IMGD 3000

Names: Frank, Jordan

Game Name – Umbra

Team – Team ABCDEFG - Frank, [ypiao@wpi.edu](mailto:ypiao@wpi.edu) Jordan, jamaraldasilva@wpi.edu

Genre – classic platformer

Game Description – One day, one guy was enjoying the twilight when suddenly a dark portal opens ups in the air in front of him, and drags him into it with a mysterious power. He then hears an ominous voice telling that he is now in the Realm of Umbra and he must past through some trials to get back to his original world. The guy would later discover that the trials include parkour through tons of platforms, fight weird enemies and try see anything in that pitch black dimension.

Technical Features –

Gravity, jumping and shooting.

A variety of obstacles, monsters

Artistic Assets –

**Main character: under /hero**

hero-jump-spr:

frames 1 width 3 height 3 color green

hero-walk-left-spr: frames 2 width 3 height 3 color green

hero-walk-right-spr: frames 2 width 3height 3 color green

**Bullets:**

bullet-spr: frames 2 width 3 height 1 color gray

bullet1-spr: frames 2 width 3 height 2 color cyan

**Monsters:**

madsquare-spr frames 2 width 7 height 4 color magenta

suicider-spr frames 9 width 11 height 1 color yellow

trap-spr: frames 4 width 3 height 1color red

**Win condition:**

winflag-spr: frames 2 width 5 height 3 color red

**Special effects:**

explosion-spr: frames 8 width 5 height 3 color red

gameoverflag-spr: frames 10 width 54 height 7 color default

gameover-spr: frames 10 width 54 height 5 color cyan

gamestart-spr: frames 2 width 70 height 9 color default

gamewin-spr: frames 10 width 54 height 7 color default

technical challenges:

1. Drawing customizable platforms required special algorithm.
2. Monsters have different behaviors, which required the use of virtual functions.
3. Animating the main character required a binary state flag and animation updater.
4. Incorporating jumping with platform movements required major changes in the engine movement handling procedure to expose more information about collision, object movement, and states.
5. Using visual studio debugger to change memory info of platforms and other objects to edit the level and check result in runtime.

Distribution of Work – Evenly distributed, constantly sharing screen with each other, and both have worked on all aspects of the technical challenges.

Schedule -

10/6 – Setup workplace, draw platforms and handle collisions, load “stage map” from .txt file.

10/9 – Fix any possible bugs, draw character and enemies with different move patterns

10/12— Load every art asset from file, set up interface and a basic level.

10/13 – fix bugs, polish the game.

10/14 – Final checking & submission.

Our alpha has all the major features implemented. We set up gravity, shooting, traps, drawing background and drawing platforms.

In the final version, we added more enemies instead of only having a trap. We completed the level, incorporated many moving platforms and interesting level design, and added interesting enemies.