Agenda:
- What do we want to get done in the next two weeks for each category?
  - Documentation
    - Suggestion:
      - Work on all 3 (presentation, high level, detailed) a little each day and by end of two weeks it'll be finished (or mostly finished).
      - High-level: 3-6 pages
      - Detailed: 8-12 pages
  - Artwork
    - Suggestions:
      - start with creating campus map level
      - towers
      - 2-3 enemy types
  - Programming
    - Suggestions:
      - setup defined paths
      - setup multiple enemy spawns
        - create enemy that attacks towers
        - create enemy that attacks base
      - make towers auto-fire
      - setup base health
      - setup tower health
- This two-week period (sprint) ends on October 18 (Saturday).

Initial Notes:
- If needed, here's a good 11 minute intro video on using Github for Windows software: https://www.youtube.com/watch?v=u12zHGRfiKU
- We'll need to start on the presentation, high level design document and detailed design document fairly soon. The due date is October 23.
  - We are Group 1 so we're the first group to present.

Meeting Notes:
- We can use Inkscape to vectorize campus map.
  - Inkscape is free software for creating vector graphics.
- David has already started working on auto-firing towers.
- High-level design document needs to be done before the detailed design.
- Towers will shoot medieval weapons -- arrows, boulders, etc.
- We are able to draw the available paths onto the map level.

- ○ Besides the general A* path finding algorithm, we can program a way to add some randomness to the path walking of the enemies. Use a counter to keep track of how many go each way (such as at a fork).
- How will the enemies attack the base? Sit outside it and hit, or disappear inside?
  - ○ The Student Union will have tower points, so the enemies sit outside the tower attacking it, and the Union has the ability to defend itself (as long as the player puts towers on it).
- Google Play Service has a $25 fee to use. Another possible option is GameJolt.
- One way to show projectiles hitting is to fade them out using opacity. This is a possible alternative to making new artwork for an impact/hit.
- David mentioned needing a state flow/diagram for the upgrading of towers. We can use arrays in GameMaker, so that will be helpful to track the state/upgrade status of the towers.
- For Git, the Master branch has been reset back to where it's needed. For future features/prototypes, let's make a new branch. For example, David made the AutoFire branch for his feature of making the towers auto-fire at enemies.
  - ○ When we're ready to put the feature back into the Master branch, we will do a merge request in Github so everyone can review it and then we can choose to merge it into the Master.