RMS

Brett Terpstra

Mr.Kedwell

ICS3U1

Due: 2019-06-16

Location: Inside the Legacy-3D-Visual-Basic folder

**PLEASE READ THE INTRODUCTION BEFORE USE**

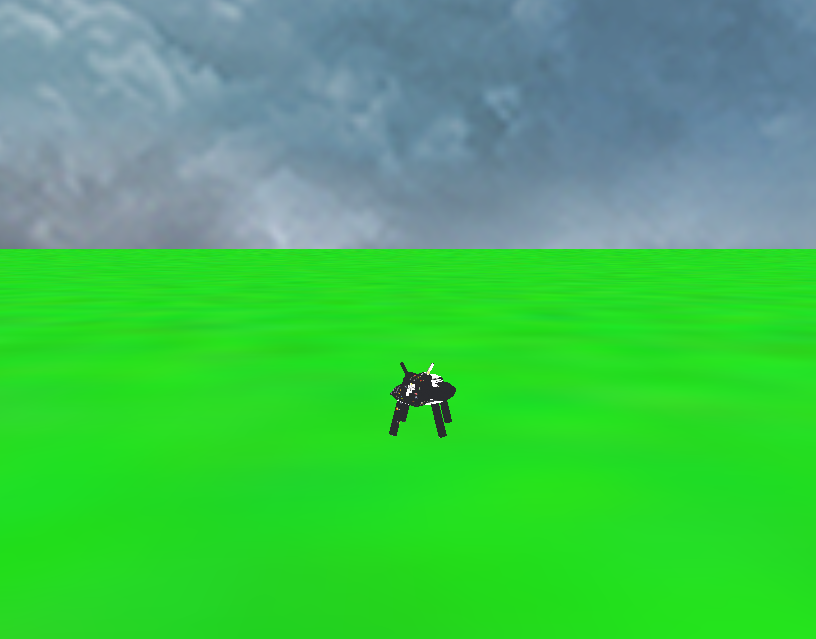


Table of contents

**Introduction and information about the game**

1..6

**Step 1 (Defining the Problem)**

7..7

**Step 2 (Analysis and Design)**

8..10

**Sites used as reference**

11..11

**Final work log**

12..13

**Introduction**

The application, RMS, which named after Richard Matthew Stallman, who is the creator of the free software foundation, allows for a user to define entities and traits inside a world file and then watch them interact inside a virtual environment where they hop around, breed and die. It is recommended that the user has at least 1gb of ram and a dual core computer. Windows 7 will result in the **best** performance, as there is an issue with windows 10 (see bugs section). Im writing this both for you kedwell and for the github readme file. DO NOT WORRY IF IT TAKES A MINUTE TO LOAD. There are a lot of assets in the game that need to be loaded from file. (Sorry if im missing information in this readme. If information is missing please make an issue report on the github ( <https://github.com/Tri11Paragon/Legacy-3D-Visual-Basic/issues/new>), I will update the github will bugfixes and docfixes)

**Bugs**

There are a couple of bugs within the game that are known and are out of my control.

1. There is an issue with stuttering in windows 10. This must have something to the way windows 10 handles its windows, and the way OpenTk handles windows 10. Please use a computer with windows 7. I don’t know if regular windows 7 works, but I know the school computers windows 7 works very well. The only solution to the windows 10 bug that I have found is to alt tab a couple of times and it can fix the issue.

2. On windows 10 the camera look movement is inverted and I have no idea why. This again seems to be a problem with OpenTk not having a mouse lock. The solution is to use the key command Alt + Shift + F (see below) to invert the camera controls or by editing the settings.dat file found in the data folder, to contain a “flipRotate:-1”.

Please report any found bugs to the github issues page or by emailing me at [brett.terpstra@lkdsb.com](mailto:brett.terpstra@lkdsb.com)

( <https://github.com/Tri11Paragon/Legacy-3D-Visual-Basic/issues/new> ). I am not responsible for ANY crashed caused by user error or by not following the proper format instructions inside this file. Please do not submit a bug report if the crash is caused by user error.

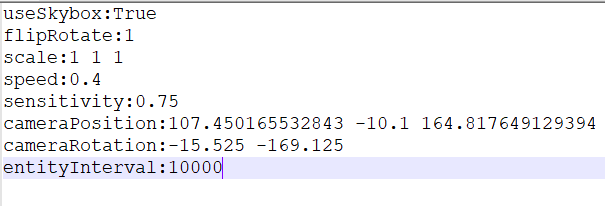
**Loading Custom Assets**

All the games assets are loaded at startup allowing for the user to load custom models, textures, and music. The music is loaded from the music folder and is played randomly at all times. There is always a song that that is playing. Songs must be in .wav format. Textures are loaded from the textures folder, and are assigned a number at startup. They are loaded in alphabetical order but if you want to be sure then you can check the console at startup. The game supports up to 498 textures in .png or .jpg format, other formats should work fine, but these are the only 2 types tested. When creating a entity, this number MUST be used as it servers as a pointer to the texture. Unless a texture is added that changes the order of files, textures will always point to the same texture. 2 textures are required by default. The grass.png texture found in the textures folder, and cube.png found in the skybox folder.(folder inside textures) These come with the game. Loading models into the game is a lot easier. To load models into the game, place any .obj file into the primitives folder. Thats it! Now you can when defining an entity reference the .obj file by its name. (note the name must include .obj at the end.)

**Settings / World Control**

**Settings**

There are many settings that can be control inside this application. The settings file is pre-generated by the application at runtime and will be filled out when the game is closed. All user-controllable settings are inside this file. This is an example of the settings file. (The settings.dat and world.dat are case-sensitive)

Everything within this file can be changed by the user.

**useSkybox** explains itself, true and the skybox is enabled, false and the skybox is not.

**fliprotate** inverts the camera rotation. (See bugs section)

**scale** is the scale of the universe (change if you want everything to get bigger) (this includes the terrain)

**speed** is how fast the camera moves.

**sensitivity** is how sensitive the game is to mouse movements.

**cameraPostion** is the camera’s position.

**cameraRotation** is the camera’s rotation.

**entityInterval**: how how long(in ms) it takes the entity to do its checks for birth, death, and movement. Default is 10 seconds.

**Controls**

The game has many user controls that allow you to change settings at runtime without editing the file. (settings cannot be reloaded at runtime). All changes made will be saved when the game closes

**Alt + Shift + Plus** increases the speed movement of the camera by 0.1

**Alt + Shift + Minus** decreases the speed movement of the camera by 0.1

**Alt + S** saves the entities to the world file (useful when wanting to edit the file at runtime)

**Alt + Shift + C** reloads the world file (without closing the application)

**Alt + Shift + F** flips the mouse movement

**F12** resets the camera position to 0, 0, 0

**F2** takes a screenshot and saves to the screenshots folder.

**W** moves the camera forward

**S** moves the camera backwards

**A** moves the camera left

**D** moves the camera right

**Left Shift** moves the camera down

**Space** moves the camera up

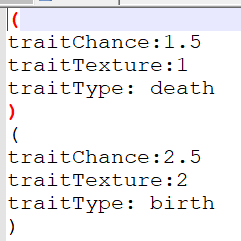
**Escape** frees the mouse movement

**World**

The world is loaded from the world.dat file. Entities and traits have a predefined format that needs to be followed. If this format is not followed, it is not my fault if the game: crashes, has unexpected issues or is buggy. Any word starting with $ means that the user is expected to enter a value into here

**Traits**

Traits are defined usually at the start of the world. They follow a very simple format. Here is an example of the format. All of **are required**:

 The start of a trait is always “(“ this defines that you wish to create a new trait.

The next can be any of the 3: (these can be in any order)  
 **traitChance:$Double** – This how much the trait effects the specific trait type, if its 1.5 and the trait type is death, and the trait texture is 1 then any entity with the texture id of 1 will have a 1.5% increase chance of death

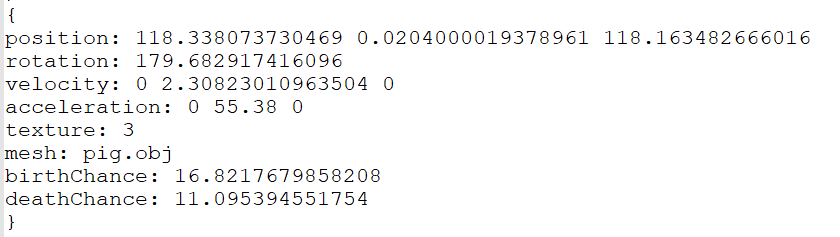
**traitTexture**:**$Integer** – this is the texture that will be effected by the traitChance

**traitType:$type** – This is the type that will be effected. The 2 options are birth and death. (has to be these to and is case-sensitive)

The end of a trait **must** be “)” which tells the application that you are done defining this trait. This is required before you start a next trait.

**Entities**

Entities are defined usually after the traits. They also follow a very simple format. Here is an example of this format:

The only thing required here is the mesh, the application won’t crash if you miss any of these as they are predefined inside the code, but it is **very recommended** that you have all of these defined inside the entity.

Entity format starts with a “{“ This is required. The following can be in any order:

**position: $x $y $z** – this is the entity x,y,z position. The position is loaded as a double.

**rotation: $r** – this is the entity rotation. It is also a double. (rotation is currently disabled)

**velocity: $x $y $z** – This is the current x,y,z velocity of the entity. These are in double.

**acceleration: $x $y $z** – this is the current x,y,z acceleration of the entity. These are also in double.

**texture: $int** – this is the integer pointer to the texture. This can be found in the console of the game output at startup.

**mesh: $name** – this is the name of the mesh that the entity will use. This needs to be its full name which includes the filename and extension. Eg pig.obj.

**birthChance: $double** – This is the entity’s default birth chance, this will be passed onto its children with an added randomness. It is also a double.

**beathChance: $double** – this is the entity’s default death chance, this will be passed onto the children with added randomness. It is a double.

The final line of an entity definition must be “}”. This is required to end the definition of an entity. If you do not follow this, the game WILL get very made at you.

ICS3U – FINAL PROJECT  
STEP 1 – DEFINING THE PROBLEM

***Type of Application*** (i.e. game, text editor, trivia, music app, other)

Simulation Game

# ***Detailed Application Description*** (ie. storyline, how to play, scoring, similarities to other programs)

My idea for this “game” is not really a game nor a education simulator, more of a curiosity settler. Ever wondered how 2 populations of animals with different advantages and disadvantages would interact? Well that’s my idea for a ‘game.’ You spawn in entities that interact, die, reproduce, etc. Similar to a game called equilinox but with a lot less features and more basic.



***Programming Knowledge Required*** (i.e. variables, loops, procedures, selection etc…)

Almost all programing + experience in 3d graphics using openGL

Varaibles – storage for everything

Loops – generating terrain/entities

Procedures/functions – everything( rendering, creating, etc)

Classes / Structures – compartmentalizes data and creates objects(yes I know it can be bad, but for games its good)

Loading/file reading – loading obj files to be rendered

\*\*Submit this completed form to the drop box on eLearning “Step 1 – Problem Definition” \*\*

ICS3U – FINAL PROJECT  
STEP 2 – ANALYSIS AND DESIGN

**APPLICATION SUB PROGRAMS**

1. List the various subprograms that you will create in your project. (Event Procedures, General sub procedures, and Functions)

Do you want all of them?  
I have a lot

Sub loadPolys  used to load all default models into the game

Function createData  used to generate a readable touple

Function loadFromFile  loads the obj

Function fn\_1293  returns whether or not to read this line of text.

Function distance  returns distance between 2 points.

Sub drawPoly  draws poly from touple

Sub keyReleased  handles keynoard input

Sub update  updates camera stuff

Sub OnRenderFrame  called once per update frame

Sub OnResize  called when game is resized. Makes sure projection matrix is good

Sub OnLoad  called when game window is opened

Sub LoadTextures  called to load textures into pointers

Sub LoadTexture  loads textures from bitmaps and stuff

These are the main functions/subs im using

**DEVELOPMENT PLAN**

Create a basic daily calendar outline and the functions and procedures will be created and a time-line for completion. List when you would like particular portions of your application complete.

*SAMPLE*

*June 3 – Create Form 1 interface and define enemy and player structure. Create setup() procedure to fill enemy structure*

*June 5 – Play event procedure()*

I need to:

-finish loading objs

-make sure objs render correctly

-add a terrain

-add collision with that terrain

-fix entities

-improve entity velocity and other properties

-decrease amount of useless code

-make a console + command interpreter.

-add GUI

A few of these should be done before the others but it does not matter what I do and in what order.

**PROBLEM SOLVING TOOLS**

1. Complete a minimum of 2 of the following 3 problem solving tools.
   1. Create an IPO chart for one of your function or procedures (general or event)
   2. Create a Flowchart for one of your function or procedures (general or event).
   3. Outline an Algorithm for one of your function or procedures (general or event).

C: (Algorithm for loading an OBJ file)

1. Load obj file into memory
2. Read a line of the file
3. Split the string by the spaces into an array
4. Set a type variable to the first element in the array
5. Remove the first element from the array
6. Use a switch with the type variable
7. If the type is v then add the first 3 elements in the array to a list of vector3s as a single vector
8. If the type is vt then add the first 2 elements in the array to a list of vector2s as a single vector
9. If the type is f then loop through the array
10. if the length of the string of the currently selected element is 0 then loop without completing the next step
11. split the currently selected element at “/” then add the first element of this new array to a list of ints called “vertexIndices” then add the second element of this new array to a list of ints called “textureIndices”.
12. Repeat at step 2.

A(IPO chart for rendering a polygon)

|  |  |  |
| --- | --- | --- |
| INPUT | PROCESSING | OUTPUT |
| Touple of vector2 and vector3 | Gl.begin(gl.quads)  For each p in touple  Gl.texCoord2(p.Y().X, p.Y().Y)  Gl.Vertex3(p.X().X, p.X().Y, p.X().Z)  Next  Gl.end() | Draws a polygon on the screen based on the input |
| t:0.0, 0.0 | v:-1, 1, 1  t:1.0, 0.0 | v:1, 1, 1  t:1.0, 1.0 | v:1, -1, 1  t:0.0, 1.0 | v:-1, -1, 1  t:0.0, 0.0 | v:1, 1, -1  t:1.0, 0.0 | v:-1, 1, -1  t:1.0, 1.0 | v:-1, -1, -1  t:0.0, 1.0 | v:1, -1, -1  t:0.0, 0.0 | v:1, 1, 1  t:1.0, 0.0 | v:1, 1, -1  t:1.0, 1.0 | v:1, -1, -1  t:0.0, 1.0 | v:1, -1, 1  t:0.0, 0.0 | v:-1, 1, -1  t:0.0, 1.0 | v:-1, -1, -1  t:1.0, 1.0 | v:-1, -1, 1  t:1.0, 0.0 | v:-1, 1, 1  t:0.0, 0.0 | v:-1, 1, -1  t:1.0, 0.0 | v:1, 1, -1  t:1.0, 1.0 | v:1, 1, 1  t:0.0, 1.0 | v:-1, 1, 1  t:0.0, 0.0 | v:-1, -1, -1  t:1.0, 0.0 | v:-1, -1, 1  t:1.0, 1.0 | v:1, -1, 1  t:0.0, 1.0 | v:1, -1, -1 | Gl.begin(PrimitiveType.Quad)  For each p in touple  Gl.texCoord2(0.0,0.0)  Gl.Vertex3(-1, 1, 1)  Next  Gl.end() |  |

**Sites Used as Reference**

**Copy and paste the URL for all websites outside of the course website that you used in the creation of your final project. If you submit work that you do NOT give credit to the creator, this can result in loss of marks or a zero on your final project.**

1. <https://opentk.net/api/index.html> (used to find out how the API bindings work)
2. <https://github.com/Tri11Paragon/3DJavaGame> (I stole some code from myself)
3. <http://www.vbforums.com/showthread.php?752731-RESOLVED-Create-Bitmap-from-raw-image-data> (writing data to bitmap)
4. <https://github.com/dabbertorres/ObjRenderer>
5. [https://github.com/Tri11Paragon/3DJavaGame/tree/master/3d%20game/3D%20Game/src/objConverter](https://github.com/Tri11Paragon/3DJavaGame/tree/master/3d game/3D Game/src/objConverter)
6. <http://ogldev.atspace.co.uk/www/tutorial25/tutorial25.html> (I’ve done this in modern GL in java only)
7. [https://github.com/Tri11Paragon/3DJavaGame/tree/master/3d%20game/3D%20Game/src/skybox](https://github.com/Tri11Paragon/3DJavaGame/tree/master/3d game/3D Game/src/skybox) ( This is my skybox renderer in modern GL) (Used it as a basic reference)

ICS3U - Culminating Activity – Work Log

Work log should be completed **daily** outlining your accomplishments for the day and what you plan on working on the next day. This log will be checked randomly throughout the process to verify it is being completed along the way.

|  |  |
| --- | --- |
| **Monday – May 27, 2019** | Completed: Worked on implementing OBJ loader into vb  Tomorrow: continue work on OBJ loader |
| **Tuesday – May 28, 2019** | Completed: Worked on a Legacy renderer using touples. It does not like to work due to loading methods of the OBJ. Need to revise OBJ loader or use a different rendering method.  Tomorrow: Work on OBJ loader/renderer |
| **Wednesday – May 29, 2019** | Completed: got distracted and ended up adding screenshotting  Tomorrow: work on OBJ loader/renderer |
| **Thursday – May 30, 2019** | Completed: Hopefully finished the OBJ loader and renderer.  Tomorrow: test the OBJ loader |
| **Friday – May 31, 2019** | Completed: Ok, so the OBJ loader does not want to work the legacy way and the new legacy way I have no idea how to load textures into, but the model loading does work and I think im just going to have them without textures and just set the model color to be whatever the entity color is.  Tomorrow: See if I can get textures working, if not, then maybe terrain.  **STEP 1 DUE**  **Work Log 1 Due** |
| **Saturday – June 1, 2019** |  |
|  |  |
| **Monday – June 3, 2019** | Completed:was having big issues with the obj loader, but I got it working with textures!. Turns out the crash when calling gl.drawarrays was because I was including texture verticies in the count, only needs to be vertices  Tomorrow: Terriain?! |
| **Tuesday – June 4, 2019** | Completed: Tried adding skyboxes, but for some reason the openGl did not wish to work. I have no idea why it was not working, as I was using code that is being used to render entities. Entities are clearly working, so I have no idea why skybox is not.  Tomorrow: |
| **Wednesday – June 5, 2019** | Completed: Skyboxes are now working. Added GUI renderer, but there is a bug where it does not render on screen and instead renders in the world. Need to fix.  Tomorrow:  **STEP 2 DUE** |
| **Thursday – June 6, 2019** | Completed:worked on GUI rendering. Does not really work but it renders.  Tomorrow: screw GUi and work on entities. |
| **Friday – June 7, 2019** | Completed:  Tomorrow:  **Work Log 2 Due** |
| **Saturday – June 8, 2019** |  |
| **Sunday – June 9, 2019** |  |
| **Monday – June 10, 2019** | Completed: Added sounds / sound player. Made some changes to how camera stuff works and tried to make the game load a little quicker. Added terrain and terrain collision ( with camera ). Added a tree model.  Tomorrow: Work on entities and make trees spawn around the world |
| **Tuesday – June 11, 2019** | Completed: Added world saving and entity loading  Tomorrow: add traits to entities |
| **Wednesday – June 12, 2019** | Completed: added clock class and added acceleration to entities  Tomorrow: add more traits |
| **Thursday – June 13, 2019** | Completed: Added gravity to the entities  Tomorrow: more traits |
| **Friday – June 14, 2019** | Completed: added trait loading  Tomorrow: add birth/stuff |
| **Saturday – January 15, 2018** | Added birthing / death but it has weird bug were the entities rotate with the camera. Have no idea what I causing this. |
| **Sunday – January 16, 2018** | **Figured out that the issue with camera rotation is caused by having a 0 death chance. I have no idea why this causes this but I have fixed it by not allowing 0 death/birth chance.**  **- PROJECT DUE BY END OF DAY**  **- FINAL WORK LOG DUE** |