**מכללת​ ​הדסה,​ ​החוג​ ​למדעי​ ​המחשב**

**תכנות מונחה עצמים ופיתוח משחקים**

**סמסטר​ ​ב',​ ​תשף פרויקט**

* פרטי סטודנטים :

עבדאללה גולאני, ת.ז : 206461394

* הסבר כללי על התרגיל :

מימוש של משחק BomberMan בשפת C++ וב- SFML

* קישור מראה סרטון למשחק :

<https://www.youtube.com/watch?v=3smytj9Bu_E&t=295s>

* הזכרתי בבחירת המשחק שאני לא ממש תזוזה למסך
* אופן התנהלות המשחק:
* הורדת פצצה מתבצעת ע"י לחיצה על כפתור X.
* אם השחקן נגע באש אז הוא נפסל
* אם השחקן שם פצצה אז הוא לא יכול לדלג עליה.
* כדי לסיים את השלב צריך להרוג את כל המפלצות ולמצוא את הדלת לפני שהטיימר יסיים.
* יש במשחק שני סוגים של מפלצות אחת זזה בצורה אקראית
* ומפלצת חכמה שמחשבת את דרך הקצרה ביותר במידה והמרחק בינה לבין המשחק 6 תאים, אחרת היא זזה רנדומלית, והיא זזה לאט ( בשל שלא יהיה קשה לשחק )
* בהתחלה מותר להוריד פצצה אחת אחרי קבלת בונוס אפשר להוריד יותר.
* יש בונוס לאש שהeffect שלה יהיה יותר גדול
* הבונוסים ממשיכים עם השחקן כל עוד שהוא לא נפסל.
* קבצים נוצרים ע"י הסטודנטים הנ"ל + תיכון האוביקטים:
* **Main.cpp**

Initializes game menu ,settings menu and Calls for an object of the type Play to run the program .

**Abstract classes:**

**MainMenu,**

**MenuOptions,**

**SettingOptions**

**Menu :**

**It has a register function that registers commands inside the menu in order to run them later.**

**Settings<Menu>:**

**Template class that inherits from menu of type T and from MainMenu abstract class , it has back button and a register function in order to register an object of type submenu to a Menu of Type T,it was made for implementing settings menu .**

**Play<GameType = Y>:**

**Label that adds command in menu of type T in order to run game of type Y**

**Help:**

**Prints constructions in order to know how to play the game .**

**Exit:**

**Exits the menu .**

**SoundPlayer:**

**Switches sounds on and off in the menu.**

**Resolution<T>:**

**Takes the type of the singleton window it wants to resize**

**//-------------------------------------------------------------------------------------------------------------**

**Window :**

Derived class from RenderWindow that is singleton and has some facilities

**CollideManager:**

Handle collisions between the characters in the game by multiple dispatch algorithm

**Resources:**

Class that is useful for getting sound buffers ,music,fonts & sprites from it .

**AnimBase**: class to handle the animation of the entities

//--------------------------------------------------------------------------------------------------------------

**LogFile :**

Functional in writing game exceptions to file.

**Board:**

Class that reads the objects from file and put them in data struct

And returning it to the stage loader

**StageLoader:**

Loading the stages from the board and returning a stage to play

**Stage :**

Getting a Stage to play , handling the movements ,the collisions,

And the stage statuses

**Game:**

Plays stage by stage from stage loader.

//-----------------------------------------------------------------------------------------------------------------------

**ENUMS :**

**GameState:**

Tells the state of the current game ,from winning to losing .

**GameObjects:**

Saving the objects characters in order to read them from file .

//---------------------------------------------------------------------------------------------------------------------

**definitions :**

It has some definitions of sprite images, background and files directories and paths .

It has also some text definitions.

* מבני נתונים עיקריים ותפקידיהם :

1. I used **unordered maps** of pointers in resources.h&.cpp ”resource manager” in order to save and retrieve textures and sound buffers every time we need them and to destroy all of them at once in destructor .

it’s insertions finding and extractions time complexity is O(1) because they are based on hashing

1. **deque** for containing menu's commands
2. vector for saving the board.

* אלגוריתמים הראויים לציון :
* Polymorphism.
* multiple dispatch.
* Command pattern
* State pattern
* Singleton.
* Distance and path finder according to bfs algorithm
* I overloaded some operators from sfml library classes and implemented additional functions in order to ease the use of the code .
* A lot of algorithms
* באגים ידועים :

לא יודע