**PROJECTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **PROJECT NO.** | **NAME** | **ROLE** | **DESCRIPTION** |
| 1 | GAME CODE | MAIN PROJECT | HOLDS THE MAIN GAMEPLAY LOOP AND FUNCTIONS |
| 2 | GAME SCREEN | SCREEN & TRANSITIONS | HOLDS THE INDEPENDENT SCREENS (CONTROLS, GAMEOVER AND TITLE) TO SWITCH BETWEEN |

**FUNCTIONS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NAME** | **PARAMETERS** | **RETURN TYPE** | **DESCRIPTION** | **PROJECT LOCATION** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**VARIABLES**

|  |  |  |  |
| --- | --- | --- | --- |
| **NAME** | **TYPE** | **DESCRIPTION** | **PROJECT LOCATION** |
| playerDead | BOOLEAN | STATES WHETHER THE PLAYER CAN STILL PLAY (“IS ALIVE”) | 1 |
| platform | STRING ARRAY | HOLDS THE ASCII PLATFORM (-----) | 1 |
| platformObj | STRING ARRAY | HOLDS THE ASCII OBJECTS (X) | 1 |

**STRETCH GOALS**

|  |  |  |
| --- | --- | --- |
| **NAME** | **TYPE** | **DESCRIPTION** |
| OFFSCREEN HAZARDS | GAMEPLAY | HAVE “BOMBS” (<--) COME FROM OFFSCREEN (LEFT OR RIGHT) AND DESTROY A PART OF THE PLATFORM (MAKE THE AVAILBLE SAFE AREA SMALLER) |
| SOUND EFFECTS | ENHANCEMENT | HAVE SOUNDS PLAY FOR PLATFORM LOADING AND JUMPING |
| COLOURS/ASSETS | ENHANCEMENT | ADD COLOURED ASCII TO MAKE ASPECTS OF THE GAME STAND OUT/PROVIDE CHARACTER COLOUR CUSTOMABILITY OPTION |
| SCORE | GAMEPLAY | ADD A SCORE TO THE TOP OF THE SCREEN RECORDING HOW MANY PLATFORMS THE PLAYER ACSENDED |
| DIFFICULTY SPIKE | GAMEPLAY | CODE THE OPTION TO GRADUALLY DECREASE THE PLATFORM WIDTH OR THE AMOUNT OF OBSTICLES LOADED TO INCREASE THE DIFFICULTY |

**All suggestions bellow can be found referenced in tutorial 5**

* Can use MessageBoxA(0, "*[text to display]*", "*[dialogue box title name*", *dialogue box type]*); to display a pop-up box displaying an error stating that a potential input is invalid
* Can use while (true) { HDC dc = GetDC(GetDesktopWindow()); Rectangle(dc, 100, 100, 200, 200); } to draw shapes
* Can use inline [function type] [function name] to speed up processing by having short functions (i.e simple calculations or print statements) to be ran as part of the main code rather than shifting the control flow to handle the function sepereatly
* Can pass structs by reference instead of by value to demonstrate function control
  + STRINGS are recommended to be passed
  + Can also state the passed parameter as [function](const [parameter]) to save publishing time by preventing a large variable/structure from being copied
    - Recommended to pass parameters as const if not going to alter them
* Can look at overloading the function to provide more utility
  + i.e you could overload a function to return an int, float or double depending on the type of parameters given
    - could look at adding to this by using template functions (prevents the need to write out several overloads)