

# Block Shooters

By: Jordan, Vivian, and David

GitHub Repo: <https://github.com/SairenDelight/block-shooters>

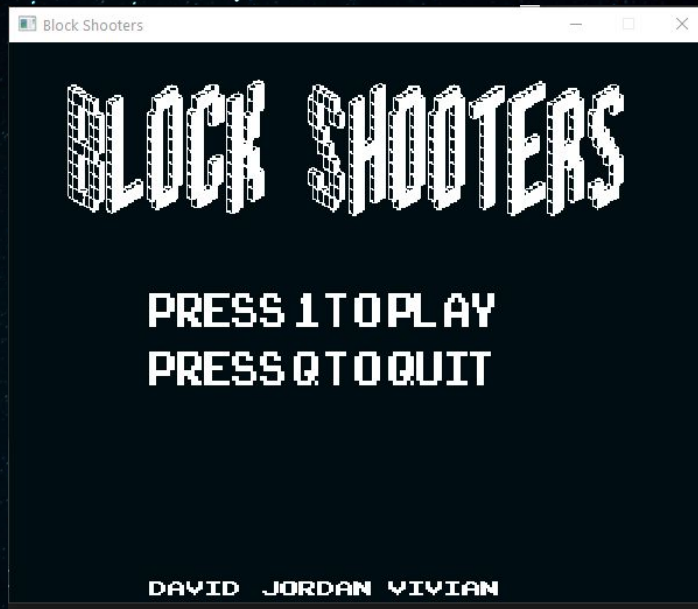
# Members

- Jordan
- Vivian
- David



# Game

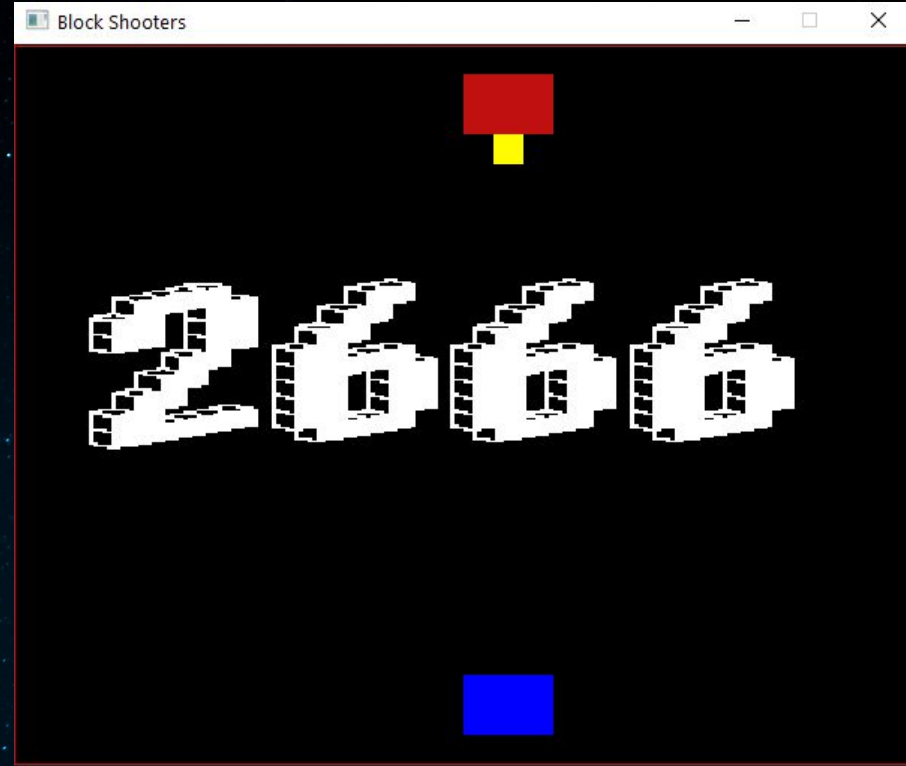
- Simple Shooter
- Protagonist
- Enemy





# Gameplay

Block Shooters is a 2-player fixed shooters game where both blocks are able to shoot but can only move in a horizontal direction. While the first block is at the bottom of the screen, the second block is at the top of the screen. The objective of the game is to shoot the opponent in order to win the game.



# Controls



## Player 1:

A - move left  
S - shoot  
D - move right

## Player 2:

J - move left  
K - shoot  
L - move right

# Concepts Used

- Switch statements
- Functions
- Pointers and malloc()
- Input/Output
- One dimensional arrays
- Constants and Variables
- If/else statements
- Enums
- Prototypes
- File Open - Writing
- Bit Shifting/Manipulation
- Loops
- Bubble Sort
- Structures

```
21 int start_menu() {
22     bool initialize = init(); // This will create the window and renderer
23     if (initialize == false) { // If creation fails, program will exit with code -1
24         return -1;
25     }
26     else {
27         Game_State state; // The game state is initialized (will need to implement start m
28         state = PLAY;
29         do {
30             switch (state) {
31                 case PLAY:
32                     state = game_loop();
33                     break;
34                 case MENU:
35                     state = show_menu();
36                     break;
37                 default:
38                     state = STOP;
39                     break;
40             }
41         } while (state != STOP);
42         close(); // Clean up
43         return 0;
44     }
```



# Conclusion and Future Improvements

- Add names to the scores
- Use a better scoring function
- Use a better sorting function instead of Bubblesort
- Use C++ so we can use classes for cleaner code
- Add greater delay to users' ability to shoot
- More add more functions and files to clean up the code and improve readability
- A better graphical user interface with a tutorial
- Add more documentation regarding about what each function does
- We learned about the necessity of documentation and organizing code - as well as how to use the SDL2

# References and Libraries Used

- <https://wiki.libsdl.org/Introduction>
- <http://lazyfoo.net/tutorials/SDL/>
- [https://www.libsdl.org/projects/SDL\\_ttf/](https://www.libsdl.org/projects/SDL_ttf/)
- [https://www.libsdl.org/projects/SDL\\_image/](https://www.libsdl.org/projects/SDL_image/)
- Stackoverflow occasionally for solving errors and creating multiple C files...

## Libraries:

- SDL2 Graphics library
- SDL2 True Type Font
- SDL2 Images