

שולה מוקשים - דף הנחיה 1

חישבו על מהלך המשחק. נסו לחשוב "מלמעלה למטה" - top-down, אל תצללו לפרטים מוקדם מידי. ככל שתחשבו מופשט יותר, גבוה יותר, כך יהיה לכם קל יותר לעמוד במשימה.

בדף הנחיה זה תימצאו שני דברים:

1. כיצד נראית התכנית הראשית שאני כתבתי. כל היתר (כמעט) נמצא באובייקט `MinesGame`. שימו לב, שהפעולה הראשית מספקת לכם אינפומציה (חלקית) מה קיים באובייקט `MinesGame`.
2. לנוחיותכם, סיפקתי לכם פעולה סטטית בשם `GetLocation`, המאפשרת למשתמש "לנוע" עם החיצים על גבי הלוח, ובהקשת `Enter` הפעולה תחזיר מערך באורך 2, עם קואורדינטות הלחיצה. זו הפעולה היחידה פרט ל-`Main` הנמצאת המחלקה הראשית...

הפעולה הראשית

```
static void Main(string[] args)
{
    int rows = 10, cols = 10, m = 10;
    MinesGame mines = new MinesGame(rows, cols, m);    // Create a new board
    mines.PrintBoard();

    int[] loc;
    bool gameOver = false, lost = false;

    while (!gameOver)
    {
        loc = GetLocation(2, 2, rows, cols);    // gets the user click position

        lost = mines.Click(loc[0], loc[1]);
        gameOver = mines.PrintBoard();

        if (lost)
        {
            Console.SetCursorPosition(4, 15);
            Console.WriteLine("You lost....");
            gameOver = true;
        }
        else if (gameOver)
        {
            Console.SetCursorPosition(4, 15);
            Console.WriteLine("\n\n Hurey...");
        }
    }
}
```

הפעולה GetLocation

טענת כניסה: הפעולה מקבלת את קואורדינטות ההתחלה של הלוח על גבי מסך המחשב (החלון השחור) וכן את גודל הלוח במשבצות: מספר שורות ומספר עמודות.

טענת יציאה: הפעולה מאפשרת למשתמש לנוע עם החיצים במרחבי הלוח, עד הקשת Enter, הפעולה מחזירה מערך של שלמים באורך 2, במקום הראשון מספר השורה, ובשני מספר העמודה.

```
static int[] GetLocation(int x0, int y0, int rows, int cols)
{
    int x = x0, y = y0;
    int xmax = x0 + rows - 1;
    int ymax = y0 + cols - 1;

    Console.SetCursorPosition(y, x);
    Console.CursorSize = 75;
    bool enter = false;

    while (!enter)
    {
        ConsoleKeyInfo key = Console.ReadKey(true);
        switch (key.Key)
        {
            case ConsoleKey.UpArrow:
                x = Math.Max(x0, x - 1);
                break;
            case ConsoleKey.DownArrow:
                x = Math.Min(xmax, x + 1);
                break;
            case ConsoleKey.LeftArrow:
                y = Math.Max(y0, y - 1);
                break;
            case ConsoleKey.RightArrow:
                y = Math.Min(ymax, y + 1);
                break;
            case ConsoleKey.Enter:
                enter = true;
                break;
            default:
                break;
        }
        Console.SetCursorPosition(y, x);
    }

    int[] pos = { x - x0, y - y0 };
    return pos;
}
```

למי שיבחר לשדרג את המשחק עם מקשים נוספים, זו הפעולה אותה תצטרכו לשנות.
גם את הלוגיקה וכנראה גם את המערך המוחזר.

בהצלחה!