CSE251: Graphics - Spring 2013:

Assignment 2: Moving in a 3D World

Due Dates. Part 1: Jan 28th, 7 pm. and Part 2: Feb 8th, 7pm.

1 The Problem

This time, we will make a 3D world. The world will be an adventure course where a person has to go from their current position to a destination, while avoiding obstacles. During this journey, your program should allow the action to be viewed from multiple positions, which is controllable from the keyboard/mouse. The assignment submission is divided into two parts. The first part (2.1), which is due before Mid-1, includes creating a moving person (or robot) with limited camera views and a simple world. Part-2 (2.2) will be the final submission, which will include the complete obstacle course and all camera views.

The requirements mentioned below is the minimum set. It is recommended that you include additional objects/elements in the world according to your imagination and ability.

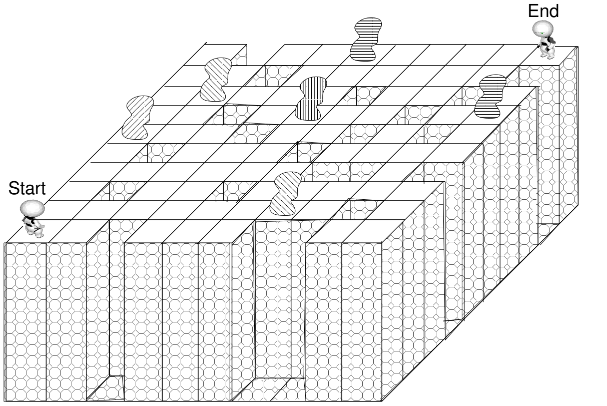

Figure 1: Mock-up of a sample world with missing and blocked tiles and the start and end positions of the person.

2 The 3D World

The world consists of a flat square tiled land of 10 × 10 tiles, surrounded by water. You start on $Tile_{1,1}$ and has to reach $Tile_{10,10}$ (see Fig. 1). Some of the tiles are missing, and if you step on these spots, you will fall down and the adventure will be over. Some of the tiles keep moving up and down. You cannot jump too high or too deep, and jumping too deep will cause injury and game termination. Some of the tiles contain blocks, and trying to move on to these tiles will result in a collision and you will remain in the current tile.

Figure 1 is just a rough mock-up. So do apply your imagination and ingenuity to come up with a better looking world.

2.1 Character Specifications

- The character which you will create and control should be humanoid, i.e., have a head, torso, arms and legs.

- How complex (and beautiful) you make your character is totally upto you. It can be made of simple 3D primitives –spheres, cuboids, cylinders etc - or you can create more interesting models using blender/3dsmax etc.

- When moving (walking/running), the character's hands and legs should move. We aren't expecting exact animations. It should just appear as if the charecter is walking when he moves.

- The character moves and jumps using user controls through the keyboard.

2.2 Camera, Lights, Action

The primary character is a human(oid) who walks, runs and jumps around in the world. The motion of this character should be controllable by the arrow keys and jumping is triggered by the j key. Around 50% of the tiles move up and down as described before. These tiles are decided randomly when the program is started.

The world also obeys laws of physics gravity in the sense that the humanoid will fall down if it steps on a missing tile, and will bounce back if you move into a block. Similarly, if you step into a tile that has moved up, you either fall down beneath it or run into a wall as per your own definition.

There is a small amount of ambient light in the world. In addition, there is a spot light mounted on a tower at a corner of the world, and illuminating the tile occupied by the persona and its neighborhood. This light will follow the character as it moved in the world.

There are different camera positions and the player can select the view point from the keyboard.

1. *First Person View:* View from the head of the character, where one can see the world to the front where the character is facing.
2. *Third Person View:* This is basically a ”smooth chase camera”. The camera is at a slight distance behind the character and slightly above, and smoothly follows the character.
3. *Tower View:* A view of the world from atop the tower so that we can see the whole tiled world at a time.
4. *Tile View:* The camera that is attached to any tile, which is either static or moving. One should be able to select the tile view and move it to tiles nearby.
5. *Helicopter Cam:* Here the camera is movable with the mouse in an intuitive manner. Click and drag should change the look angle, up vector is always up, and scroll wheel will move the camera close or away from the scene.

***Optional*:** Apart from the views mentioned here, you are encouraged to come up with innovative camera views.

2.3 The Level

The 3D world (let’s call it the ”level”) can be thought of as being made of many block (cube) like structures. The character is to be navigated along this level avoiding/overcoming obstacles that he may encounter. Here are some of the types of world ”blocks” that we would like you to use somewhere in the levels:

- Static: These blocks do not move.
- Periodically Moving: These blocks move periodically up/down/right/left or a combination of all of these
- Falling: These blocks start falling down and get destroyed (say maybe within 2 seconds) when the player jumps on it.
- Trampoline/spring: You jump on one, and you bounce back up. If you jump on such a block from a higher point, you bounce more.
- Teleporter: As the name suggests, it teleports you to some particular block on the level.

***Optional*:** Apart from these, you can use your creativity and come up with many more things from your side. Make the level by using combinations of the above mentioned blocks, and any other stuff you come up with. It’s upto your brave warrior of a character to get across these to the finish.

2.4 Physics

Your game must obey real-world physics. You must simulate gravity and the character motion should apear realistic when he jumps. (eg: jump in a particular direction and you should have a nice parabolic path). If you jump from too high a point, you die (or lose health if there is a concept of health in your game). Collisions must be implemented. You can’t just walk through or fall through solid blocks.

3 Controls

The keyboard controls moving the person by a fraction of distance of a tile size (left,right arrow keys). Keeping the key pressed will increase the speed of motion and releasing the key will result in stopping the motion, but after some motion due to inertia. The j key will cause the character to jump, while moving in the direction of motion. In addition, one should also be able to control the position of the camera to any of the above positions.

4 Optional

Feel free to include additional objects, animations, textures, etc. to make the world more realistic and rich. You may also provide moving obstacles or other traps the make the journey from start to end more treacherous. Additional interesting camera views may be provided.

5 Coding

It is essential that you do a clean design of your system before starting to code. This is especially critical as you would later be required to extend the work in this assignment. The TAs will be uploading a framework for coding. You are urged to use the framework, but more importantly, understand and appreciate the reason of the design.

6 What you should make

1. **Assignment 2 Part 1 (due 28th Jan)** Your Humanoid character should be completely ready along with basic hand and leg animations when he moves. Simple gravitational physics must apply on him. No level design is expected (ie. create the character, add physics and make him simply walk, run, jump on a flat ground). You must implement the first two camera views: first person and third person views. *This submission will account for 30% of the grades.*

2. **Assignment 2 Part 2 (due 8th Feb)** All the camera modes must be implemented. Only one level is expected. The first two kind of ”blocks” must be implemented (static and periodic) while building the level. The character must obey full physics and collisions with the environment. *This submission will account for the remaining 70% of the grades.*

The remaining parts will be included in assignment 4.

7 Submission

You submissions should include your source code, a makefile and a compiled executable. You need to include a readme file that describes any additional information that is needed in compiling/executing you code. Do not use any non-standard libraries (other than OpenGL, Glut and any that is compiled into your code). You should explicitly state it in your readme and report if you make any use of code from the web. Without this, any such use will be treated as plagiarism, and will be dealt with in severe terms.

The submission will be in two stages: The first, due on the 28th should contain at least the basic animated character with its motion controls and the first two camera positions working and controllable from the keyboard. The world should at least be a simple set of tiles with some missing. You are welcome and advised to do more by this deadline.

The final submission, on Feb 8th, should contain the remaining camera views, tile motion, gravity and collision detection, and the complete world with texture and any optional items you may choose to add.

8 Grading

You will be graded based on the correctness and efficiency (speed) of the implementation of the minimum elements described above. This will contribute to 90% of your grade. Remaining 10% will be given based on the improvements that you do over the basic world. In addition, submissions that are found to be exceptional by the graders, will be showcased, and will be awarded extra credits up to 10%.

30% of the grades will be based on the submission on 30th and the remaining 70%, based on the final submission.