### Name: Qidu Fu

### AndrewID: Qiduf

### TP1: Project proposal

* + **TP2 Update**
    - * I made my birds:
        + 1) change their direction,
        + 2) add the acceleration features to my flowers,
        + 3) add different flapping speeds to my code
        + 4) reorganized my code in a cleaner way. These are the main features that I added.
      * However, in terms of changes to my overall plan/goals, I have not made any design changes to my initial plans. I may add more customized features after I reach MVP.
  + **TP1 Update**
    - * There are not many updates on the following sections. The only update would be that I am getting stuck on making the helper birds, which can fly automatically. I have not figured out the logic yet. Other than that, I have finished my player bird and flower classes. I have not made changes undesigned changes to my initial plans. I may add more customized features after I reach MVP.
  + **Project Proposal Components**
    - **Title and Brief Description**:
      * Title: This term project titled “A Game of Flying Birds” is based on the scaffolded project developed by Mike (15112 instructor, Mike hereafter), which originates from the bee game, the [Google Doodle for Earth Day 2020](https://www.google.com/doodles/earth-day-2020) .
      * Brief Description: As stated earlier, this term project uses the main ideas from the scaffolded project developed by Mike. Thus, it contains three main components: the main flying bird (player flying bird, as Mike calls it player bee) moves along the cursor and collects the pollen (I know that it may not make sense to relate birds with pollination, but I do not like bees.), two flying birds (helper flying bird, as Mike calls it) automatically help the main flying bird to collect the pollen, and flowers. Each pollinator (flower) can be gathered twice, but each pollinated flower can only be pollinated once.
    - **Similar projects**: After I look through some term projects built by previous students for 15112, I do see some similarities between the flying bird game that I want to build and their projects. Specifically, my expected project and Lucia Shen’s project seem to have a lot in common (<https://www.cs.cmu.edu/~112/gallery.html>). For example, in her project, roadblocks (maybe, another kind of animal, I am not sure what they are) were generated automatically, preventing the dog from going back home. This feature is similar to the moving and flowers in my expected project. However, in Lucia’s project, a cat and a duck follow the dog automatically. Although this feature is similar to the helper flying birds in the class, it is different in the aspect that my helper flying birds may not follow my players. I can see after my player collects the flowers/pollination. There is no need for me to make the helper follow the player’s flying bird since the flowers/pollination would be gone.
    - **Structural Plan**: Inspired by Mike’s idea, this project will consist of two major components. Component one includes three major classes, namely, player flying bird class, helper flying bird class, and flower class, which will carry methods that can be called to bring the expected features into reality, such as pollination. The second major component will include the actual animation function calls, such as onAppStart, which can be used to call the methods/functions created in the aforementioned classes. Basically, the second component will integrate the three major classes created in the first component and animate them.
    - **Algorithmic Plan**: I feel like the trickiest part of the project is to have the distance-dependent speed. In other words, the further the distance is, the faster my bird will fly. I am thinking about using a loop. Also, maybe, I can just update the current location by adding the distance divided by a constant.
    - **Timeline Plan**: 1) The player flying bird class and flowers will be created by the first half of TP1 deadline and features such as pollination and flowers movements will be added to the classes by the end of TP1 deadline. 2) Helper flying bird class and its features will be added during the first half of TP2, and instructions will be added by the end of TP2 deadline. 3) Further instructions and customized features will be added by the TP3 deadline.
    - **Version Control Plan**: I will use GitHub to back up my term project code, and I will update/commit the backup file every time I stop working on it. See attached picture.
    - **Module List**: No external modules will be used other than the built ones, such as math, random, and cmu\_graphics.
  + **Storyboard**

Please see the storyboard.png in the folder.