (במקום ה- null משלב 1). כל תא מיועד להיות עצם, אך העצם עדיין לא נבנה. לכן בכל תא יש הפנייה ריקה – הקבוע null.

שלב 3 - students[i] = new Student();



עבור כל תא במערך נבנה עצם מטיפוס תלמיד. ההפניה אל העצם מושמת בתא המערך ה-i.



דוגמה
פתורה

דוגמה כתורה 1 - מערך שירים

נשתמש בטיפוס Song ונגדיר תכנית ראשית SongProgram המשתמשת במערך של שירים.

בתכנית הראשית מתבצע האלגוריתם הבא:

- (1) הגדר מערך של 10 שירים.
- (2) עבור כל שיר, קלוט את התכונות שלו, בנה עצם מטיפוס שיר והשם אותו במערך השירים.
- (3) חשב והדפס את הממוצע של אורך שיר מבין השירים במערך, בשניות.

כתרון

```
using System;
class Ch7Solve1SongProgram1
{
    // הפעולה הראשית
    public static void Main(string[] args)
    {
        Song[] songs = new Song[10];
        string n;
        string p;
        int len;

        // קליטת נתוני כל שיר בניית עצם מהטיפוס Song והשמתו במערך השירים
        for ( int i = 0 ; i < songs.Length ; i++ )
        {
            Console.WriteLine("Enter song name:");
            n = Console.ReadLine();
            Console.WriteLine("Enter song performer:");
            p = Console.ReadLine();
            Console.WriteLine("Enter song length:");
            len = int.Parse(Console.ReadLine());
            songs[i] = new Song(n, p, len);
        }

        // חישוב ממוצע אורכי השירים והדפסתו
        double sum = 0;
        for ( int i = 0 ; i < songs.Length ; i++ )
        {
            sum = sum + songs[i].GetLength();
        }
        Console.WriteLine("The average song length is: " + sum/songs.Length);
    }
}
```



דוגמה
פתורה

דוגמה כתורה 2 - מערך שירים

הרחב את התכנית הראשית כך שתכלול גם פעולות. הפעולות שהיא תכלול הן בהתאם לממשק הבא:

כותרת הפעולה	תיעוד הפעולה
double AverageSongLength(Song[] sn)	ממוצע-אורכי-שירים(מערך_שירים) פעולה המקבלת מערך שירים, ומחזירה את ממוצע אורכי השירים בשניות.
string TheLongestSong(Song[] sn)	השיר-הארוך-ביותר(מערך_שירים) פעולה המקבלת מערך שירים, ומחזירה את המצוין של השיר הארוך ביותר.
int NumberOfShortSongs(Song[] sn)	מספר-השירים-הקצרים(מערך_שירים) פעולה המקבלת מערך שירים, ומחזירה את מספר השירים שקטגוריית אורך השיר שלהם היא "short".

בתכנית הראשית מתבצע האלגוריתם הבא:

- 1) הגדר מערך של 10 שירים.
- 2) עבור כל שיר, קלוט את התכונות שלו, בנה עצם מטיפוס שיר והשם אותו במערך השירים, מערך_שירים.
- 3) חשב והדפס את הממוצע של אורכי השירים מבין השירים בערך על ידי הפעולה ממוצע-אורכי-שירים(מערך_שירים).
- 4) חשב והדפס את שם השיר הארוך מבין השירים במערך על ידי הפעולה השיר-הארוך-ביותר(מערך_שירים).
- 5) חשב והדפס את מספר השירים הקצרים מבין השירים במערך על ידי הפעולה מספר-השירים-הקצרים(מערך_שירים).

כתרו!

using System;

class SongProgram2

```
{
    /* טענת כניסה : הפעולה מקבלת מערך של שירים */
    /* טענת יציאה : הפעולה מחזירה את ממוצע אורכי השירים במערך בשניות */
    static double AverageSongLength(Song[] sn)
    {
        double sum = 0;
        for ( int i = 0 ; i < sn.Length ; i++ )
        {
            sum = sum + sn[i].GetLength();
        }
        return sum/sn.Length;
    }
}
```

```

/* sn טענת כניסה : הפעולה מקבלת מערך של שירים */
/* טענת יציאה : הפעולה מחזירה את המציין של השיר הארוך ביותר במערך */
/* הנחה : אורכי השירים שונים */
static int TheLongestSong(Song[] sn)
{
    int n = 0;
    for ( int i = 1 ; i < sn.Length ; i++ )
    {
        if (sn[i].GetLength() > sn[n].GetLength())
            n = i;
    }
    return n;
}

.....

/* sn טענת כניסה : הפעולה מקבלת מערך של שירים */
/* טענת יציאה : הפעולה מחזירה את מספר השירים הקצרים במערך */
static int NumberOfShortSongs(Song[] sn)
{
    int count = 0;
    for ( int i = 0 ; i < sn.Length ; i++ )
    {
        if (sn[i].Category().Equals("short"))
            count++;
    }
    return count;
}

.....

/* הפעולה הראשית */
public static void Main(string[] args)
{
    Song[] songs = new Song[5];
    string n;
    string p;
    int len;
    for ( int i = 0 ; i < songs.Length ; i++ )
    {
        Console.WriteLine("Enter song name:");
        n = Console.ReadLine();
        Console.WriteLine("Enter song performer:");
        p = Console.ReadLine();
        Console.WriteLine("Enter song length:");
        len = int.Parse(Console.ReadLine());
        songs[i] = new Song(n, p, len);
    }
    Console.WriteLine("The average song length is: " + AverageSongLength(songs));
    Console.WriteLine("The longest song is: "+song[TheLongestSong(songs)].GetName());
    Console.WriteLine("The number of short songs is:"+NumberOfShortSongs(songs));
}
}

```




דוגמה
פתורה

דוגמה כתורה 3 - מערך שירים

הרחב את התכנית הראשית כך שאתחול המערך על ידי השירים יהיה אף הוא בפעולה.

כותרת הפעולה	תיעוד הפעולה
<code>static Song[] InitialSongArray()</code>	אתחול-מערך-שירים() פעולה הקולטת את מספר השירים הנדרש, ובונה בהתאם לכך את השירים ומשימה אותם למערך.

האלגוריתם בתכנית הראשית לא משתנה.

using System;

class SongProgram3

{

/ הפעולה קולטת מספר שירים נדרש, מאתחלת ומחזירה מערך של שירים בגודל הנדרש */*

static Song[] InitialSongArray()

{

Console.WriteLine("Enter number of songs:");

Song[] sn = **new** Song[Console.ReadLine()];

for (**int** i = 0 ; i < sn.Length ; i++)

{

sn[i] = **new** Song();

}

return sn;

}

// יתר הפעולות כפי שהוגדרו בדוגמה 2

// הפעולה הראשית

public static void Main(string[] args)

{

Song[] songs = InitialSongArray();

Console.WriteLine("The average song length is: " +

AverageSongLength(songs));

Console.WriteLine("The longest song is: " + TheLongestSong(songs));

Console.WriteLine("The number of short songs is: "

+NumberOfShortSongs(songs));

}

}

הפעולה קולטת תוך כדי משפט ההצהרה את גודל המערך ובונה את תאי המערך בהתאם. לאחר מכן יש בנייה של העצמים (שירים) והשמתם לתוך תאי המערך על ידי שימוש בפעולה הבונה `Song()` הקולטת את פרטי השיר. המערך מוצהר בפעולת האתחול והיא מחזירה אותו. בהתאם לכך משפט הזימון לפעולת האתחול בפעולה הראשית. בפעולה הראשית נמצאת רק ההצהרה על המשתנה `songs` שיהיה מטיפוס מערך של שירים. כל תהליך היצירה מתבצע בפעולת האתחול והמערך המוחזר מושם בתוך המשתנה `songs`.



הסבר
הפתרון

תרגילים

★ ★ ★ תרגיל 1: הגדרת קטעי תכנית - שימוש בטיפוס תלמיד

לפניך רשימה של משימות קצרות. כתוב לכל משימה שורות קוד לביצועה. המשימות מתייחסות אל הטיפוס Student.

- א. הגדר מערך של 30 תלמידים.
- ב. כתוב לולאה לאתחול מערך התלמידים הקולטת שם ו ת.ז. של כל תלמיד.
- ג. כתוב לולאה הקולטת את ציוני כל תלמידי הכיתה בלשון, ומעדכנת את הציון שלהם בהתאם. הנח כי סדר הקליטה זהה לסדר התלמידים במערך.
- ד. כתוב פעולה המקבלת את מערך התלמידים, ומדפיסה את שמות התלמידים שממוצע הציונים שלהם הוא מ- 85 ומעלה.
- ה. כתוב פעולה המקבלת את מערך התלמידים, ת.ז. של תלמיד, ואחוז כמספר שלם. הפעולה תעלה את כל הציונים של התלמיד באחוז המתקבל כפרמטר.
- ו. כתוב פעולה המקבלת את מערך התלמידים ומחרוזת ובה שם מקצוע, ומחזירה את מספר התלמידים אשר נכשלו במקצוע.
- ז. כתוב פעולה המדפיסה לכל מקצוע את מספר התלמידים שנכשלו בו.

תרגילים: כתח מחלקה



★ ★ ★ ★ ★ תרגיל 2: הטיפוס רובוט

פתח מחלקה ראשית חדשה לניהול משחק הרובוטים שתואר בתרגיל 18 בפרק 6. המשחק יתקיים בין מספר רובוטים שייקלט מן המשתמש. לצורך שמירת הרובוטים יוגדר מערך. הרובוטים יפעלו כל אחד בתורו באופן מעגלי, תחילה הרובוט הראשון במערך, אחריו השני, וכך עד לאחרון, ..., וחוזר חלילה. כללי התנועה של הרובוטים וכללי ההכרזה על הניצחון הם כפי שהוגדרו בתרגיל הקודם.

★ ★ ★ ★ ★ תרגיל 3: פרויקט מרוץ מכוניות

פתח פרויקט המדמה מרוץ מכונית. במרוץ משתתפות n מכוניות. לכל מכונית יש את המהירות המקסימלית אליה היא יכולה להגיע. במהלך המרוץ המכונית מאטה או מאיצה לחלופין. עם פתיחת המרוץ כל המכוניות מזנקות בו זמנית. מדמה המרוץ שלנו יגריל עבור כל מכונית בכל שלב מספר שלם בתחום של 1-4. קוד 1 – המכונית מגבירה מהירותה ב- 10 קמ"ש (בתנאי שלא עברה את המהירות המקסימלית שלה), קוד 2 – המכונית מאטה את מהירותה ב- 10 קמ"ש (בתנאי שלא ירדה ממהירות 0 קמ"ש), קוד 3 – המכונית מאטה את מהירותה ב- 20 קמ"ש (בתנאי שלא ירדה ממהירות 0 קמ"ש), קוד 4 – ארעה תקלה או התנגשות – המכונית פורשת מן המרוץ. המכונית שחוצה ראשונה את קו הסיום היא המנצחת במרוץ. הפרויקט יכול שתי מחלקות:

- (1) המחלקה Car המייצגת את הטיפוס מכונית מרוץ,
 (2) מחלקה ראשית CarRace אשר בה יוגדר מערך המכוניות ובה תתקיים הסימולציה של המרוץ כפי שתוארה לעיל.
- הסימולציה של ניהול הזמן והמרחק תתבצע באופן הבא. כל "שלב" במרוץ הוא מעבר על כל אחת מן המכוניות – והוא יילקח בחשבון כזמן של 2 דקות. בכל שלב יוגרל עבור המכונית קוד ההתקדמות שלה לקטע הבא, בהתאם למהירותה ולמשך הזמן של השלב, יחושב המרחק אותו עברה בקטע ויעודכן. אורך המסלול הוא 10 ק"מ.
- פתח את הפרויקט על פי השלבים הבאים:
- תאר את הטיפוס Car על ידי דיאגרמה לתיאור טיפוס. שים לב במיוחד לבחירת התכונות המאפיינות מכונית ולפעולות שיש להגדיר בעבור מכונית.
 - כתוב פעולות בונות מתאימות.
 - כתוב פעולות קובעות ומאחזרות לכל תכונה.
 - פתח את המחלקה Car בשפת התכנות.
 - פתח את המחלקה CarRace שהיא התכנית הראשית. בתכנית הראשית: יוגדר מערך המכוניות, יתנהל המרוץ על פי כללי הסימולציה שתוארו לעיל, ויוכרו המנצח במרוץ. הערה: נסה להגדיר כמה שיותר פעולות משמעותיות במחלקה הראשית כך שהפעולה הראשית תתאר אלגוריתם קצר ובהיר.

דג'לים לסיכום הסדרק

1. מהו מערך של עצמים?
 מערך בו כל אחד מן האיברים הוא עצם. העצמים יכולים להיות מטיפוס בנוי כמו string, או מטיפוס שהשתמש מגדיר כמו Student, Car, Robot.
2. השימוש במערך של עצמים?
 מערך של עצמים משמש לעיבוד מידע בדיוק כמו מערך מכל טיפוס. כל תא הוא משתנה לכל דבר אלא שהערך שלו הוא עצם, ויש לפעול איתו בהתאם לפעולות המוגדרות בטיפוס המתאים.
3. כיצד נגדיר מערך של עצמים?
 הגדרת מערך של עצמים נעשית בשלושה שלבים:
 שלב 1 – הגדרת המערך. למשל:
 Car[] cars;
 שלב 2 – יצירת תאי המערך. למשל:
 cars = new Car[n];
 שלב 3 – יצירת עצמים והשמתם בתאי המערך המתאימים. למשל:
 cars[i] = new Car();