Project 1: Minesweeper

Official Fortran Fanclub

Meeting Log

1. Date: Friday, August 31  
   Location: In-class  
   Outcomes: Decided on meeting schedule  
   Attended: Everyone
2. Date: Wednesday, September 5  
   Location: In-class  
   Outcomes: Reviewed progress so far (Pygame)  
   Attended: Sydney Combs, Daniel Gonzalez, Daniel Hidalgo
3. Date: Wednesday, September 5  
   Location: Spahr Library  
   Outcomes: Mapped out broad overview of functionality, classes and methods  
   Attended: Sydney Combs, Daniel Gonzalez, Daniel Hidalgo
4. Date: Friday, September 7  
   Location: In-class  
   Outcomes: Reviewed progress so far  
   Attended: Everyone
5. Date: Friday, September 7  
   Location: Spahr Library  
   Outcomes: Divided work between members, mapped out class interaction  
   Attended: Sydney Combs, Daniel Gonzalez, Nathan Pelletier
6. Date: Monday, September 10  
   Location: In-class  
   Outcomes: Reviewed progress so far  
   Attended: Everyone
7. Date: Monday, September 10  
   Location: Spahr Library  
   Outcomes: Set up GitHub repo, began combining files  
   Attended: Everyone
8. Date: Wednesday, September 12  
   Location: Eaton Commons  
   Outcomes: Coding session  
   Attended: Everyone
9. Date: Wednesday, September 12  
   Location: Spahr Library  
   Outcomes: Coding session  
   Attended: Everyone
10. Date: Friday, September 14  
    Location: Spahr Library  
    Outcomes: Coding session – achieved working game, looked at solutions for New Game button and getting input from the game window  
    Attended: Everyone

Workload Division

Sydney Combs:

* Get start-game input
* Run game
  + Revealed/flag tiles
* Put together write-up

Daniel Gonzalez:

* Buttons/UI
  + New Game
  + Help
  + Quit
* Update tiles/place flags

Daniel Hidalgo:

* Board class
  + Create minesweeper board
* Doxygen
* Cell class

Nathan Pelletier:

* Recursion – when a tile is revealed, if it is not a mine, reveal all orthogonally adjacent tiles until tiles adjacent to a mine are revealed
* Labeling of adjacent tiles
* Game over function – when a mine is clicked, end the game

Challenges

* Learning Python
  + Learning Python was one of our biggest challenges. The majority of our team had marginal experience with the language before this project, so a sizeable portion of our time was devoted, not to coding Minesweeper, but to familiarizing ourselves with the language itself. This difficulty seems to have resolved itself, as we have gained deeper familiarity with Python in working on this project.
* Using Pygame
  + We chose to use Pygame for optimal user interaction. However, we had difficulty getting Pygame’s functions to do what we wanted (i.e. drawing a rect, etc.), among other things. We were able to create a working game, but it’s not unlikely that we will experience similar difficulties in the future if we choose to continue working with Pygame.
* Using Git/GitHub
  + We experienced a number of difficulties with pushing to and pulling from the GitHub repository (in some cases having to re-clone it entirely). These issues were resolved fairly painlessly.
* New Game button
  + The New Game button gave us quite a bit of trouble. We wanted to give the user the option of starting a new game by clicking a button at the end of the previous game. However, actually implementing this functionality was challenging. We were able to create a new game only a few times before getting a segmentation fault. We resolved this issue by removing the feature.

Unused Features

* Clock
  + We discussed adding a clock during our planning stages. However, this feature was never implemented.
* Get input from the game window
  + We originally planned to have all user input be obtained from the game window itself, and not the terminal, in order to optimize the user’s experience. However, this proved more difficult than we anticipated. Although our team was able to implement a rudimentary method of obtaining user input in this way, we ultimately decided against using this version, as we were too close to the deadline for comfort.
* New Game button
  + Our team also planned to include a New Game button, both at the top of the game window and at the end of any given game. Unfortunately – possibly due to our class design – implementing this feature was extremely difficult. Our final version, before scrapping the feature, allowed the user to play three or four games before a segmentation fault occurred. We were unable to mitigate this issue, and ultimately chose to remove the New Game feature entirely.

Reflection

* Sydney Combs:
  + I would like to have organized a more in-depth design and understanding of who was working on what. I think that we should have mapped out classes, methods, interactions between classes, and functionality in much greater detail. We also waited quite a while to actually start writing code, so in the end we had much less time to work than we could have (despite it having worked out, I think in the future we should be more mindful of such things and set ourselves deadlines).
* Daniel Hidalgo:
  + I waited too long to see who needed help and with what. I’m not the boss, but do have the most Python experience, so I could have helped out more throughout.
* Daniel Gonzalez:
  + Before we start working on our next project I’d like to take more time to determine what each team member’s work needs from other team member’s; i.e. which classes and functions will be called across files, and who is in charge of each file. I would also like to begin working sooner, both together and individually, so that we can merge files sooner.
* Nathan Pelletier:
  + For the next project, I would like to better communicate the flow of the project as a whole, specifically things like the hierarchy for the classes, in advance.  I think this would lead to less confusion over the course of the project. In addition, I think it would be best to first make sure we can get a project that meets all of the requirements as quickly as possible, and then improve it step by step so we don’t have to do any panicking at the end.
* As a team, we agreed that we needed to plan out the work much more thoroughly, and to collaborate more efficiently and effectively.

Citations

Pygame