**README**

מגישים: רימה רונן, ת"ז: 315700723

רון רובין, ת"ז: 302544531

הסבר כללי של התרגיל: בתרגיל זה אנו מממשים את המשחק "שישה צבעים" תוך שימוש בספריה הגרפית SFML ושימוש במבני נתונים של הספרייה הסטנדרטית.

תיכון: אינטראקציה בין האובייקטים השונים:

ה-Controller מפעיל את חלון התצוגה ויוצר את הלוח. הלוח אחראי ליצירת "מטריצת הקדקודים", שכל תא בה מחזיק את קודקודי הצורה שתיווצר, ואת סוג החיתוך שלה (האם היא תישאר שלמה, חתוכה בצורה מאונכת או חתוכה בצורה אופקית). הוא מנהל האירועים במהלך התוכנית.

ה- Board בונה את וקטור הצורות מסוג Shape , אשר כל צורה בו יודעת להחזיק מידע אודות השכנים שלה, איזה צבע היא מחזיקה והאם היא נכבשה על ידי השחקן הממוחשב או השחקן האנושי.

ה- Player מופעל על ידי ה – Contoller ומבצע את תורו בכל זיהוי של לחיצה על צבע שאינו בשימוש של השחקן הממוחשב.

רשימת קבצים:

**Controller.h, Controller.cpp-** מנהלת את המשחק, אחראי על יצירת הלוח והצגתו תוכנו על המסך.

**Board.h, Board.cpp-** לוח המשחק והצורות, מכיר את השכנים של כל צורה ואת אופן החלוקה שלה על הלוח (האם מחולק בצורה אנכית, אופקית, מחולק לשני משולשים וכדומה..).

**Button.h, Button.cpp-** מחזיק כפתור על המסך, יודע אם הוא לחוץ או לא ואיזה צבע הוא מחזיק

**Button\_util.h-** מחזיק את שמות הכפתורים בעזרת ENUM (לפי צבעם על המסך).

**Player.h, Player.cpp-** מחלקת אב ל- ComputerPlayer ו- HumanPlayer. מנהל את תזוזת השחקנים במשחק.

**ComputerPlayer.h, ComputerPlayer.cpp-** שחקן ממוחשב. פועל לפי אלגוריתם חמדני ...

**HumanPlayer.h, HumanPlayer.cpp-** השחקן שלנו, מקבל צבע מה- Controller וכובש את שכניו הצבועים באותו הצבע.

**GameResources.h, GameResources.cpp-** מחלקה שבנויה לפי תבנית העיצוב Singleton לניהול הטעינה של קבצי מדיה (תמונות, סאונד וגופנים).

**Shape.h, Shape.cpp-** מחלקת אב לצורות המוצגות על הלוח, מחזיק לכל צורה את המידע הבא (שכנים, האם הצורה נכבשה על ידי השחקן, איזה צבע יש לה, ומה השורות שמגדירות אותה).

**Square.h, Square.cpp-** מחלקת ריבוע, יורש ממחלקת Shape, מאתחל את השורה של כל צורה

**Triangle.h, Triangle.cpp-** מחלקת משולש, יורש ממחלקת Shape, מאתחל את השורה של כל צורה לפי הנקודות שמקבל מהלוח.

**ToolBar.h, ToolBar.cpp-** מחזיק את הכפתורים על המסך, יודע לזהות אם עכבר לחוץ.

**Util.h-** מחזיקה Struct של נקודה, ובה יש 4 נקודות להחזקת צורות משולש ויהלום על המסך.

**MessageBox.h, MessageBox.cpp-** מחלקה המחזיקה את הנתונים הנכתבים על הלוח, מקבלת מחרוזות להדפסה (ניקוד השחקנים והודעת ניצחון/הפסד), מנהלת את צבע, גודל וגופן הטקסט.

מבני נתונים עיקריים ותפקידיהם:

וקטור מסוג unique\_ptr במחלקת Board, מחזיק את הצורות מעוין ומשולש המוצגות על הלוח.

וקטור של וקטור שמחזיק בכל תא std::pair במחלקת Board, כל תא מחזיקה Point ו- typeShape\_t, ובו המידע על הנקודות של הצורה, וכיצד היא מתחלקת באופן מוגרל. ( האם מתחלקת בצורה אופקית או אנכית).

מערך של שורות מסוג Std::pair במחלקת Shape המחזיק שני sf::vector2f המסמן שורה בין שתי נקודות של צורה על הלוח.

אלגוריתמים ראויים לציון:

אתחול השכנים מתבצע ב-O(n). – עובר על וקטור הצורות פעם אחת בלבד, ולכל צורה מתבצעת בדיקה האם קיימים קווים משותפים לשתי הצורות- אם כן מתווסף שכן למערך השכנים של הצורה.

באגים ידועים: לעיתים השטח של השכנים לא נכבש לגמרי לחלוטין (ישנן צורות שאינן נצבעות בצבע הנדרש – של השחקן)

לעיתים הצבע המוגרל ההתחלתי אינו מאותחל להיות שונה מהצבע של שכניו (על מנת לעשות את הצעד הראשוני של המחשב).

הערות: במשחק אנו עושים שימוש במחלקת Gameresources הבנויה לפי תבנית העיצוב singleton.

בחרנו לבטל את הכפתורים שאינם ברי בחירה במקום לסמן עליהם X

השטח של השחקן מוקף בלבן – סימן לזה שהתור של השחקן.

כאשר לוחצים על כפתור הrestart הפעולה מתבצעת אך זה לוקח זמן. (המשחק אינו נתקע כפי שאפשר לחשוב. לראייה אפשר להקטין את גודל הלוח וללחוץ על כפתור הריסט)