***כותרת:***

בניית שלבים באופן גרפי למשחק "BomberMan".

===================================================

**פרטי סטודנטים:**

שמות:

1. יעקב בונדר, ת"ז: 322336488.
2. שלי צ'אוסובסקי, ת"ז: 323998526

**הסבר כללי על בניית השלב (בעזרת ממש גרפי SFML):**

כללי:

בהרצת התוכנית, במידה ולא קיים קובץ חיצוני אשר אליו נכניס את המידע שהמשתמש הזין, אזי יוצג למשתמש שעליו להכניס ערכים עבור מספר השורות והעמודות (תחילה יש להזין את העמודות, ורק אחר כך את השורות). מספר העמודות חסום בטווח [33, 1] ומספר השורות חסום בטווח [15, 1], זה נועד בשביל למנוע מהטבלה לזלוג מגבולות מסך המחשב. במידה והקובץ קיים כבר, תיפתח מיד אפשרות לערוך את הלוח הקיים מבלי להזין את גודל הלוח.

ייצוג כפתורים (ToolBar):

כל דמות מיוצגת על ידי תמונה באופן הבא:

רובוט -

A metal door with blue lights

Description automatically generatedדלת -

A person in a uniform with an object and a mask

Description automatically generated

שומר -

A black metal structure with many columns

Description automatically generated with medium confidenceגבול –

A close up of a ball

Description automatically generated

אבן –

A black and white button

Description automatically generatedמחיקת אובייקט -



מחיקת לוח -

A black circle with white text

Description automatically generatedשמירה –

הסבר על כל כפתור

1. רובוט, דלת, שומר, גבול ואבן - הינם אובייקטים שניתן להניח לפי רצונו של המשתמש על הלוח (המיקום יהיה אך ורק בתוך הטבלה שהמשתמש בחר). כל בחירה של אובייקט תתרחש באופן הבא: תחילה יש ללחוץ על אחד מהכפתורים לעיל, ואז ניתן ללחוץ כמספר הפעמים שהמשתמש רוצה בכדי למקם את אותו האובייקט.
   * רובוט ודלת יכולים להופיע בדיוק פעם אחת. אם יתבצע ניסיון להזין את אחד מהם יותר מפעם אחת, אז המופע הקודם שלו יימחק, והמיקום החדש יכיל את האובייקט.
2. מחיקת אובייקט – כמו עם הכפתורים הנ"ל, נלחץ על כפתור המחיקה, ואז כל מקום שנבחר על הטבלה (הריבועים המוצגים), אם קיים שם אובייקט כלשהו, אזי האובייקט יימחק.
3. מחיקת לוח – בניגוד לכפתורים שהצגנו עד כה, לחיצה על מקש זה תוביל לביצוע מיידי. הלוח הנוכחי יימחק, וכך גם הקובץ, ובמקום תיפתח אפשרות למשתמש להזין קלט חדש אשר ייצג את גודל הלוח החדש. לאחר מכן, יופיע לוח חדש ריק, בגודל שהמשתמש הזין.
4. שמירה – כמו מחיקת לוח, כפתור זה הינו כפתור מיידי. לאחר לחיצה על הכפתור, הטבלה תיצבע בירוק ל-2 שניות וכך המשתמש יוכל לדעת שהלוח שלו נשמר לקובץ בהצלחה.

**רשימה של קבצים שיצרנו:**

Button.h – מכילה את הגדרת המחלקה Button.h.

Button.cpp – מכילה את הבנאים והפונקציות השונות של מחלקת Button.h.

Tile.h– מכילה את הגדרת המחלקה Tile.h.

Tile.cpp – מכילה את הבנאים והפונקציות השונות של מחלקת Tile.h.

CustomizeWindow.h– מכילה את הגדרת המחלקה CustomizeWindow.h.

CustomizeWindow.cpp – מכילה את הבנאים והפונקציות השונות של מחלקת CustomizeWindow.h.

Toolbar.txt – שם יוצגו סדר הכפתורים עבור רובוט, דלת, שומר, גבול ואבן (בר שינוי).

clear.jpg – תמונה המייצגת את הכפתור מחיקת הלוח.

door.jpg - תמונה המייצגת את כפתור הדלת.

eraser.jpg - תמונה המייצגת את כפתור המחיקה.

guard.jpg - תמונה המייצגת את כפתור השומר.

robot.jpg - תמונה המייצגת את כפתור הרובוט.

save.jpg - תמונה המייצגת את כפתור שמירת הלוח.

stone.jpg - תמונה המייצגת את כפתור הסלע.

wall.jpg - תמונה המייצגת את כפתור הגבול (קיר).

**מבני נתונים עיקריים ותפקידיהם**

1. **וקטור המכיל וקטורים המכילים טיפוס** TILE:
   * **וקטור זה מסמל מטריצה אשר תכיל את כל הטבלה שהמשתמש מעדכן.**
2. **וקטור המכיל טיפוס** Button**:**
   * **וקטור זה מסמל את כל הכפתורים אשר מוצגים למשתמש. בנוסף כל כפתור יכיל את ה –** Texture **שלו, אשר יוטען בדיוק פעם אחת בתוכנית.**

**אלגוריתמים הראויים לציון:**

**תיכון (design):**

**האובייקטים המרכזיים הינם החלון, המטריצה והכפתורים.**

1. **חלון – אחראי להצגת כל הצייר (המטריצה והכפתורים), ונועד לקבל עדכונים דרך** SFML **על כל פעולה שהיא (במקרה שלנו, לחיצה עם העכבר במקום מסוים, או ניסיון סגירה של החלונית). כל פעולה שרלוונטית למטריצה, החלון מעדכן אותה לפי האירוע הרלוונטי.**
2. **מטריצה – המטריצה הינה הטבלה השומרת את כל הבחירות של המשתמש. החלון כל שינוי מדפיס את המטריצה, בכדי להראות למשתמש את השינויים אותם הוא ביצע. כל תא במטריצה מקבל (במידת הצורך) את ערך התמונה שלו, על ידי הכפתורים.**
3. **כפתורים – נועדים בכדי להראות את אפשרויות הבחירה עבור המשתמש, ושמירה על התמונה אשר הם מציגים. בכל עת שיש לשלוף פוינטר (**sprite**), נוכל לקבל אותו דרך הכפתור המכיל אותו (לדוגמה, אם יש להכניס תמונה לתוך תא במטריצה).**

**באגים ידועים:**

**הערות נוספות:**

לאחר ייצור הקובץ, ניתן יהיה למצוא אותו בכתובת הבאה: out\build\x64-Debug.