Assignment 10 – Reflection Assignments

1. Assignment 1
    a. Do you still want to major in CS? Why or why not?
        i. Yes, so much yes. It's what I've always wanted to do and this class has not changed that.
    b. How would you approach this assignment differently? This can include picking different companies or areas to research?
        i. I probably wouldn't. I still like those companies and I still like what I would have done. I might pick more companies that specialize in network security.
    c. What did you learn from this assignment?
        i. I didn't learn a whole lot. I already knew what I wanted to do before I joined this class.
    d. Did you like the assignment? Why or why not?
        i. The assignment wasn't bad per say. It was less useful for me because I already know what I want to do but I presume that for students who don't, it could be helpful. I remember I really wanted to jump into the actual code so if I'm honest this assignment was a bit disappointing for me.
    e. How would you change this assignment for future students?
        i. I don't think I have any good input to change this assignment for other students. It's well rounded.
2. Assignment 2
    a. How would you approach this assignment differently? This can include spending more time on completing the assignment, developing a better test plan (bad/good inputs), coming up with a different solution, etc.
        i. All of my solutions were correct. I did most of the calculations in my head. I don't think I would change the way I did this assignment.
    b. What did you learn from writing the steps to convert a base 10 number less than 256 to a base 2 number?
        i. When I learned how to do this I learned the real meaning of a base of a number. It makes it easier to understand how math was setup to be base 10 and how computer works.
    c. Did you like the assignment? Why or why not?
        i. It was tedious, but useful.
    d. How would you change this assignment for future students?
        i. I would add more problems to the first part and make it more points while giving less points to writing the algorithm.
3. Assignment 3
    a. Did you like robozzle? Why or why not?
        i. It was just another puzzle solver. It was a bit outdated. Some people had a hard time getting it to run in the first place.
    b. What did robozzle teach you?
        i. Algorithmic thinking.
    c. What did you learn from writing the steps to finding the largest number?

        i. How to wait and select things.
    d. Did you like the assignment? Why or why not?
        i. Somewhat. It was basic overall. Just a loop and an if statement.
    e. How would you change this assignment for future students?
        i. Let them use for loops if they know it already. It's ridiculous to restrict that it's so fundamental in programming.
4. Assignment 4
    a. What did you struggle with the most in this assignment? Why?
        i. I wanted to use a for loop so, so bad.
    b. Did you spend enough time designing the algorithm for converting the base 10 number to a binary number in assignment 2 and 3 before coding in python.
        i. No I spent almost no time on the design before I wrote it. It wasn't worth the time to design it before.
    c. How would you approach this assignment differently? This can include variable names, spacing, logical differences, etc.
        i. I don't think I would approach it differently. It was super easy to do it with a string build. I suppose I could do it with a list and then join it but I don't know how that would be more helpful.
    d. Did you like the assignment? Why or why not?
        i. I did. It was pretty interesting actually. I've never really had to convert this type of thing manually. It was something different and I liked that.
    e. How would you change this assignment for future students?
        i. I think I wouldn't change this assignment. Except for letting us use for loops. Heck I think that you should allow conditions as well, but I understand that would break the fundamentals of the class at that point.
5. Assignment 5
    a. Did you design the programmer vs. scientific calculator BEFORE coding in python? Why or why not?
        i. No. I wrote it first. I wrote it first because I was able to write it anyways because it's just so simple.
    b. How would you approach this assignment differently? This can include designing before you code, putting more effort into the design, choosing a different solution, etc.
        i. I might put more effort into my design. I think that I didn't spend enough time on the design as it was. The code itself is tight though I'm a fan.
    c. What did you learn about coding and testing from this assignment?
        i. I think I learned how to do the testing like you want, but I don't think I learned anything about coding itself while wrote it. I suppose I used modulo more than before.
    d. Did you like the assignment? Why or why not?
        i. I did like the assignment. It was difficult to figure out what I needed to do. Once I figured out what I needed to do it was easy.
    e. How would you change this assignment for future students?
        i. I don't think I would change this assignment. I still think the code should be worth at least double the design.

6. Assignment 6
    a. What did you struggle with the most in this integration assignment?
        i. I think the hardest part was nailing the rectangular approximation function. It was somehow harder than the trapezoidal one for some reason.
    b. Did you design BEFORE programming the assignment? Why or why not?
        i. I did the program first. I usually do the program first. I don't think I will design a program first if I am writing it by myself. And even still, I probably only take minimal notes at best.
    c. How would you approach this assignment difficulty? This can include looking at the point breakdown in the assignment, going to TA office hours, etc.
        i. I think I would ask people questions if I had any. I'm not sure I would have had any other differences. I quite like how I did this class.
    d. What did you learn from this assignment?
        i. I learned how to programmatically do integral calculations using approximations.
    e. Did you like the assignment? Why or why not?
        i. I liked the assignment. I think that there were some problems with it. Specifically that many people who looked at the assignment had to ask me what an integral was and I don't think the writeup did a good job of explaining that.
    f. How would you change the assignment?
        i. I think I would make it a different kind of math operation.
7. Assignment 7
    a. What did you struggle with the most in the ASCII art assignment? Why?
        i. I think the spacing was the issue. I don't really think there was anything else that was an issue.
    b. Did you create a function for each letter?
        i. Yes
    c. Did you struggle with the functions or logic with the assignment?
        i. No
    d. What did you learn from this assignment?
        i. I learned a good way of printing horizontal text using ASCII art.
    e. Did you like the assignment? Why or why not?
        i. I thought this assignment was a step backwards in difficulty. I think that the integral math was a good assignment and then all we had to do after was print text. I wish we did other things after that.
    f. How would you change this assignment for future students?
        i. I would probably put it before other assignments.
8. Assignment 8
    a. What did you find to be the hardest part of the turtle assignment?
        i. I think the hardest part of the turtle assignment was to keep all the letters on screen.
    b. How would you approach this assignment differently? This can include making your diagrams clearer, writing better pseudocode, etc.

        i. I think I would write better pseudocode. I had a fine time writing this but I think if I explained it to someone it may be a bit confusing.
    c. What did you learn from this assignment?
        i. I learned that even when the graphics driver does all the heavy lifting, it can still be difficult to do some yourself.
    d. Did you like the assignment? Why or why not?
        i. Yeah. I thought it was fun to do the turtle drawings and see it draw out all the letters. It was an interesting assignment to say the least.
    e. How would you change the assignment for future students?
        i. I don't think that I would change the assignment. I think that I would keep this one the same and in the same place. It's well suited in difficulty at this point in the class.
9. Assignment 9
    a. Did you take design seriously for the TicTacToe assignment? Why or why not?
        i. No, because I wrote it in about 90 minutes the Monday it was assigned. I didn't do the actual writeup until the Sunday it was due.
    b. What did you struggle with most in this assignment (design, implementation, or both)?
        i. I think the design was harder because I had to sit there and write it up. Plus I had to change the design every time I changed the program.
    c. How would you approach this assignment differently? This can include time management for debugging, asking for help, etc.
        i. I would not approach this assignment difficulty. Maybe I would spend time writing the AI part.
    d. What did you learn from this assignment?
        i. That Python doesn't have ENUMs on any version before the one on my laptop.
    e. Did you like the assignment? Why or why not?
        i. I did like the assignment. It was a fun one to write because I like to make games.
    f. How would you change this assignment for future students?
        i. I wouldn't.