Gold Miner

1- Backgrounds
- Different backgrounds are displayed in different game states
- Background Animation - Clouds move with time implementing three functions

GameAction(), BackgroundAnimtaion(), RenderImageForBA(ImageData & image)

- User name input animation

2- User Controlled Moving Object
- Hook

Drawn using five white lines

Have five functionalities

1. 160 degree rotation
2. drop and catch objects
3. get objects back / hit the wall back
4. trigger the bomb
5. explode pig

3- Multiple States/ Stages
- 9 States
- 10 Levels - Different types/number of objects and goals are set up in different levels

4- Automated Moving Objects
- Stable Objects (Gold, Stone, Diamond) - They are stable when nothing happened, but will move with hook when they are caught
- Lucky pigs - Moving around
- Diamond stamp pigs - Moving around
- All of them will display the score when they are caught to the root of the hook(And the score is actually an object)
- Hook do rotation and move at proper time

Class hierarchy (Showing inheritance, the behaviours of each classes varies)
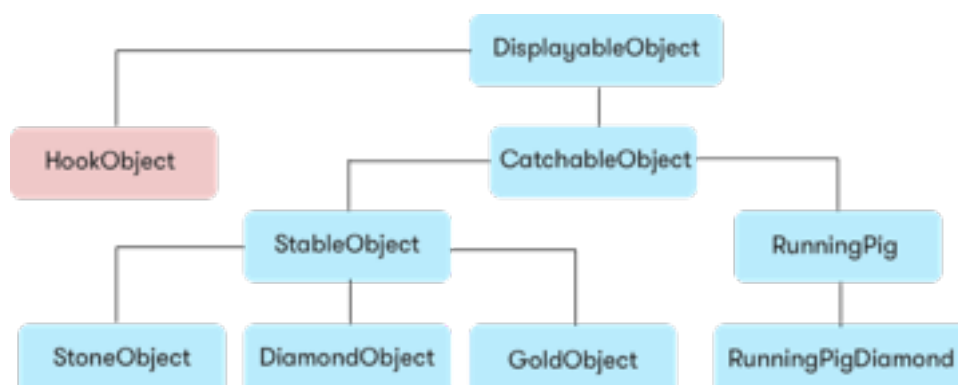
5- Tile Manager

- Bomb and explosion - When the tile in the central (the bomb) is toughed by the hook, 8 tiles around this tile and also the central tile display animation of the explosion. It also informs the nearby stable objects to disappear, and scares the pigs.
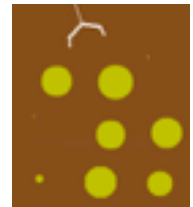
6- Load and Save Data

- Top 10 player names, scores and levels of are saved into the file and loaded for display. Data are stored in format NAME.   SCORE-   LEVEL
- The file is updated when the player ends the game and also after every time displaying the score to make sure the the file only records top ten
- Scores are displayed in ascending order

7- Display text on the screen

- User name input (actually is drawing the background)
- Level, time, score and goal information (change with time, level and objects got)
- Objects scores are displayed when the objects are caught to the root of the hook
- Lucky pigs have changeable value displayed on the head

8- Interaction

All the objects on the screen have interaction with each other

- The hooks catch objects and trigger the bomb
- The bomb explodes and makes the nearby stable objects disappear
- The stable objects block the moving of the pigs
- Pigs detect for red bomb explosion and user "up key" bomb
- The diamond stamping pig detects for diamonds and stamps them
- The red bomb explosion scares the pigs
- The score is added when the objects (draw like the strings) disappear

…

9- Algorithm and AI

- Uniform distribution algorithm - Ensure relatively uniform distribution of the stable objects in each level
- Uniform-cost Search Algorithm - One special pig will search for the nearest diamond and stamp 20 times (seconds) on it to make it disappear

10- Container Class

- Unordered-map is implemented here with reasons. As the types and the numbers of the objects varies in different levels, two special classes of the objects (Hook and Diamond stamping pig) are stored in the unordered-map with special keys. In that case, they can be gotten using the key without considering how many objects are there in each level and how are they ordered, as the keys of those objects are also useful in other functions.
- This functionality works for multiple classes such as GoldManEngine.cpp, RunningPigDiamond.cpp, StableObject.cpp and also its child classes.