**The Maze**

פרוייקט 5 יח"ל

בתכנון ותכנות מערכות – הגנת סייבר

שם התלמיד: מתן

שם משפחה: עודד

כיתה: י"ב 3

ביה"ס אוסטרובסקי רעננה

מורה מלווה אתי בררו

קיץ 2018

תוכן עניינים

[מבוא 3](#_Toc509358763)

[אודות המשחק 4](#_Toc509358764)

[מהלך המשחק 4](#_Toc509358765)

[אסטרטגיות: ( אסטרטגיות שהשחקן יכול להשתמש בהן) 4](#_Toc509358766)

[טיפוסי נתונים במשחק 5](#_Toc509358767)

[תרשים זרימת המשחק 6](#_Toc509358768)

[תיאור המחלקות 7](#_Toc509358769)

[אלגוריתם מרכזי 7](#_Toc509358770)

[מבנה המסך 9](#_Toc509358771)

[תא במבוך 10](#_Toc509358772)

[יצירת התאים במבוך והגדרת השכנים שלהם 12](#_Toc509358773)

[השרת 13](#_Toc509358774)

[מסקנות 14](#_Toc509358775)

[ביבליוגרפיה 15](#_Toc509358776)

# מבוא

בחרתי לעסוק בנושא זה כיוון שהוא משלב עבורי כלים רבים שרלוונטיים לעולם התכנות – Events, Threading, UI ועוד. כל אלו, באים לידי ביטוי ברמה גבוהה בשפת C# שבה השתמשתי כיוון שהכרתי כבר את השפה JAVA וידוע שהשפות דומות אחת לשנייה. רציתי ללמוד דברים נוספים ב-C# ועל כן החלטתי לפתח את הפרויקט בשפה זו.

במהלך שיעורי הסייבר היה לי את הזמן המתאים להשקיע בעבודה רצופה על הפרויקט, והצלחתי להגיע לרמה טובה ולהגשים את רוב המטרות. למרות זאת, במהלך העבודה נתקלתי בקשיים רבים, בעיקר של זמן:

* התקיימו מיונים לצבא שהיה צורך להשקיע זמן עבורם
* התקיימו בגרויות שונות שהקשו על הרצף התקין של פיתוח הפרויקט, ובמקרים רבים מצאתי את עצמי מנסה להיזכר את מטרתי הנוכחית, וכן היה צורך בחידוד נושאים שונים שהספקתי לשכוח.
* נתקלתי בבעיות תכנותיות שונות שהיה צורך להשקיע עליהן זמן נוסף, ועל כן התכנון הכללי של הזמן שלי היה צריך להשתנות לטובת השקעת זמן נוסף עבור הפרויקט.
* חלק מהכלים בהם השתמשתי דרשו ממני ללמוד אותם, כמו Events ו Threading

בתיק פרויקט זה אפרט את רכיבי הפרויקט שלי – באילו כלים השתמשתי, כיצד, הפונקציונליות של הקוד וכן ארד לפרטים צורת עבודתו.

# אודות המשחק

מבוך הוא חידה הבנויה ממעברים מתפצלים, אשר על הפותר למצוא נתיב דרכה. המבוך שונה מן **הלבירינת**, שבו מסלול אחד בלבד, והוא לא נועד להקשות על מציאת הדרך בו. המבוך עשוי להיות בנוי חדרים המובילים לחדרים אחרים, כפי שמצוי לעיתים קרובות במשחקי תפקידים בסגנון מבוכים ודרקונים: במובן זה, החדרים מתפקדים כמעברים.

בפרויקט זה, מימשתי צורה של **לבירינת**, שכן ישנה רק דרך אחת להגיע לסוף המבוך, ואין חדרים מתפצלים אלא דרכים אחרות שבסופן יש קירות ללא מוצא.

## מהלך המשחק

השחקן יתחיל את דרכו בפינה השמאלית העליונה של הלבירינת, שתמיד תהיה נקודת ההתחלה. השחקן ינסה למצוא את דרכו אל נקודת הסיום. כאשר יגיע אליה, יקבל הודעה שהצליח להגיע לנקודת הסיום ותינתן לו האפשרות לשחק שוב.

במקרה של 2 משתתפים או יותר – השחקן שיגיע ראשון אל קו הסיום הוא המנצח.

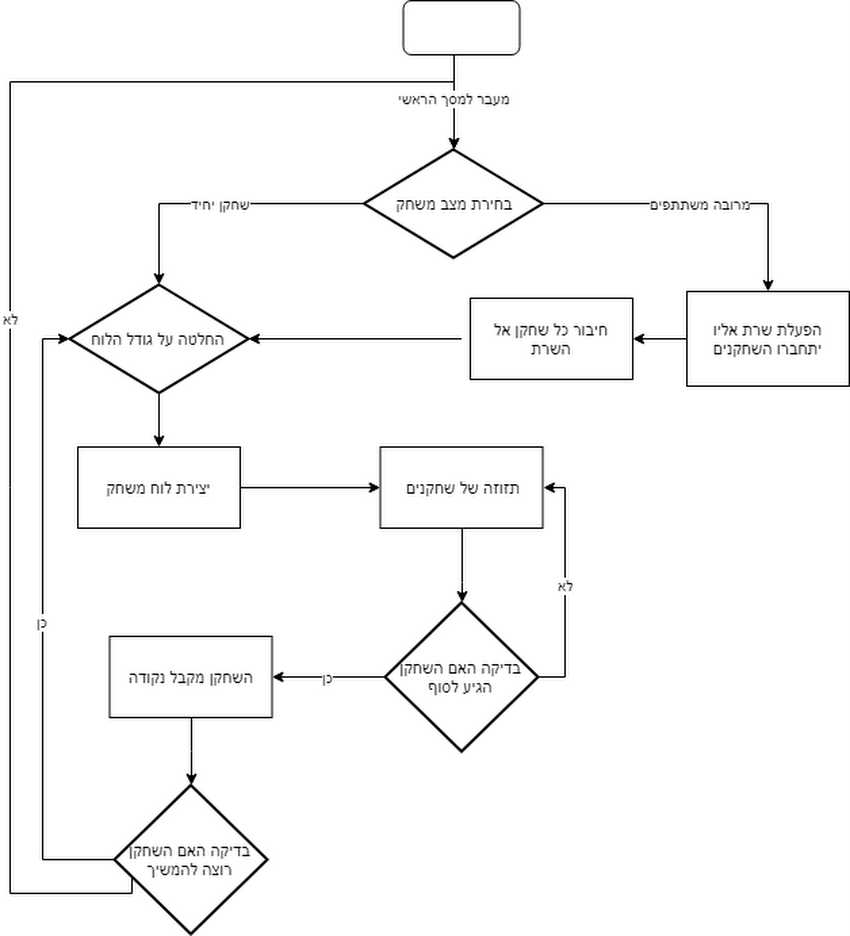
## אסטרטגיות: ( אסטרטגיות שהשחקן יכול להשתמש בהן)

כמו במבוכים רגילים, השחקן יכול להסתכל בעיניו על נקודת הסיום, ומשם לתכנן את דרכו אל נקודת ההתחלה/נקודה בה הוא יודע שהוא יכול להגיע אליה מנקודת ההתחלה. משם הוא פשוט ימשיך על אותה הדרך עד אשר יגיע לסוף.

# טיפוסי נתונים במשחק

* מערך של MazeNode אשר משמשים כתאים במבוך. לפי מערך זה ניתן לדעת את מיקום התאים במבוך.
* רשימה של חיבורים בין 2 MazeNode

# תרשים זרימת המשחק



# תיאור המחלקות

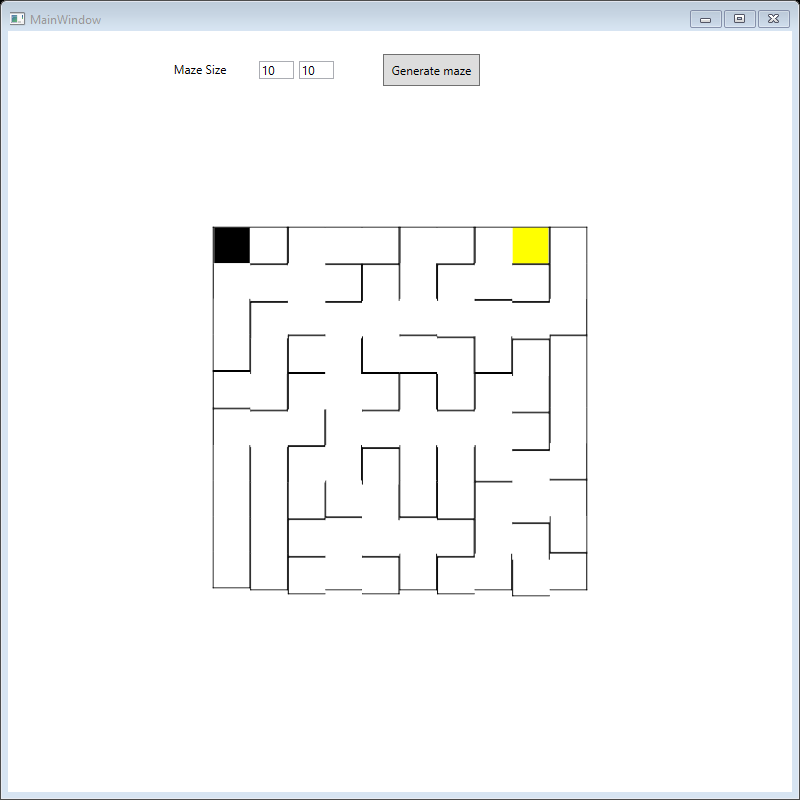
## אלגוריתם מרכזי



האלגוריתם המרכזי עובד בצורה הבאה:

הוא יוצר רשימה של חיבורים אפשריים לכל אחד מהשכנים של תא המקור, ולאחר מכן נכנס ללולאה שמתבצעת כל עוד ישנם חיבורים של תאים לטיפול. הלולאה מחברת באופן רנדומלי תאים חדשים לדרך הראשית של המבוך, ומסירה את כל החיבורים המובילים אל התאים שנוספו מהרשימה של החיבורים. לאחר מכן היא מוסיפה את כל החיבורים האפשריים לתאים חדשים לרשימת החיבורים, כאשר התאים שנוספים לא יכולים להיות חלק מהמבוך.

## מבנה המסך



המסך כולל את תצוגת המבוך, נקודת ההתחלה תיצבע בשחור, נקודת הסוף בצבע צהוב

למעלה מופיעים תיבות הטקסט האחראיות לגודל המבוך

## תא במבוך



כל תא במבוך מורכב מכמה תכונות:

* השכנים שלו (במערך של 4 MazeNode)
* מי מוביל אליו
* האובייקט המוחשי שיופיע על המסך
* הגבולות שלו, שיקבעו על פי מי מוביל אליו וכן אם הוא נמצא באחד הקירות של המבוך

לאחר מכן, מוגדרות פעולות האחראיות ליצירת האובייקט שיופיע על המבוך, וגבולות התא

## יצירת התאים במבוך והגדרת השכנים שלהם



פעולה די פשוטה, האחראית ליצירת מערך התאים במבוך, והגדרת השכנים של כל תא בהתאם למיקומו במערך

## השרת

# מסקנות

# ביבליוגרפיה