Team Name: CGEZGMG

**Team project: SpaceSwim**

Member 1

ID: 2019314505

Name: Jaesung Park

Department: Software department

Member 2

ID: 2019311680

Name:

Department: Software department

1. Creativity in the design and scenarios
2. Humor and fun
3. Fidelity of implementations
   1. title screen  
      Using imgui, we make title screen. You can start game by click the button.
   2. graphical (no text-only) help with ‘F1’ key  
      When you press F1 button, it shows you how to control this game. It is made by imgui.
   3. in-game reset (with ‘R’ key) or continue  
      If you had got save point and press R key, then your character revise at save area.
   4. multiple difficulty levels  
      In title screen, you can select multiple game mode.
   5. 3D shading  
      We adopt directional light because the theme is the universe.  
      And, we use Phong illumination model.
   6. resizable window and its constant aspect-ratio viewport  
      Even if you resize the window, the program shows you proper game screen because it renders it according to window size.
   7. text rendering  
      When we started this project, we implement text shader. But we found more useful tool, that is imgui. So, our text rendering is changed to imgui.
   8. sound rendering  
      Using irrKlang library, we implement background sound and sound effect.
   9. textured 3D skybox/cylinder/sphere  
      To make universe background, we make rolling cylinder textured as Milky way.
   10. dynamic 3D camera movement  
       When your character move, then you can see camera movement.   
       We can add trackball, but we didn’t it. The reason is funny of game.
4. Hand-drawn 2D images you created on your own.  
   We make hand-write textures such as “radioactive.png”, “spacehead.png”.
5. Sound clips you created on your own.  
   We make background sounds that are “1 - Quite Space.mp3”, “2 - Can you swim... in space.mp3”, “3 - Space Swim!.mp3”, “4 - Too Dangerous To Swim.mp3”.
6. Shadows  
   To implement it, we invested three days….  
   it is too hard than our expectation, and there are so many bugs in implementation process.  
   To render shadow, it measures depth from light, and draw depth in FBO. Next, by some calculation and some technics, it applies shadow discretely.
7. Light reflections  
   Although we fail implement ray tracing, but Phong illumination method is also nice light reflection technic.
8. Particle systems  
   When meteors are hit by bullet of player, then the meteor disappear, and it burst out its particles. After two second, the particles also disappear.
9. Physics on gravity and acceleration  
   By Bullet3, we implement gravity, inertia, velocity and acceleration. By collision detection, we control events.
10. Moving 2D/3D NPCs with AI  
    UFO enemy shot bullet, so if player is hit, then the player’s HP is decreased. When we started this project, we planned to make RL-AL coded by python and it communicate by socket. However, it is too hard. Therefore, the enemy is made by rule-based AI.
11. Deformable 2D/3D character animation  
    The player’s character is consisted of some object. There are head, arms, body….  
    And, each part move according to the character’s direction.
12. Design Pattern
13. Discussion
14. Test Environment 1
    1. AMD 4750g (CPU/GPU)
    2. Window10
    3. Visual studio 2019
15. Test Environment 2