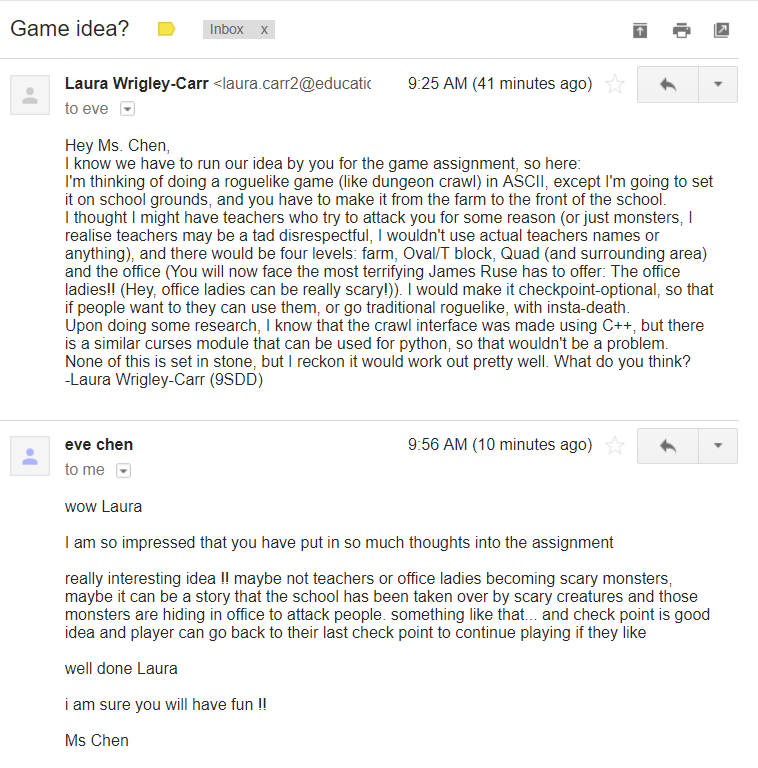
23/11/17

We did our SDD exam. Despite this being an invariably sad occasion, due to the scarring experience of completing the exam, I can now begin to once again focus on the game that I’ll get to make. I already have a few ideas!

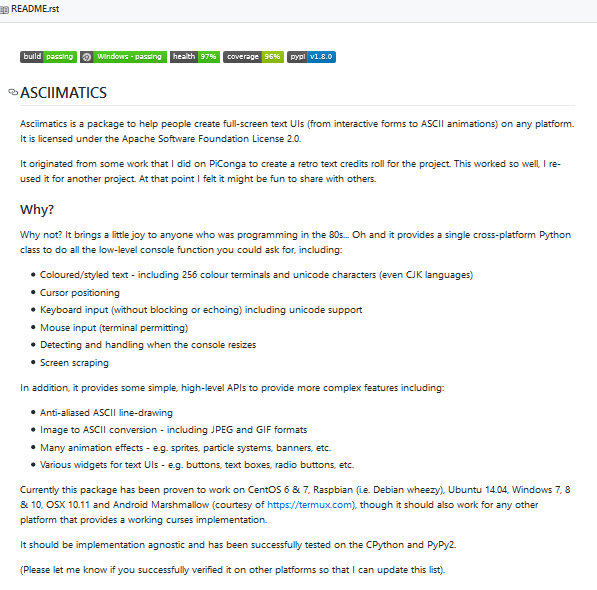
24/11/17

Today we got the assignment in SDD. I’ve already been thinking about what I’m going to do for a while now, so now that we’ve got the go-ahead, I can pitch my idea to Ms. Chen. I’m thinking of maybe doing a roguelike game with an ASCII interface (like crawl), except I’m going to set it at school, and the player needs to make it from the farm to the front of school. Hopefully, Ms. Chen will approve my idea, so I can get started ASAP!

25/11/17

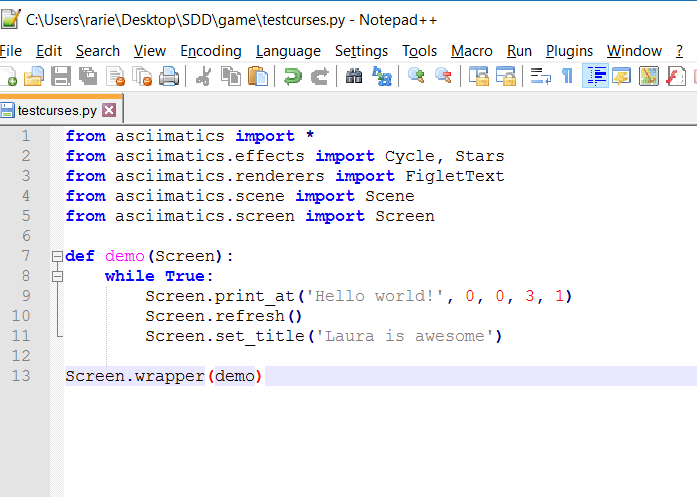
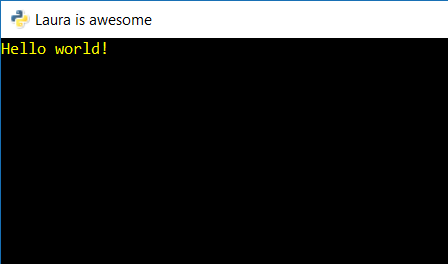
Today I emailed my idea to Ms Chen, to see what she thought. I got my response surprisingly quick.

Now that I’ve got approval, I can get started. I’m hoping to get a lot done in the next little while, as once I get back to school next year, there’ll be tons of other work I’ll have to do, and I don’t want to have to finish this in a rush.

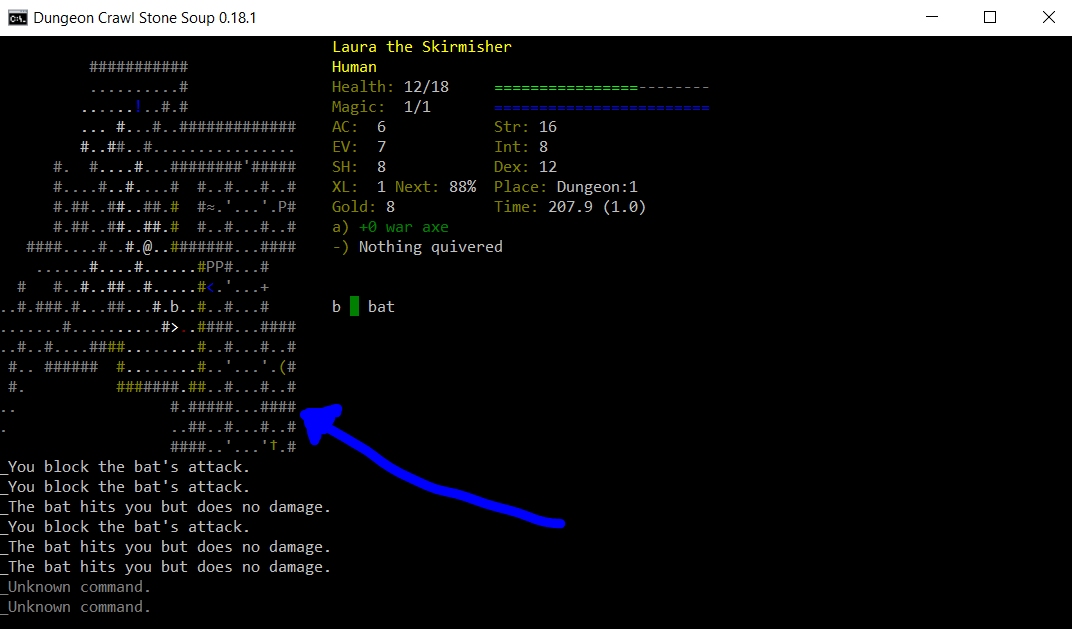
26/11/17

Guess what? I’ve already run into a slight problem (actually, it’s a pretty big problem). I discovered that python curses aren’t available on the windows version of python. My first thought was ‘Use another operating system!’ But since I need to have it working on a windows computer when I hand it in, it’s probably not the best idea to develop it on a different operating system. I decided to call in the cavalry, after unsuccessfully finding a solution on the internet. (There are a few tools that are meant to be like python curses but for windows, but the majority seem to be pretty ancient (like windows xp ancient)) By ‘the cavalry’ I of course mean my dad, Gavin Carr. He’s a geek himself so he knows a good bit about computers. Of course, 5 minutes after outlining my problem, he does a quick google search and manages to find a solution (why?????). I’ve decided to use a module called asciimatics. It’s pretty much exactly like python curses (even down to the same codes for the colours!), except it runs on windows, and it looks to be regularly maintained. Thanks, Dad!

28/11/17

Today in SDD, we were given time to look at past projects, to give us inspiration for what we will do. Of course, since I already planned what I’m going to do, I basically just played around with asciimatics, and am starting to get the hang of it. I’ve made a little program to try different things out. At the moment it doesn’t do much of anything, but I’ll build on it, and eventually have it doing backflips (figuratively).

30/11/17

Today we got another period to check out the game folders, so I continued my work with figuring out asciimatics. Since Monday, I’ve figured out how the taller text works (FigletText), and so can now make the titles seem more dramatic (see first picture). Next I plan to figure out how to make a layout interface, particularly the map, like in the second picture.

I’m not sure how far into that I’ll be able to get before having to figure out how to use input commands, but it’s impossible to start with user commands, so there’s not exactly a way to do the user commands without the interface (unless I plan to randomly rearrange titles in response to input, though why you’d want to know how to do that is beyond me.

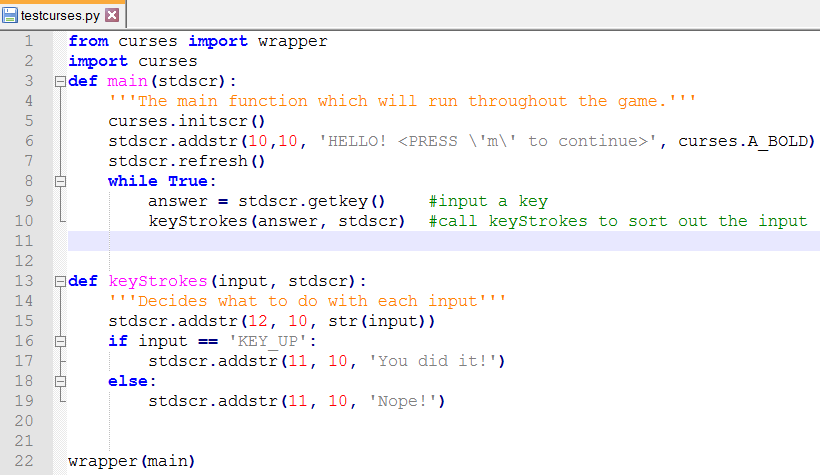
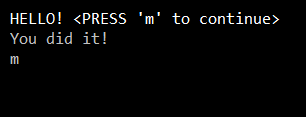
01/12/17

We got some more time to brainstorm our project today. For me, that meant trying to figure out how to make the map interface. Thus, my first stop was the asciimatics documentation. However, to my surprise and horror, there was no part of the documentation referencing this anywhere! Eventually, after much more research I concluded that maybe asciimatics wasn’t such a perfect tool after all for this. I was therefore back to square 1.

I proceeded to do some research on the other possible tools that I could use – these included Unicurses, and another unofficial port of curses for windows. Both would end up doing the same thing, but I ended up discovering that Unicurses has no support for x64, which was my machine (seriously???). By this point I was extremely annoyed (it was also late afternoon at this point, after spending the entire day chasing up dead ends). That isn’t particularly relevant, but in order to properly paint the picture of what occurred, you really need to appreciate my state of mind at the time. Anyway, this left the other port as my only option.

I ended up going through this whole rigmarole of having to make pip work on the command line instead of Cygwin (and I hate Command prompt with a passion!! Who even uses batch anymore?) but eventually made it work, and the module, conveniently named ‘curses’ was installed. I tested it, and everything was fine. In conclusion, it has been a very annoying and frustrating day, which has involved coding in ¾ of my periods in which I had class, and way too much hatred of the windows operating system in general. (I do not include any screenshots in this particular log, as there isn’t too much to screenshot as of yet.)

05/12/17

Yeah, so I kinda took the weekend off from programming, as I was still rather annoyed about it (not to mention I had ISCF day conference and had to put up the Christmas tree…). Anyway, I got back into it today during DT, when we had some free time. I began to once again figure out how everything worked, even playing around with a little user input stuff. (see below)

In the image above, the first line was the original message, the second line popped up after the user pressed ‘m’. The third line printed what the user typed.

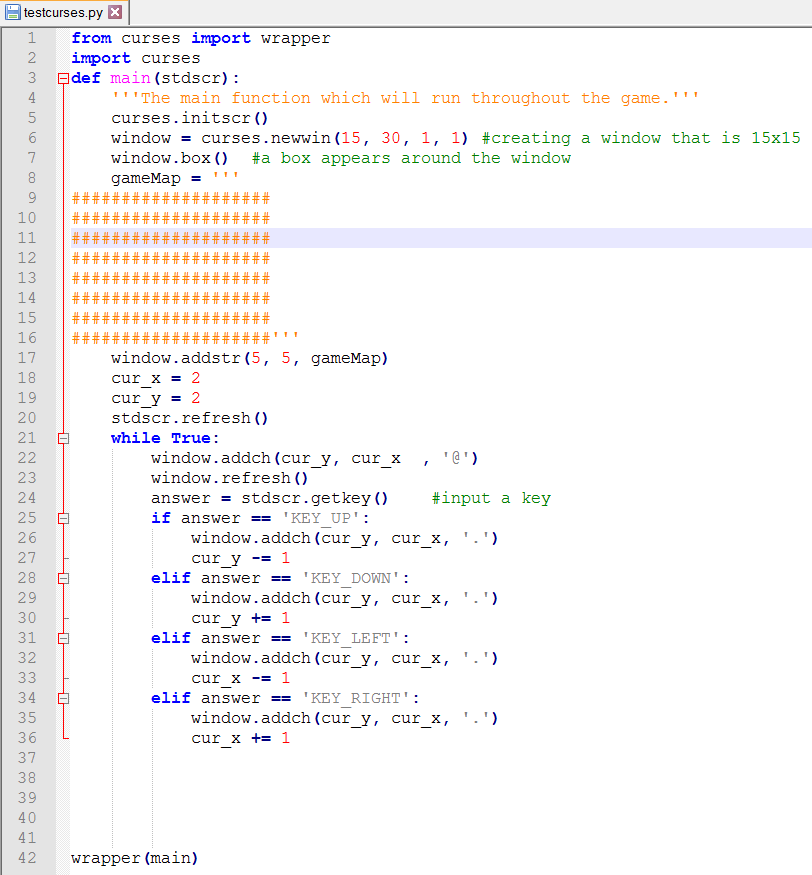
Anyway, now we have double SDD, and so I intend to make the most of the time by doing some research about the map. I reckon that making the interactive map, with all the items and the player moving through it, will probably be the most difficult part of the game, so I want to get started on it as soon as possible. The obvious place to start would be Dungeon Crawl, the original inspiration for my game. Although it was made in C++, and mine will be in python, they both use the curses module, meaning that the general outline of the code should be the same, barring syntax changes. Thus, it’s the perfect place to go to figure out how the map works.

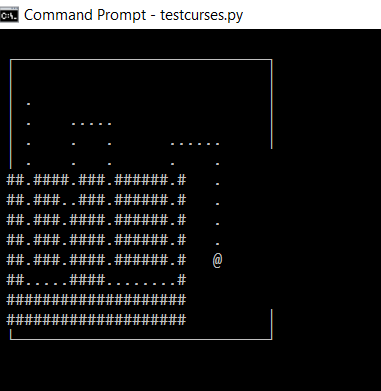
[EDIT] Lol, never mind, we got our assignment back instead, so I didn’t get a chance to work on this. Hopefully the next SDD lesson.

08/12/17

Over yesterday and this morning, I’ve been trying to figure out how to make this whole map conundrum thing work. I did some research, and fiddled with things, basically dumping half my code and re doing it, but I’ve made some big progress. I first made a window, and drew a box around it so I could see it. I then made stuck @ inside, representing a character. I made little functions so that they could move around with the arrow keys, and just for fun, made them leave dots wherever they went.

From there, I knew I’d need to figure out how on earth to do the whole map conundrum, especially since you apparently can’t use addstr() to print multiple lines. Or can you? I discovered that if you define the string beforehand (e.g. gamemap = ‘’’…..’’’), you can put multiline strings in a single addstr(). This of course made it significantly easier. Here’s what I have so far:

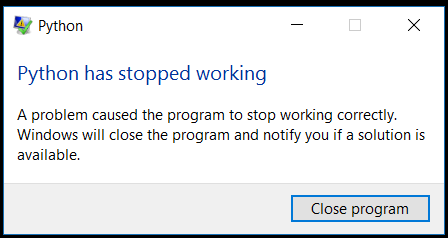




I once again have SDD in the last period of the day, so I should be able to work on this more then.

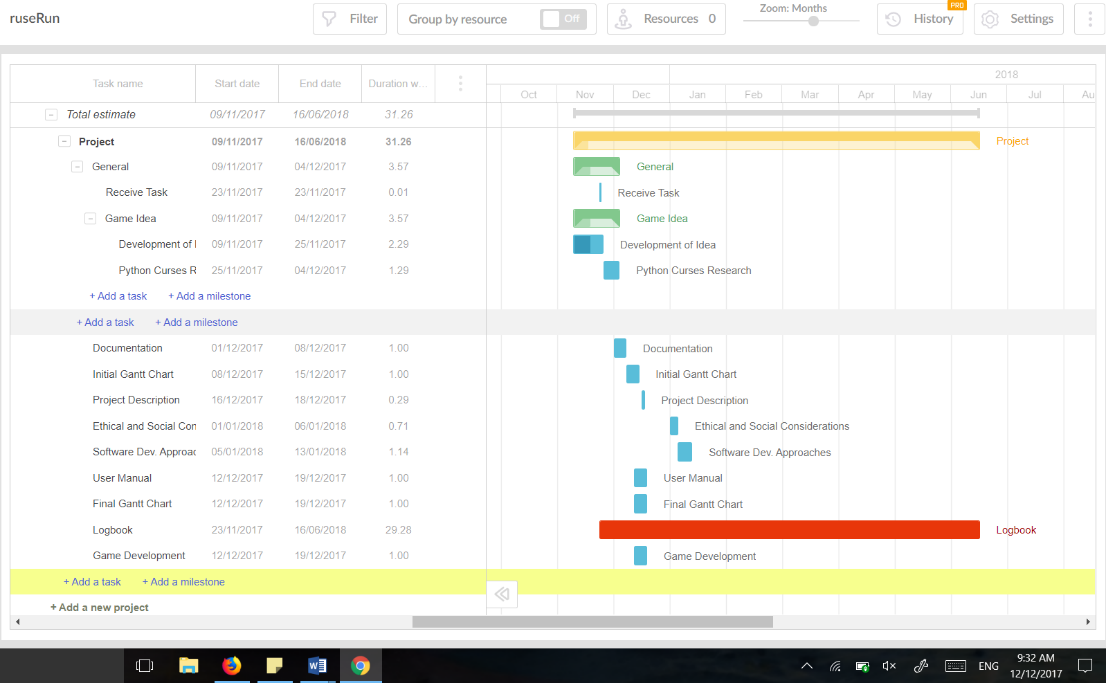
09/12/17

Lol, we didn’t exactly have a productive SDD lesson yesterday, so I got exactly zilch done. On the up side, today I managed to get loads of stuff done, until I hit the wall that is the update\_panel() command. Let’s start at the beginning. I first wanted to make the code more structured, so I moved the whole if statement into a separate function.

Then I began to once again work on the map conundrum. Firstly, when the character moves through the world, there are dots in between the walls representing floors. Of course, when the character moved through the world, they couldn’t just leave floor everywhere, as there are other things to consider too, like items or monster carcases. You know, the normal things you see lying around. I therefore had to figure out a way to make the character leave behind them the same tile as was there in the beginning. I made this whole weird complex solution that took ages, and was very long. But then, I was sitting there staring at the code, and I was all like ‘There has got to be an easier way to do this!’, so I went on the curses docs, and found a more elegant solution in the form of panels. The idea is that you can layer these panels, so that they wouldn’t affect one another. It was at this point that I hit the wall.

After getting rid of the chunky solution and implementing this one, for some reason python suddenly closed. I managed to narrow the problem down to one line – the update\_panel() line. However, as I have no idea as to why this is, I have temporarily hit a wall. I think I’ll just try again tomorrow…

12/12/17

Well, nothing really ended up being done on Sunday after all. Or Monday for that matter. It is not Tuesday, and I still have no idea as to how to fix this. Luckily, my Dad’s getting back from New Zealand this afternoon, so hopefully he’ll be able to help. In the meantime, I’ve begun the Gantt Chart. It’s very annoying and stupid, and I have no clue as to which thing we’re meant to use. Despite this, I’ve gotten a good way in, though I think I’m a little too optimistic…

26/12/17

Today, after a long break which included a holiday at the beach, the end of school and Christmas, I am finally back with the intent to continue my project, especially in respects to this panels issue that I have for so long been struggling. Though my Dad returned from his travels a while ago, I didn’t get back to this until this morning, when he reminded me of it, and so we once again jumped into the wondrous world of programming.

\*Phew\* well that was slightly confusing and unnecessary. I wasn’t sure how to start so I just started rambling. Today I got back into programming. I enlisted my dad to help me out with my problem, and made him a collaborator on my github repository so that he could edit the game. We tried running it on his computer, and to my surprise and moderate annoyance, it worked perfectly. It seemed that there was nothing wrong with my code at all, but simply the unofficial port that I’m working with for windows doesn’t support panels. It’s not really that much of a loss. It should be easy enough to do by hand (or computer!).

As I already had my dad enlisted and on loan (so to speak) for a while yet, I decided to pick his brain as to my map conundrum. If you don’t recall, this was the fact that I had no current way of doing the whole scrolling map thing that is seen in so many roguelike games. It would be doable by hand, but it would be significantly easier if I didn’t have to make this myself, but instead used a module inside curses.

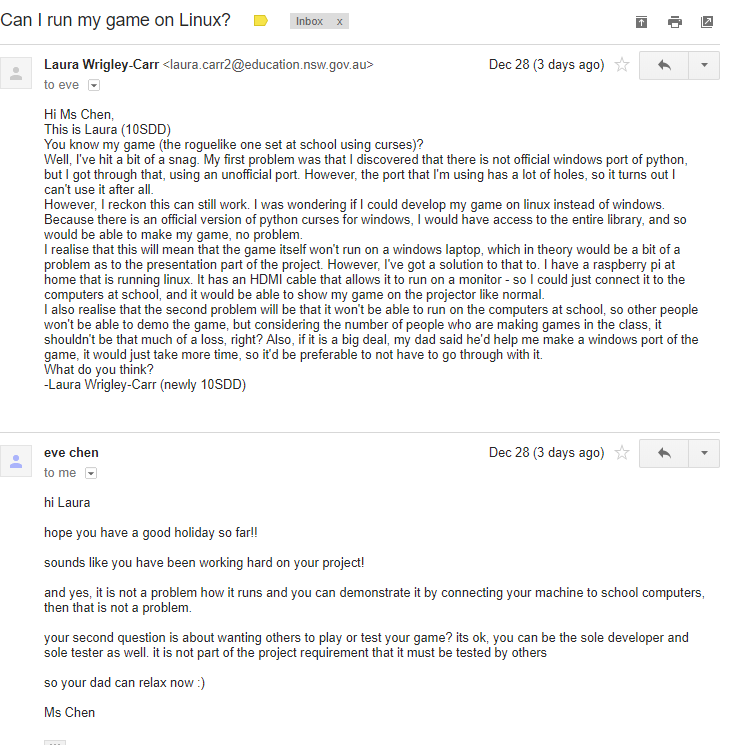
Of course, all my searches on the internet related to maps or scrolling maps and curses had turned up nothing, but my dad sits down at the computer and 10 seconds later, he’s got a solution. (Why is he so good???) By using pads (weird name, huh?), you can have a larger window, but only show a certain part of the window at a time. Using this, it should be easy to just get the part of the map shown to move when the player reaches the approximate centre of the viewpoint (not sure if that’s the right word…). I’m going to continue to peruse both this and the panels problem tomorrow.

27/12/17

Today I made tried out the pads solution. However, I hit a snag very quickly. It turns out that my port of curses for window doesn’t contain the parts for both pads and panels. (note to self: never again trust anything that says ‘unofficial’ again) Here I find myself in a bit of a pickle – without pads, it is neigh impossible to go through with this – I am seriously considering finding another module for the job. But I’ve been down that road before – remember that? I really don’t want to have to go down that road before. I think I’ll sleep on it, see what else I can come up with.

28/12/17

I had a chat with my dad this morning – and we’ve come up with a plan. If I could make the game on linux, then I could bypass my crummy curses module, and the annoyingness of windows altogether. We came up with another way to do it – see email below for details. (I emailed Ms Chen (hi!) for confirmation that this is technically allowed.)



31/12/17

Today I continued (on linux!!!) on my project. In order to continue working on my laptop (it’s easier), I am using ssh and working on one of my dad’s computers. Let’s just say, a lot of things have happened today, as I have been very productive, and frankly, I don’t remember most of it. I will therefore, just this once, resort to dot point form.

1. I discovered that you cannot use panels inside pads, and so decided to just use pads, and do the panels myself (easier than the alternative)
2. I attempted to familiarise myself with vim (the code editor on storm, faster but I need to learn how to use it properly)
3. I split the ruseRun.py file into two, one still called ruseRun, and another called interface (yes, you guessed it – to handle the interface. There’s your intrinsic documentation!!)
4. I tested it, found a small issue, and, being unfamiliar with vim (the editor on storm for editing code), I accidentally deleted the interface file.
5. I did steps two and three again to make up for step four
6. I wrote this log, being very sick of coding altogether

That’s about the size of it. Good night, I am extremely sick of coding. Happy New Year!!