**OOP, ex4  
Digger Editor – using SFML**

-----------------------------

**סטודנט**:

Harel Savitzky - 305390692 - הראל סביצקי

-----------------------------

**הסבר כללי:**

בתוכנית הנוכחית נבנה עורך שלבים למשחק דיגר אשר יצרנו בתרגיל מספר 2.

להבדיל מתרגיל 2 השלב לא יהיה מסוג NXN אלה מסוג שורות ועמודות (עלול להית מלבני).  
  
נבנה עורך שלבים בעזרת הספריה הגרפית.  
נפתח חלון וניצור סרגלי כלים, סרגל כלים בעזרתו נוכל למקם אובייקטים במסך וסרגל כלים נוסף שיכלול את האפשרות למחיקה נקודתית, לשמירה שלב השלב שערכנו לקובץ, או בעזרת כפתור הניקוי למחיקת כל המסך וקבלת נתונים מחדש (שורה עמודה ומספר אבנים).

נדגיש את המיקום בטבלה בו ההשמה של האובייקט תתבצע, נדגיש (בעזרת צבע אדום או ירוק) אם ההשמה חוקית (לא יכולה להתבצע על אובייקט אחר), נדאג שהשמת הדיגר יכולה להתבצע רק פעם אחת, בכל השמה נוספת ההשמה הקודמת תמחק.

\*בפתיחת התכנית ננסה לקרוא שלב מקובץ, במידה ואין קובץ לקרוא ממנו, המשתמש יזין ערכים רצויים בסדר הבא (עם רווחים ביניהם):

1.מספר שורות

. 2.מספר עמודות

. 3.מספר אבנים מותרות לאכילה.

\*\*\* חשוב לציין כי בהזנת ערכים (יש גם קלט המסביר על זאת) הערכים המקסימלים לקבלת עורך אופטימלי, אשר לא יחרוג מהמסך, הם 20 שורות ו40 עמודות.

(בכדי שכל משבצת לא תהיה קטנה מדי, ואז זה פשוט נראה מגוחך, גם הגודל הנוכחי הרגיש לי קטן מדי, בתחילת העבודה על התכנית קבעתי משבצות של 150 על 150)

(משמע זו לא בעיה בקוד אלה שיקולים כדי שהעורך לא יראה מצחיק וקטן בצורה מוגזמת, ניתן לשנות את הקבוע SLOT\_SIZE ואז לבחור מספר גדול יותר של שלב, הנחתי ש20 על 40 זה דיי ויותר ובכך גם לא יפגע מדי באיכות הנראות של העורך)

ליציאה כמובן שנלחץ על האיקס אשר סוגר את החלון, במידה ושמרנו קובץ בעזרת הכפתור, הוא ישמר, במידה ולא שמרנו ונריץ שוב את התכנית (אשר לא קראה קובץ מלכתחילה) אז נתבקש להזין נתונים בשנית

**קבצים שנוצרו ע"י הסטודנט:**

EditorManager.cpp

EditorManager.h

Slot.cpp

Slot.h

Board.cpp

Board.h

Object.cpp

Object.h

כמובן שגם קובץ main.cpp

ובנוסף תיקיית resources אשר מכילה את קבצי התמונות ואת גופן הטקסט לכותרת השלב

-----------------------------

**מבנה נתונים בשימוש:**

ווקטורים לשמירה ייצוג השלב בצורת טקסט או לייצוג בחלון הSFML.

-----------------------------

**אלגוריתמים ראויים לציון:**

יצירת חלון המשחק מורכב באופן הבא:

sf::RenderWindow window(sf::VideoMode(((windowCol + 2) \* SLOT\_SIZE) + VERTICAL\_GAP + (windowCol \* SLOT\_GAP),

((windowRow + 2) \* SLOT\_SIZE) + VERTICAL\_GAP + (windowRow \* SLOT\_GAP)),

windowCol/WindowRow מייצגים את כמות השורות והעמודות (כמות העמודות גם לא תהיה קטנה מגודל סרגלי הכלים, לכן אם הכניסו קלט נמוך, הגודל המינימאלי יקבע לגודל סרגל הכלים, בהתאם לתנאי מקדים לפקודה הנ"ל)

SLOT\_SIZE מייצג את גודלה של כל משבצת , SLOT\_GAP מייצג את הרווח בין המשבצות (בכל צורת בניה ללא רווח בין משבצות עלול להווצר מצב בו לחיצה על הגבול, או מעבר, יסמנו שתי משבצות, בנוסף חשבתי שזה הרבה פחות אסטטי כשהן צמודות)

VERTICAL\_GAP מייצג את הרווח בין המטריצה לסרגלי הכלים.

האלגוריתם הנ"ל חוזר בנקודות שונות הרי מספר אובייקטים ומצבים צריכים להתחשב במיקומים הנ"ל.  
בנוסף קיים קבוע HORIZONTAL\_GAP אשר מתבטא בבניית סרגלי הכלים והרווח ביניהם.

אלגוריתם נוסף הראוי לציון:

במידה והשחקן ילחץ על כפתור הCLEAN (אשר הלוג שלו הוא פח זבל) לאחר מספר פעולות ועדכוני משתנים אשר קריטים לפעולה הזו, תתבצע קריאה רקורסיבית בחזרה לפונקציית הrunEditor() אשר מפעילה ומנהלת את כל עורך השלבים – חשבתי על זה ברגע האחרון והיה נשמע לי שזה רעיון מוצלח, לא בטוח לגבי מידת הנכונות מבחינת תכנות אבל היה נשמע לי רעיון מקורי ונחמד!.

**באגים ידועים:**

אין באגים ידועים

-----------------------------

**הערות נוספות:**

מחלקת הSlot – משבצת – הינה מחלקת הבסיס שלי, בה נבנית משבצת בסיסית ביותר ובה קבעתי את כל הCONSTים הרלוונטים לתרגיל. מחלקת הBoard ומחלקת הObjects יורשות ממנה. בעזרתה מייצרות ריבועים בסיסיים, לדוגמא, אם זה במחלקת הBoard ליצירת ייצוג המטריצה ב SFML ,או במחלקת הObjects ליצירת אובייקטים.

מחלקת הBoard -(יורשת מSlot)מורישה לEditorManager. קובעת, מעדכנת, יוצרת ומחזירה לנו את ייצוג הלוח בצורת הTXT וייצוג הלוח בצורת הSFML.

מחלקת ה-Objects (יורשת מSlot) קובעת, מעדכנת, יוצרת ומחזירה לנו אובייקטים להדפסה ואת סרגלי הכלים.

מחלקת הEditorManager- (יורשת מBoard) מנהלת את כל עורך השלב ואת כלל התוכנית בכל צורה שהיא, מפעילה פונקציות ומעדכנת את Board בעזרת ירושה, ומקבלת בחזרה אובייקטים שנבנו בObjects בעזרת member מהסוג שלו. היא מנהלת את הכל, ומגשרת בין מחלקת הלוח, האובייקטים ועצמה.  
  
  
 **SLOT**

**Objects** **Board**

**EditorManager**