

תרגיל מס' 2 – אותלו לקונסול

מטרות

- הטמעה של עבודה עם מחלקות ותכנות מונחה עצמים
- עבודה עם מערכים/מחלקות אוספים
- שימוש במחלקה string
- שימוש ב-Dll (אסמבלי) חיצוני

ידע נדרש

- היכרות עם כלי הפיתוח Microsoft Visual Studio .NET
- הכרות עם תחביר בסיסי של שפת C#
- עבודה עם מחלקות (Access modifiers, Constructors, Properties, Indexers)
- עבודה עם מערכים / אוספים / מבני נתונים
- שימוש במחלקה string
- שימוש ב-Dll (אסמבלי) חיצוני

הכינותי מראש

- Microsoft Visual Studio מותקן על המחשב.
- קובץ Ex02.ConsoleUtils.dll שנמצא כחלק מחבילת ההורדה מהאתר.

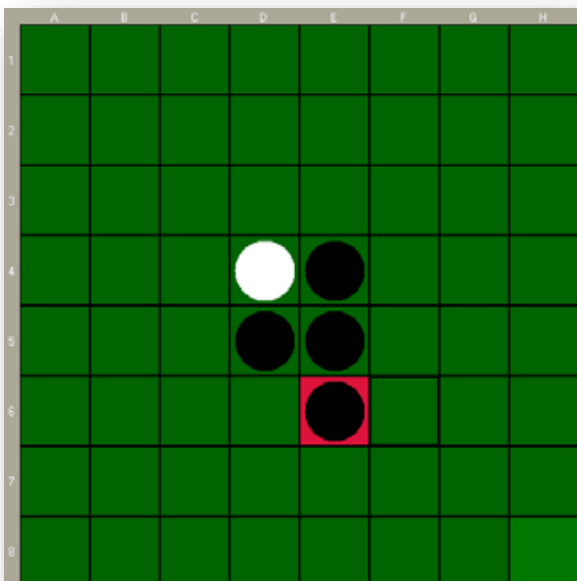
התרגיל

יש לממש את משחק "אותלו" כאשר ה-console הוא הממשק למשתמש.

המשחק הוא משחק לוח ידוע שהומצא בשנת 1880..

במקור, לוח המשחק הוא מטריצה בגודל 8 על 8 (אולם יש לאפשר גם לקבוע אותה לגודל אחר!) שיכולה להכיל מטבעות צבועים שחור לבן (צבע שונה בכל צד של המטבע). לכל שחקן יש את הצבע המייצג שלו (כמו בדמקה). כל משתמש מוסיף בתורו מטבע ללוח באופן כזה שהיא תחסום רצף של מטבעות של השחקן השני שחסום מהצד השני במטבע נוספת שלו, ואז המטבעות החסומות מתהפכות לצבע שלו. המנצח הוא זה שיש לו יותר מטבעות בצבע שלו על הלוח.

התרגיל ימומש בעזרת קונסול ולא ע"י ממשק משתמש גראפי.



	A	B	C	D	E	F	G	H
1								
2								
3								
4				O	X			
5				X	X			
6					X			
7								
8								

מהלך המשחק

1. המשתמש מתבקש להזין את שמו (הוא השחקן הראשון).
2. המשתמש מתבקש להחליט האם המשחק הוא של שני שחקנים או נגד המחשב. במידה והוא בוחר באפשרות של שני שחקנים, הוא מתבקש להזין את שמו של השחקן השני.
3. המשתמש מתבקש לקבוע את גודל המטריצה (6x6 או 8x8).
4. בתחילת סיבוב המטריצה מכילה 4 מטבעות, שניים בכל צבע (X מייצג שחור, O מייצג לבן):

	A	B	C	D	E	F	G	H
1								
2								
3								
4					O	X		
5					X	O		
6								
7								
8								

5. כל שחקן בתורו יתבקש לבחור את המשבצת שבה הוא רוצה לשים את המטבע שלו.
 6. אם המשבצת אינה חוקית (כלומר אינה מייצרת חסימה של רצף של מטבעות של השחקן השני), תינתן הודעה מתאימה וכך הלאה עד אשר הוא יבחר משבצת חוקית.
 7. אחרי שהשחקן בוחר משבצת, ננקה את את המסך, נצייר את הלוח כפי שהוא כרגע אחרי שהפכנו את המטבעות של היריב, והתור יעבור ליריב.
- מצב הלוח אחרי שהשחקן הראשון בחר את המשבצת E3:

	A	B	C	D	E	F	G	H
1								
2								
3					O			
4					O	O		
5					X	O		
6								
7								
8								

8. אם אין לשחקן מסויים משבצות פנויות וחוקיות, תצא הודעה מתאימה, והתור יעבור לשחקן היריב.
9. אם אין לשני השחקנים משבצות פנויות וחוקיות, המשחק נגמר ונקבע ניצחון לשחקן עם מספר המטבעות הגדול ביותר בלוח (תצא הודעה מתאימה הכוללת את מצב הנקודות בין השחקנים. כל שחקן מקבל נקודות לפי מספר המטבעות שלו על הלוח).
10. אחרי סיום משחק, המשתמש יישאל אם ברצונו להתחיל סיבוב נוסף. אם כן חוזרים לסעיף 4. אם לא, תצא הודעה והתוכנית תסתיים.

הוראות כלליות

- לצורך המהלכים של המחשב יש להגריל משבצת מבין המשבצות שמהוות מהלך חוקי. לצורך כך ניתן להשתמש במתודה Next של המחלקה Random. (כלומר אין צורך לממש אינטליגנציה מלאכותית כלשהיא עבור המהלכים של המחשב)
- בכל שלב ניתן לפרוש מהמשחק ע"י בחירת "Q" במקום משבצת (פרישה כללית מהתוכנה).
- יש לבדוק תקינות קלט בכל בקשת קלט מהמשתמש ולהציג הודעה מתאימה במקרה של קלט לא תקין (יש להבדיל בין קלט לא תקין תחבירית לבין קלט לא תקין עניינית).
- לפני כל הדפסה של מצב הלוח יש לבצע ניקוי מסך. לשם כך, יש להיעזר בקובץ `Ex02.ConsoleUtils.dll` שמכיל את מחלקת השירות `Ex02.ConsoleUtils.Screen` שמכילה את המתודה הסטטית `Clear()`.
 הדרך לפניה ל-dll שלי מתוך הקוד שלכם היא פשוטה:
 ללחוץ כפתור ימני של עכבר בחלון ה-Solution Explorer, מעל ה-References של הפרויקט. לבחור באופציה Add Reference, ואז לבחור ב-dll באמצעות האפשרות Browse. כעת, ניתן לגשת ל-namespace שמוגדר בתוך ה-dll ולהפעיל מתודות מתוך ה-dll באופן זהה להפעלת מתודות ספריה רגילות.
 חשוב: אין לצרף את הקובץ הזה בהגשה (אחרת GMAIL ידחה את העבודה שלכם). (לבדוק יש את הקובץ אצלו)
- למחלקה שבה יש את המתודה Main (ורק אותה) קוראים Program. במתודה Main אין לכתוב יותר מ-4 שורות קוד (ועדיף כמה שפחות).
- שם הפרוייקט יהיה `Ex02_Othelo`
- ארכיטקטורה והנדסת תוכנה:
 - יש להשתמש בתיכנות מונחה עצמים! בחירה נכונה של מחלקות וחלוקה נכונה למתודות מהווים נקודות חשובות בתרגיל זה. אין להשתמש בהורשה ו/או פולימורפיזם.
 - יש להפגין שימוש נכון ביכולות השפה ודוט נט.
 - יש לבצע הפרדה מתאימה בין המחלקות שמנהלות את הלוגיקה והנתונים של המשחק לבין המחלקות שמציגות את מצב המשחק ומייצרות אינטראקציה עם המשתמש. במילים אחרות: הפרדה בין המימוש של ממשק המשתמש (UI) למימוש של הלוגיקה של המערכת. קחו בחשבון שחלק מהמימושים אמורים לשרת אתכם גם בשלב שבו נרצה לפתח את המשחק לסביבת "חלונות" והשאיר את השאר כמה שיותר חלקים ללא שינוי!
- מימוש אינטליגנציה מלאכותית (AI) כלשהיא למהלכים של המחשב יזכה בבנוס של עד 6 נקודות! לצורך כך אתם מוזמנים להעזר בשלל מאמרים שמתארים שיטות למימוש AI למשחק זה, שניתן למצוא בקלות באמצעות [חיפוש קל בגוגל](#).
- ניתן להעזר בקבוצת הקורס בפייסבוק כדי לשאול שאלות בנוגע לתרגיל.
- יש לעמוד בתקנים לכתיבת קוד כפי שמפורט במסמך הרלוונטי שניתן למצוא באתר הקורס. שימו לב בעיקר לתקנים לגבי שמות של שדות מחלקה ופרמטרים לפונקציה. נקודות יירדו למי שלא יעמוד בתקנים אלו.
- יש לעמוד בהוראות ההגשה כפי שמפורט במסמך הרלוונטי שניתן למצוא באתר הקורס. נקודות יירדו למי שלא יפעל באופן מדויק לפי הוראות אלה.
- נא להמנע מהעתקות (הן מתגלות מאוד בקלות)