

מערכים

מהו מערך (array) ומדוע יש בו צורך?

נניח שרצינו לכתוב תוכנית שצריכה לקלוט את מרחקי הנסיעה של כל תלמיד מבית הספר, לחשב את הממוצע של המרחקים ולאחר מכן להדפיס את המרחקים שמעל הממוצע ולפי זה לקבוע למי מותר לאחר (על כל קילומטר מעל הממוצע, איחור של דקה).

איך נעשה זאת על פי הידע הנוכחי שלנו?

אפשר לקלוט את המרחקים ותוך כדי הקליטה לחשב את הממוצע, אולם אחרי הקליטה והחישוב, כיצד 'נזכור' מה היה כל מרחק? היכן נשמור את הנתונים?

אם היו רק 3 תלמידים, כיצד היינו עושים זאת? מה אם יש 300 או 3000 תלמידים?

מערך זהו רצף של תאים בזיכרון שכולם מאותו סוג. זה אומר שיהיה לו סוג ויהיה לו שם.

המערך שלנו יהיה בסך הכל מורכב משני חלקים: שם המערך והאינדקס של האיבר במערך. כך נוכל תמיד להתייחס לשם המערך ולאנדקס הרצוי. כל איבר במערך יהיה מאותו סוג.

צורת ההגדרה (נגדיר ב-2 שורות)

```
שם-המערך [ ] סוג-המערך
[גודל] סוג =new שם-המערך
```

למשל:

```
int [ ] array1;
array1= new int[10]
```

כאן הגדרנו מערך בשם: array1 מסוג שלם (יכול להכיל רק שלמים) והוא בגודל של 10 איברים. ההתייחסות לאיבר בודד במערך נעשית בעזרת שמו והאינדקס של האיבר שאליו רוצים להתייחס, למשל:

```
array1[3]= 5;
array1[0]=18;
```

האינדקס של המערך מתחיל באפס ומסתיים באחד פחות הגודל שלו, ולכן בהגדרה הנ"ל נוכל להתייחס לאיברים: array1[0], array1[1], וכך הלאה עד לאחרון שהוא: array1[9].

ניקח דוגמה נוספת: הגדר מערך של 50 מחרוזות בשם: strings

```
string [ ] strings;
strings = new string[50];
strings[0] = "initial value";
```

הסיבה שדרושות 2 שורות להגדיר מערך ב- C# זה כיוון שכל הגדרה בשפה היא למעשה עצם, והשורה הראשונה רק מציינת שם של מיקום בזיכרון, השורה השנייה מקצה מקום בזיכרון (על ידי הפקודה new). מערכים נחשבים חלק מהזיכרון הדינאמי של התוכנית, כלומר זיכרון שמוקצה בזמן הריצה על ידי פקודת ה-new. משתנים שאינם מערכים לא נזקקים למילה new כדי ליצור אותם בזיכרון (זיכרון סטאטי).

ניתן לקצר את ההגדרה של מערך לשורה אחת כדלקמן:

```
int [ ] array1 = new int[10];
```

דוגמה לתוכנית שעושה את מה שמתואר למעלה עם 10 מספרים (לא 10,000), כלומר קלטה 10 מספרים ואיחסנה אותם במערך. לאחר מכן ביקשה מהמשתמש מספר סידורי של אחד מהמספרים שהכניס קודם, והתוכנית הדפיסה את ערכו.

```
int which;
int [ ] ar1 = new int[10];
Console.WriteLine("Please enter 10 integers of any size:");
for (int i=0; i<10; ++i)
{
    ar1[i]= int.Parse(Console.ReadLine());
    Console.WriteLine("Next:");
}
Console.WriteLine("Please enter which number you want back (its sequence)");
which= int.Parse(Console.ReadLine());
Console.WriteLine("Your {0} number was {1}", which, ar1[which-1]);
```

שם המערך הוא: ar1 והאינדקס איתו משתמשים בכל פעם בלולאה הוא: i - כך כל פעם שהמשתמש מקיש מספר, הוא נקלט לאיבר שונה במערך (ar1[0], ar1[1],.....ar1[9]) ולכן המספר שהוקש ראשון יהיה באיבר אפס, המספר שהוקש רביעי יהיה באיבר שלישי. כיוון שמערך מתחיל ב-0, כדי להחזיר את המספר הרצוי, אנו מורידים מהאינדקס אחד בהדפסה החוזרת עבור המשתמש (which-1). שימו לב שהמשתנה which כאן משמש עבור מספרו הסידורי של הערך שבו המשתמש מעוניין.