PS0

HELLO WORLD WITH SFML

The main purpose of this assignment is for you to get up and running with your build environment. You'll write a bit of code. Also you'll join the class Google discussion group.

1. GET YOUR BUILD ENVIRONMENT SET UP

You must use **gcc / g++** and **Make** to configure and build your code.

For more information about Virtualbox see
https://www.virtualbox.org/manual/ch01.html#hostossupport

There are various options:

2. You can use the CS department's Linux machines in the labs.
3. You can install Linux and dual-boot your machine. We recommend the Ubuntu LTS version (currently 18.04) Ubuntu Desktop. Often, it is straightforward to set up for dual-boot—but there is potential for things to go awry and render your machine non-bootable. So, we recommend this only if you are confident about being able to recover in case of problems, and that you definitely have your Windows machine backed up. (You have to have Windows on the machine first; then install Ubuntu.)

We are using the C++ Simple Fast Media Library, SFML, www.sfml-dev.org. Any version 2.0 and up will work.

You may use any IDE that you like, or none at all.

2. BUILD THE SFML HELLO WORLD CODE

The SFML "did I install everything correctly" code is about 20 lines long and results in your computer popping up a window like this:

Go to the tutorial for your platform, e.g. https://www.sfml-dev.org/tutorials/2.5/start-linux.php, and get the code running on your machine.

3. EXTEND THE DEMO CODE.

Now that you've got things running, do something fun with SFML. In addition to the green circle, look at the documentation and get your window to show:

1. drawing an image sprite

For #2 and #3 above, you can make x, y variables that represent the position of the sprite, and update them when a keystroke is detected. You're welcome to do something more complicated if you like. The `window.setFramerateLimit()` method might be helpful if things move too fast (see "Controlling the framerate" at https://www.sfml-dev.org/tutorials/2.5/window-window.php).

Notes:

- It's correct to clear everything in the window and redraw stuff each time through the event loop. That's how SFML works.
- You reposition things (e.g. a sprite) by setting its position before you draw it. Check the sprite API docs.
- Use only relative paths to load your sprite image—e.g. `"./sprite.png"`, *not* `"/home/fredm/ps0/sprite.png"`.
- If you draw stuff outside the bounds of the window, you won't see it :)

Compiling program:

- When you compile your code, add the flags "`-Wall -Werror -ansi -pedantic`". We will be using these to test your code, and your code must compile when these flags are used.
- To use C++11 features use flag `-std=c++0x`

Make sure your code file is named `main.cpp`. You will submit this. Make sure your name is at the top of the file.

Make sure your sprite image file is named `sprite.png`. You will submit this.

When you are done, take a screenshot of just your SFML window, and save this

show your SFML window.

4. ADDITIONAL COURSE PARTICIPATION

Please do the following:

- Join the course discussion group at https://groups.google.com/forum/#!forum/computing4summer2018.

5. ASSIGNMENT README FILE

Before submitting, fill out info in the `ps0-readme.txt` file. Make sure you re-save it as a text only file. It must be named `ps0-readme.txt`.

6. SUBMIT!

You will be submitting at least four files.

1. Your SFML demo program, named `main.cpp`.
2. Your sprite image file, named `sprite.png`. (If you made more than one sprite, you can name the others whatever you like.)
3. Your screen grab showing your code running, named `screenshot.png`.
4. Your completed `ps0-readme` text file.

The four files must be in a subdirectory named `ps0`. Then, `cd` to the directory containing the three files, and type the following:

```
$ ls
```

You should see the following displayed:

```
ps0/
ps0/sprite.png
ps0/screenshot.png
ps0/ps0-readme.txt
ps0/main.cpp
```

Then, submit the file `my-submit.tar.gz` **using the submit utility in the CS department linux.** (Note: you don't have to call it “my-submit.tar.gz”. It would be better to name it with your real name and PS0, e.g. *ps0.tar.gz*. But it must be a gzipped tar of the correct directory.)

Submit using the submit utility in the CS department linux, as follows:

submit schakrab ps0 ps0

GRADING RUBRIC

file name correct 1,

SFML window being displayed 1,

image 1,

image being in moving sprite 1,

sprite changing direction or somehow responding to keystrokes 1,

Total 12