# Important

1. Due Date: **02/12/2020 at 11:59 pm**
2. This homework is graded out of **100** points plus a possible **20** bonus points.
3. This is an individual assignment. You may collaborate with other students in this class. Collaboration means talking through problems, assisting with debugging, explaining a concept, etc. Students may only collaborate with fellow students currently taking CS 2316, the TA's and the lecturer. You should not share your code, visually or electronically, or write code for others. For individual assignments, each student must turn in a unique program. Your submission must not be substantially similar to another student's submission. Collaboration at a reasonable level will not result in substantially similar code.
4. For Help:
   - PyQt5 Handout
   - Documentation
   - TA Helpdesk (Schedule posted on Canvas)
   - Email TA's or use Piazza Forums Notes
5. Comment out or delete all function calls and unnecessary print statements. Only global variables, and comments are okay to be outside the scope of a function. When your code is run, all it should do is run without any errors.
6. Do not wait until the last minute to do this assignment in case you run into problems.
7. Read the entire specifications document before starting this assignment.
8. IF YOUR CODE CANNOT RUN BECAUSE OF AN ERROR, IT IS A 0%

# Introduction

The goal of this homework is to showcase your knowledge of Graphical User Interface (GUI) using PyQt5. You should test the classes and methods out first on your own computer, then when you have one or more of them working, upload the entire file (which must be named HW03.py) to GradeScope. Since this is a GUI homework, we cannot deploy an auto grader on GradeScope, thus verify that your code is correct by reading the instructions below and testing accordingly.

# Background

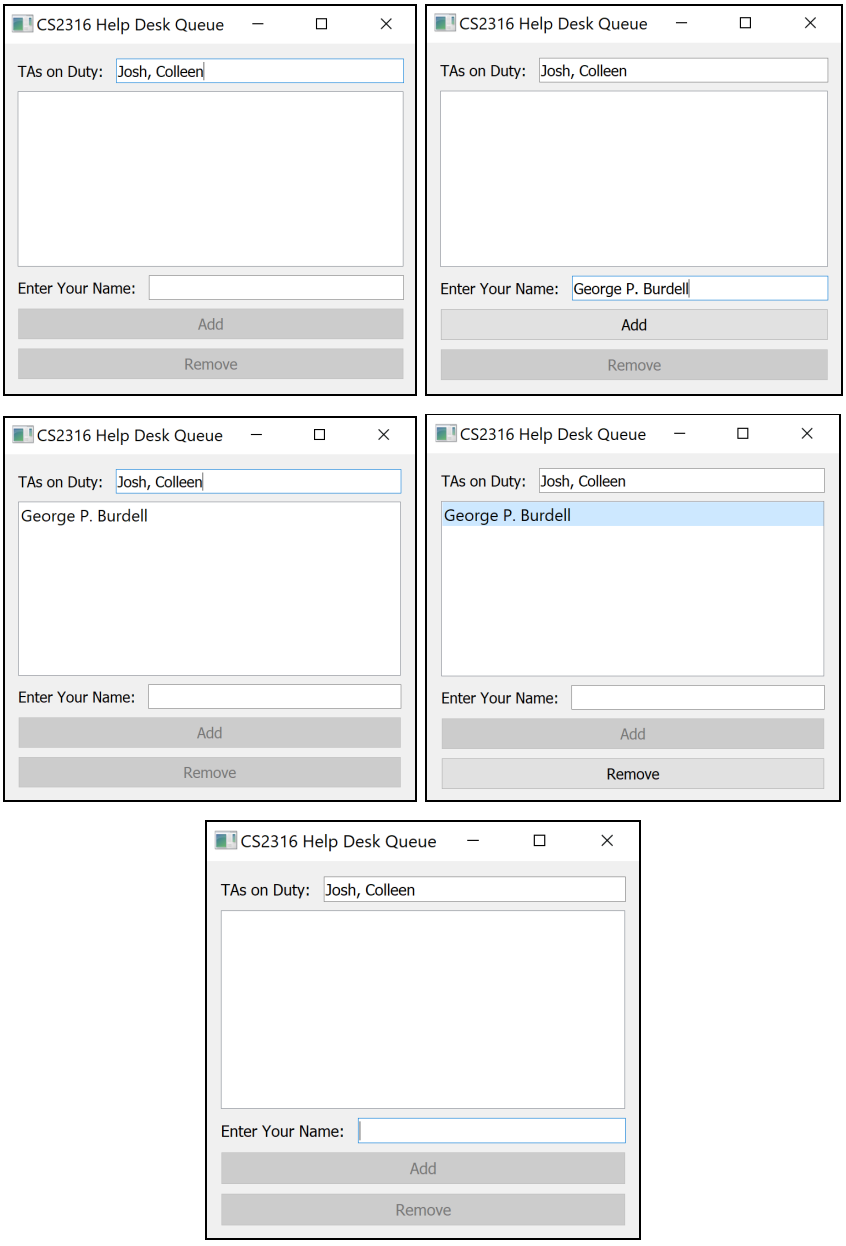
As you all may know, the CS 2316 Help Desk can get quite crowded, therefore we need a queuing system to manage the students waiting for help! In this assignment you will create a simple queue for the CS2316 Help Desk using the PyQt5 library. After grading, the TAs may select the best GUI, both practically and aesthetically, to use at our pod during office hours.

# Requirements

*Note: See the grading rubric at the end of the document for minimum requirements and point distribution.*

- **Assignment Instructions**
  - **Set The Window Title**
    - "CS2316 Help Desk Queue"
  - **Label and Textbox**
    - Create a label object that displays the text "TAs on duty"
    - Add a textbox next to the label so TAs can add their names
      - The label and textbox should be formatted horizontally as shown in the screenshots below
  - **ListWidget**
    - Create a ListWidget object that will display the names of the students in the queue
    - Each time a student is added to the queue, the ListWidget should add a new row
  - **Label and Textbox**
    - Create a label object that displays the text "Enter your Name"
    - Add a textbox next to the label so students can enter their names
      - The label and textbox should be formatted horizontally as shown in the screenshots below
      - After a student's name is added to the queue, clear the text from the text box
  - **Buttons**
    - Create two separate buttons below the textboxes, stacked vertically as shown in the screenshots
      - Add
        - This button should display the text "Add"
        - If there is text in the text box (a student's name) this button should be enabled. Otherwise this button should be disabled.
        - When clicked, this button should add the student's name to the queue.
        - After clicking this button, the text box containing the student's name should clear.
      - Remove
        - This button should display the text "Remove".

- If a student's name in the queue is selected then this button should be enabled. Otherwise this button should be disabled.
- When clicked, this button should remove the selected student's name from the queue.



# Testing Your Code

Once you have the basic structure of the GUI set up, you can write your code for the additional features, run your program, and see if the GUI showed up matches your expectations. Remember, if your code cannot run because of a runtime error, you will receive an automatic **0**.

# GradeScope Submission

- Make sure the file submitted is titled **HW03.py**
- Do not import any modules, packages, or libraries other than time, sys, or PyQt5
- Submit the HW03.py file only.
- Your assignment will not be autograded. GradeScope is being used for submission only. Your assignment will be manually graded.

# Grading Rubric

Below we have an itemized grading rubric that serves as a baseline for HW03. You may add any additional functionality or aesthetics as long as the requirements below are met.

- Window title that displays the text “CS2316 Help Desk Queue” (**5** pts)
- A label that displays the text “TAs on duty” (**5** pts)
- A functional text box or similar area for TAs to enter and display their name(s) (**10** pts)
- An object (ListWidget, ListView, TableView, etc) to display students in the queue is present and functional (**10** pts)
- A text box for a student to enter their name is present and functional (**10** pts)
- A button to add the student’s name to the queue (**25** points total)
    - This button is present and functional (**5** pts)
    - This button is enabled only when a student’s name is in the respective text box, otherwise it is disabled (**10** pts)
    - This button adds the student’s name to the display (e.g. the ListWidget) when clicked (**5** pts)
    - After the name is added to the display, the text box containing the student’s name clears the current text (**5** pts)
- A button to remove the student’s name from the queue (**25** points total)
    - This button is present and functional (**5** pts)

    - This button is enabled only when a student's name is selected within the ListWidget, otherwise it is disabled (**10** pts)
    - This button removes the selected name from the display when clicked (**10** pts)
- All widgets are in their correct relative positions within the window as outlined in the instructions. The window contents remain in these positions even when the window is resized (**10** pts)

# BONUS

- There is a ***maximum*** of **20** bonus points you can earn.
- If you decide to do the bonus ***you must*** include a comment at the top of your file with a brief description of the additional features you've implemented. This is to ensure that you receive full credit and the TAs do not miss anything when grading.
- Ideas for earning bonus points:
    - Changing the color scheme or style of the GUI (check the documentation!)
    - Using stylized fonts or colored text to make the GUI more aesthetically pleasing.
        - This includes making important text larger.
    - Including the current time somewhere in the GUI (importing time module is allowed).
        - The time should update regularly and use the EST zone.
    - Enable keyboard shortcuts that allow the user to easily add and remove people from the queue
        - For example, the enter button adds names and the backspace button removes names
    - Display students names in a table along with other information like arrival time, wait time, etc.
    - Calculate the average wait time and display it within the GUI.
    - Create a predetermined combo box to select TA names and display the names in another area.
    - Be creative! The PyQt5 documentation is extensive and there are many widgets, styles, and layouts that can be implemented to improve the GUI. Go above and beyond the basic requirements, and you will earn bonus points.