

Pokémon Snap: The Sequel v1.0
Spencer King, Hannah Porter, David Wu
2/03/2014 - 4/30/2014
CSE20212 Final Project, University of Notre Dame

Repository:

<https://github.com/spencerking/CSE20212-Final-Project>

Overview:

Our project was originally going to be a 3D reinterpretation of a level from a classic 2D Legend of Zelda game. None of us had any prior experience with OpenGL and so we were not as efficient as we could have been as far as tutorials are concerned. This led to us not realizing our project was not feasible until the end of March, leaving us with slightly over a month to plan and complete a new project. We ask that you please keep this in mind when playing our game.

Our project is based on the Nintendo 64 game Pokémon Snap. The objective of the game is simple; you are to explore the terrain and photograph Pokémon of your choosing. Other than being unable to pass through the terrain, there are no limits to your movement. You can freely explore the map and set up photos the way you want. This game cannot be won in the traditional sense; the goal is simply to have fun.

Controls:

W- Move the camera forwards
A- Move the camera to the left
S- Move the camera backwards
D- Move the camera to the right
Up arrow- Move the camera upwards
Down arrow- Move the camera downwards
E- Take a photograph (stored in the game's directory)
M- Pause the background music
Q- Quit the game

Necessary Libraries:

OpenGL

-This library is used for handling all of the graphics.

SOIL <http://www.lonesock.net/soil.html>

libjpeg <http://libjpeg.sourceforge.net/>

-The SOIL and libjpeg libraries are used for image loading and texturing.

FMOD EX <http://www.fmod.org/>

-This library is used for playing the background music and sound effects.

Compiling and Running:

In order to compile the game, you simply need to type 'make' into the terminal. After compiling successfully, the game can be played by typing './main'.

Systems Tested:

The program compiles and runs on both Mac OS and the student machines; however, the game is optimized for Mac OS. We have not fully tested the game on Red Hat Enterprise 6 on the student machines, so it may not behave as expected.

Known Bugs (Features):

Some of the Pokémon textures do not load properly.

- This could be fixed by editing the texture files, but we did not have time for some of the more complex models.

Sometimes lag is experienced after the program has been running for a bit.

The lag is sometimes worse because of the models moving, but not always.

- This is fixed by commenting out the movement function. The movement function is called in both the keyboard and arrow key functions, and is called every time a movement key is pressed

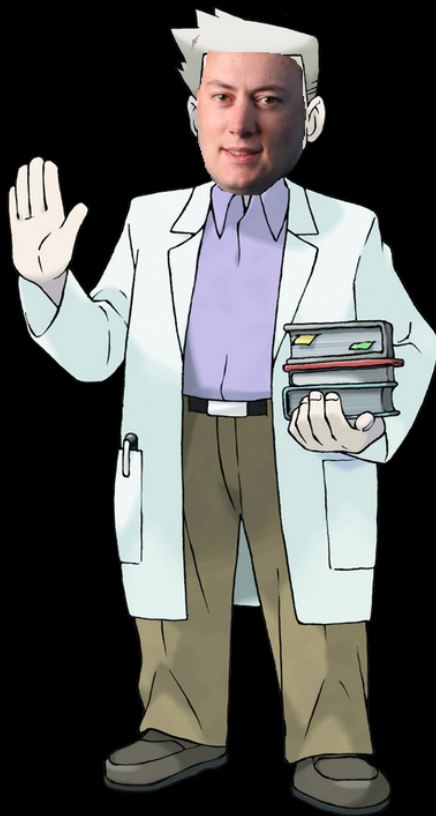
The function for moving the models is unpredictable. It sometimes, though rarely, causes segfaults. When it does segfault it has always been early on in the program's execution. We can run the program repeatedly and not have a segfault, but then try again later and experience one. The only consistency is that it is always right at the beginning.

- We are not sure what causes this, but believe it likely has something to do with how memory is managed for the Pokémon models since we allocate memory for each model.
- Due to our rather late change in project, we did not have time to fully debug this, though the program has always run fine in our tests when the model movement function is commented out.

The Pokémon can move through each other and while they cannot pass through walls, they can get stuck partially inside walls.

- This could be fixed by further optimization of our model movement function.

The camera will usually flip upside down at the very beginning, but the normal orientation can be easily recovered.



**HELLO, AND WELCOME TO THE WONDERFUL
WORLD OF POKEMON.**

MY NAME IS EMRICH.

**PEOPLE AFFECTIONATELY REFER TO ME AS
THE DREAD CODER.**

**YOUR GOAL IS TO TAKE PHOTOS OF POKEMON
TO HELP ME IN MY RESEARCH.**

DON'T FORGET TO HAVE FUN!