

קורס מורים מובילים תש"ע, פיתוח הפרויקט: ג'בארה אחמד, לודמר רחל, צרניחוב רחלי

פרויקטים שהוכנו על-ידי משתתפי קורס מורים מובילים תש"ע

ניתן להשתמש בחומרים לצורך הוראה בלבד.

לא ניתן לפרסם את החומרים או לעשות בהם כל שימוש מסחרי

ללא קבלת אישור מראש מצוות הפיתוח

תיק פרויקט משחק הזיכרון

ג'בארה אחמד

לודמר רחל

צרניחוב רחלי

סביבות הפיתוח:

JAVA / C#

מטרת הפרויקט:

זהו פרויקט המסכם את הנלמד ביסודות 1+2 על פי הגישה של עצמים תחילה. בנוסף יש כאן שימוש במערך דו-ממדי – תוספת לחומר הנלמד ביסודות.

הפרויקט מכיל 5 מחלקות.

1. Program - מחלקה ראשית
2. Game - המחלקה משחק.
3. Cell - המחלקה תא.
4. Board - המחלקה לוח המשחק.
5. Utility -מחלקת עזר. (ללא תכונות)

תיאור המשחק:

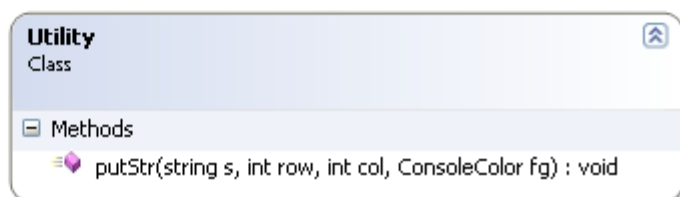
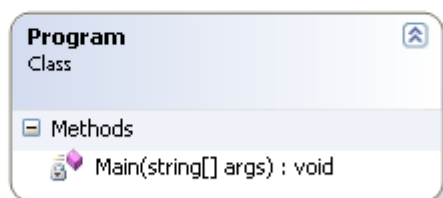
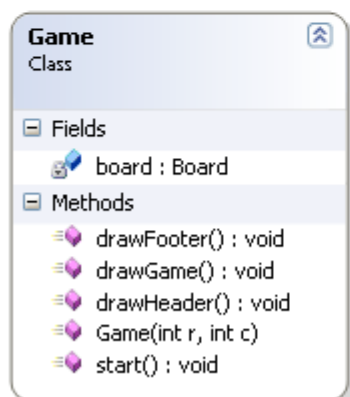
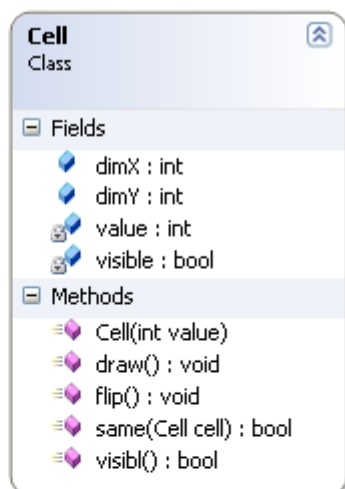
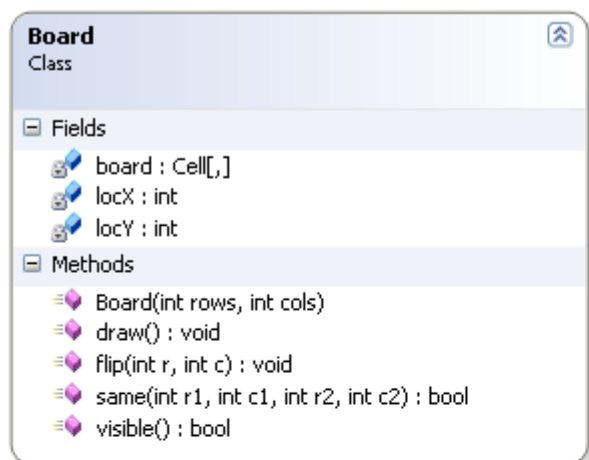
בתחילת המשחק נפתח חלון עם לוח בגודל 4*4 של איקסים.

```
XXXXX  
XXXXX  
XXXXX  
XXXXX
```

על המשתמש לבחור מספר שורה ומספר עמודה ועל פי הבחירה הוא רואה את הספרה הנמצאת במקום שנבחר. עליו לבחור מיקום נוסף ואז המשתמש רואה שתי ספרות. אם הן זהות התאים נשארים גלויים לעין. אם הספרות אינן זהות המיקומים שנבחנו מסומנים ב-X.

המשחק ממשיך עד שהמשתמש מצליח לנחש את המיקום של כל הזוגות. הערה: בפתרון בגירסת java במקום ספרות המשתמש רואה תמונה.

תרשים מחלקות:



C# פתרון

```
using System;
using System.Collections.Generic;
using System.Text;
namespace MemoryGameProject
{
    class Program
    {
        static void Main(string[] args)
        {
            Game g = new Game(10,10);
            g.start();
        }
    }
}

using System;
using System.Collections.Generic;
using System.Text;
using System.Threading;
namespace MemoryGameProject
{
    public class Game
    {
        private Board board;
        public Game(int r, int c)
        {
            board = new Board(r, c);
        }
        public void drawHeader()
        {
            String s = "Memory Game";
            Console.SetCursorPosition(Console.LargestWindowWidth/2 -
s.Length,0);
            Console.Write(s);
            for (int k = 0; k < Console.LargestWindowWidth; k++)
                MemoryGameProject.Utility.putStr("■", 3, k,
ConsoleColor.White);
        }
        public void drawFooter()
        {
            for (int k = 0; k < Console.LargestWindowWidth; k++)
                MemoryGameProject.Utility.putStr("■",
Console.LargestWindowHeight - 5, k, ConsoleColor.White);
        }
        public void drawGame()
        {
            board.draw();
            drawHeader();
            drawFooter();
        }
        public void start()
        {
            int r1, c1, r2=0, c2=0;
            bool firstTime = false;

```

קורס מורים מובילים תש"ע, פיתוח הפרויקט: ג'בארה אחמד, לודמר רחל, צרניחוב רחלי

```
do{
    drawGame();
    MemoryGameProject.Utility.putStr("enter row:", 1, 0,
ConsoleColor.Yellow);
    r1 = int.Parse(Console.ReadLine());
    MemoryGameProject.Utility.putStr("enter column:", 2, 0,
ConsoleColor.Yellow);
    c1 = int.Parse(Console.ReadLine());
    board.flip(r1 - 1, c1 - 1);
    if (!firstTime) {
        r2 = r1;
        c2 = c1;
    }
    else {
        drawGame();
        Thread.Sleep(1000);
        if (!board.same(r1-1, c1-1, r2-1, c2-1))
        {
            board.flip(r1-1, c1-1);
            board.flip(r2-1, c2-1);
        }
    }
    firstTime = !firstTime;
} while (!board.visible());
}
}

using System;
using System.Collections.Generic;
using System.Text;

namespace MemoryGameProject
{
    public class Cell
    {
        private int value;
        private Boolean visible;
        public static int dimX=4,dimY=2;
        public Cell(int value)
        {
            this.value = value;
        }
        public void draw()
        {
            if (visible)
                Console.WriteLine(value);
            else
                Console.WriteLine("*");
        }
        public void flip()
        {
            visible = !visible;
        }
        public bool same(Cell cell)
        {
            return value == cell.value;
        }
    }
}
```

קורס מורים מובילים תש"ע, פיתוח הפרויקט: ג'בארה אחמד, לודמר רחל, צרניחוב רחל

```
        public bool visible()
        {
            return visible == true;
        }
    }
}

using System;
using System.Collections.Generic;
using System.Text;

namespace MemoryGameProject
{
    public class Board
    {
        private Cell[,] board;
        private int locX=10, locY=5;

        public Board(int rows, int cols)
        {
            int[] counters = new int[rows * cols / 2];
            int k;
            Random r = new Random();
            board = new Cell[rows,cols];
            locX = Console.LargestWindowWidth / 2 - (cols / 2) * 4;
            locY = Console.LargestWindowHeight / 2 - (rows / 2) * 2;
            for (int i = 0; i < board.GetLength(0); i++)
                for (int j = 0; j < board.GetLength(1); j++)
                {
                    while (counters[k = r.Next(rows * cols / 2)] == 2) ;
                    counters[k]++;
                    board[i, j] = new Cell(k);
                }
        }

        public void draw()
        {
            Console.Clear();
            Console.SetBufferSize(Console.LargestWindowWidth,
            Console.LargestWindowHeight);
            Console.SetWindowSize(Console.LargestWindowWidth,
            Console.LargestWindowHeight);
            Console.WindowTop = 0;
            Console.WindowLeft = 0;

            for (int i = 0; i < board.GetLength(0); i++)
            {
                for (int j = 0; j < board.GetLength(1); j++)
                {
                    Console.SetCursorPosition(locX + j*4, locY + i*2);
                    board[i, j].draw();
                }
            }
        }

        public void flip(int r, int c)
        {
            board[r, c].flip();
        }
    }
}
```

קורס מורים מובילים תש"ע, פיתוח הפרויקט: ג'בארה אחמד, לודמר רחל, צרניחוב רחלי

```
public bool same(int r1, int c1, int r2, int c2)
{
    return board[r1, c1].same(board[r2,c2]);
}

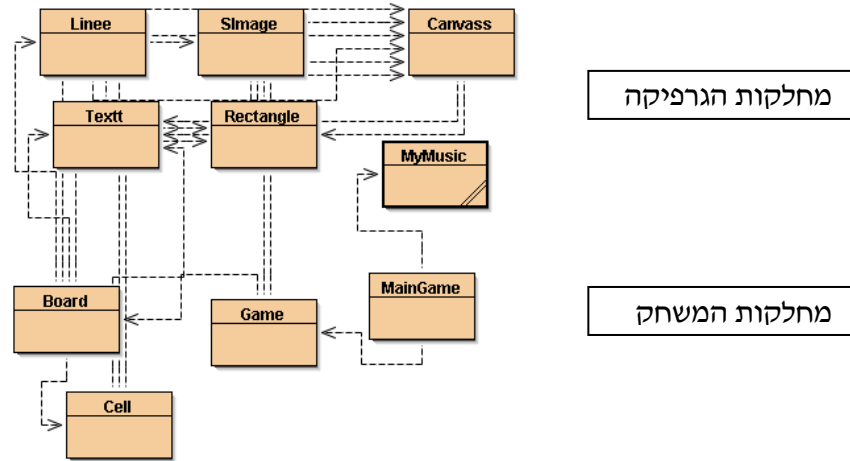
public bool visible()
{
    for (int i = 0; i < board.GetLength(0); i++)
        for (int j = 0; j < board.GetLength(1); j++)
            if (!board[i, j].visibl())
                return false;
    return true;
}
}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace MemoryGameProject
{
    class Utility
    {
        public static void putStr(String s, int row, int col, ConsoleColor
fg)
        {
            ConsoleColor tmp = Console.ForegroundColor;
            Console.SetCursorPosition(col, row);
            Console.ForegroundColor = fg;
            Console.Write(s);
            Console.ForegroundColor = tmp;
        }
    }
}
```

פתרון JAVA

הפתרון בjava מלווה בתצוגה גרפית תוך שימוש בתמונות, עכבר והשמעת מוסיקה ברקע. ולכן במחלקות המשחק יש מספר הבדלים קטנים בין גירסת ה- C# לגירסת java שהן תולדה של שימוש במחלקות הגרפיות.



תודה מיוחדת לשמוליק שוורץ ולתלמידו עמיר בגאון מכפר בתיה על פיתוח המחלקות הגרפיות. המחלקות הגרפיות הן הרחבה של הגרפיקה שפותחה ע"י פרו' מוטי בן ארי, ומאפשרות שימוש בתמונה עם הקלקה בעזרת העכבר, והשמעת קובץ מוסיקה מסוג "au". שמות המחלקות שוננו מהמקור על מנת לא לאפשר התנגשות בין המחלקות בפיתוח החדש לבין המחלקות הגרפיות של מוטי בן ארי (קובץ graphics.jar). שינוי שמות המחלקות מאפשר להשתמש בפיתוח של שמוליק ולהשאיר את הקובץ הגרפי של מוטי "שתול" ב- BlueJ או באקליפס. השינוי בשמות מזערי כמו: במקום Canvas נרשם Canvass.


```
import java.util.*;
public class MainGame
{
static Scanner reader=new Scanner(System.in);

public static void main(String[] args)
{
    MyMusic m=new MyMusic("gaya.au");//קובץ מוסיקה
    m.startPlayFile();//הפעלת המוסיקה
    Game g = new Game();
    g.start();
    m.stopPlayFile();//עצירת קובץ המוסיקה
}
}

import java.util.*;
public class Game
{
    static Scanner reader=new Scanner(System.in);
    private Board board;
    public Game()
    {
        board = new Board(4, 4);
    }
    public void start()
    {
        int r1=1, c1=1, r2=0, c2=0;
        boolean firstTime = false;
        Canvass canvas = Canvass.getCanvas();
        int side=canvas.getWidth()/4;
        Textt t1 = new Textt(200, 580, "לחץ על המשבצת הרצויה", "red");
        String color="red";
        do
        {
            board.draw();
            canvas.wait(50);
            boolean ok=true;
            while( ok)
            {
                if (Canvass.M_Press_Flag == 1)// התבצעה לחיצה
                {
                    Canvass.M_Press_Flag = 0; // מתכוון ללחיצה הבאה בכך
                    שמשנה את הדגל למצב שלא היתה לחיצה
                    c1=Canvass.MX/side+1;
                    r1=Canvass.MY/side+1;
                    if (c1>4 || r1>4)
                    {
                        t1.erase();
                        t1 = new Textt(200, 580, "הינך מחוץ לתחום", "blue");
                    }
                }
                else
            }
        }
    }
}
```

```
        ok=false;
    }
}

board.flip(r1 - 1, c1 - 1);
board.draw();
canvas.wait(50);
if (!firstTime)
{
    r2 = r1;
    c2 = c1;
}
else
{
    /*
    try
    {
        Thread.sleep(2000);
    }
    catch (Exception e)
    {
    }
    */
    if (!board.same(r1-1, c1-1, r2-1, c2-1))
    {
        board.flip(r1-1, c1-1);
        board.flip(r2-1, c2-1);
    }
}
firstTime = !firstTime;
if (color.equals("red"))
    color="black";
else color="red";
// שינוי צבע ההודעה לקראת הלחיצה
t1.erase();
t1 = new Textt(200, 580, "לחץ על המשבצת הרצויה", color);

} while (!board.visible());
t1.erase();
t1 = new Textt(200, 580, "כל הכבוד!, יש מקום לשיפור הזיכרון", color);
}
}
}
```

public class Board

```
{
    private Cell[][] board;
    private int side=560;

    public Board(int rows, int cols)
```

```

{
    int[] counters = new int[rows * cols / 2];
    int k;
    boolean flag=false;
    board = new Cell[rows][cols];

    for (int i = 0; i < board.length; i++)
        for (int j = 0; j < board[i].length; j++)
            {
                flag=false;
                while (!flag)
                {
                    k=(int)(Math.random()*(rows*cols/2));
                    if (counters[k]<2)
                    {
                        flag=true;
                        counters[k]++;
                        board[i][ j] = new Cell(k);
                    }
                }
            }
        }
    //ציור הלוח
    //Canvass.changeSize(side+1,side+1);
    Rectangle c1=new Rectangle(0 ,0, side,side, "yellow");
    // Textt t1 = new Textt(200, 580, "לחץ על המשבצת הרצויה", "red");
    int x=0, y=0;
    int d=side/4;
    for (int i=0; i<=5; i++)
    {
        Linee l=new Linee(0, y, side, y, "black");
        y=y+d;
    }
    x=0;
    for (int j=0; j<=5; j++)
    {
        Linee l=new Linee(x , 0, x, side ,"black");
        x=x+d;
    }
}

public void draw()
{
    int d=this.side/4;
    int x=d/2; int y=d/2;
    // String color="blue";
    for (int i = 0; i < board.length; i++)
    {
        for (int j = 0; j < board[i].length; j++)
        {
            board[i][j].drawCell(x,y,side/4);
        }
    }
}

```

```
        x=x+d;
    }
    x=d/2;
    y=y+d;
}

}

public void flip(int r, int c)
{
    board[r][c].flipCell();
}

public boolean same(int r1, int c1, int r2, int c2)
{
    return board[r1][c1].same(board[r2][c2]);
}

public boolean visible()
{
    for (int i = 0; i < board.length; i++)
        for (int j = 0; j < board[i].length; j++)
            if (!board[i][j].visible())
                return false;
    return true;
}
}

import java.util.*;
public class Cell
{
    static Scanner reader=new Scanner(System.in);
    private int value;
    private boolean visible;
    public Cell(int value)
    {
        this.value = value;
    }
    public void drawCell(int x,int y, int side)
    {
        // אפשר עם מספרים או אותיות
        //String s=""+tav;
        //תמונות במקום
        if (this.visible)
        {
            // כדי למחוק ערך קודם נכתוב את הערך הקודם בצבע הרקע
            // עבור גירסה עם אותיות במקום תמונות
            //Textt t1=new Textt(x,y,"*", "yellow");
            //עבור גירסה עם אותיות במקום תמונות
            //Textt t2=new Textt(x,y,s,"black");
        }
    }
}
```

```
this.paintEmpty(x, y, side);
this.paint(x, y);//x,y is center of picture

}
else
{
    // כדי למחוק ערך קודם נכתוב את הערך הקודם בצבע הרקע
    // Text t1=new Text(x,y,s,"yellow");//עבור גירסה עם אותיות במקום תמונות
    this.paintEmpty(x, y, side);
    Textt t2=new Textt(x,y,"*", "blue");
}
}

public void paintEmpty(int x,int y, int side)
{
    Rectangle c=new Rectangle(x-side/2+5, y-side/2+5, side-5,side-5,"yellow");
}

public void paint(int x,int y)
{
    int picHight=70, picWidth=70; // מימדי התמונה כפי שיוצגו
    SImage pic;
    switch(this.value)
    {
        case 0:pic = new SImage(x-picHight/2,y-picWidth/2,
"blueBall",picHight,picWidth);
        break;
        case 1: pic = new SImage(x-picHight/2,y-picWidth/2,
"flower",picHight,picWidth);
        break;
        case 2: pic = new SImage(x-picHight/2,y-picWidth/2,
"roses",picHight,picWidth);
        break;
        case 3: pic = new SImage(x-picHight/2,y-picWidth/2,
"butterfly",picHight,picWidth);
        break;
        case 4: pic = new SImage(x-picHight/2,y-picWidth/2,
"body",picHight,picWidth);
        break;
        case 5: pic = new SImage(x-picHight/2,y-picWidth/2,
"caricatura",picHight,picWidth);
        break;
        case 6: pic = new SImage(x-picHight/2,y-picWidth/2,
"cadur_poreach",picHight,picWidth);
        break;
        case 7: pic = new SImage(x-picHight/2,y-picWidth/2,
"computer",picHight,picWidth);
        break;
    }
}
```

```
case 8: pic = new SImage(x-picHight/2,y-picWidth/2,  
"smallflower",picHight,picWidth);
```

```
    break;  
  }  
}
```

```
public void flipCell()  
{  
    visible = !visible;  
}
```

```
public boolean same(Cell cell)  
{  
    return value == cell.value;  
}
```

```
public boolean visibl()  
{  
    return visible == true;  
}  
}
```