Homework 2 — Individual Project Proposal

Due October 4th at 6pm

(50 points)

Credit:

- Turn in: `individual_project.pdf`
- Schedule a meeting with Felix (it must be between October 7th and October 9th (inclusive; there will be a signup the week preceding)

Instructions:

You have chosen to prepare to embark on a 10-week long individual project journey. Be aware that receiving a good grade on this assignment is **not equivalent to approval. You must obtain explicit instructor approval** to do an individual project for the rest of this class. This approval will be based on this homework, how appropriate your project is for the requirements and timespan, and your prior submissions in this course.

You may work with up to one partner on this project. If you choose to work with a partner, your project should be appropriately scaled up in scope versus if you were working alone.

Project Requirements:

Your project must incorporate the following:

1) Version control (+ continuous integration, *strongly* recommended)
2) A testing framework
3) Use of an object-oriented language or paradigm
4) Appropriate use of inheritance
5) Design patterns — 2 of the 5 following design patterns: Singleton, Flyweight, Iterator, Factory, Prototype
    a) You may request to substitute one of these design patterns with one not listed here—to do so, describe the chosen design pattern, why it is appropriate for your project, and how you will use it.
    b) Note: when you are writing this proposal, we will not have covered these in class—there are some resources available to you in the resources.md document on github, you are welcome to come to office hours to get further information, and you are strongly encouraged to do some of your own research. Your proposal doesn't have to be perfect—that's why it's a proposal.
6) An appropriate user interface — this can be a GUI or a text UI, but the interface must make sense in the context of the project
    a) GUIs are ***strongly*** recommended.
7) A project presentation during the final week of class; details given after fall break. Expect to demo your project and give a presentation describing its architecture as well as technical hurdles that you encountered as you were developing it.

If you have a *very* strong reason that you should not have to include one of these, you may request an exemption. Any exemption requests that are “it will be easier” will be denied.

Projects that propose using Unity (or Unity-like environments) this semester must be *very* explicit about how you will be meeting the above requirements, what resources you'll be developing yourself, and what resources you'll be using from other sources. We do not recommend taking this path unless you already have experience using one or both of Unity and Qt.

Turn in a document with sections addressing the following:
Expect this document to be **at least 4 pages long**. Format it in a sensible fashion that makes it easy to read. Even if you need to change some details later, provide as much information as possible now. Use tables sensibly to format your proposal and make it easy to read. Include pictures or diagrams as helpful.

We expect you to do a significant amount of research and planning for this proposal.

1) What is your project?

2) What technologies are you planning on using? (Programming languages, libraries, etc. Be as comprehensive as possible. Rate each technology on a scale from “not familiar” to “completely familiar”)

3) What are the essential parts of the project (complete the sentence “My project won’t work if it doesn’t have _____” as many times as necessary). Why are these essential and do you have a backup plan if they don’t work?

4) What outside resources do you require? (for example, will you need a huge amount of genome data? Do you need art for your graphics? etc.) Where will the outside resources come from? Have you already located them and know that you have access to them? If you need outside data, be sure to consider any data cleaning and other tasks in your plan.

5) Make a proposal for your architecture. If you will have a front end and a back end, how will they interact, what will your database be? What will your data model be? How will your front end work? How will you deal with your user interface? How will it integrate with your object models? What classes/objects do you plan on having? What is each in charge of and how will they interact? How will the different technologies that you are planning on using work together?
    a) This should be a comprehensive initial plan of the objects you plan on having, what they will be in charge of, and how you will transfer data/information between the different components of your application.
        i) Make a proposal for how you will use inheritance with these objects.
        ii) Make a proposal for which design patterns you will use (we won't have covered them all in class yet, so do some research—we'll talk about your proposal during your meeting in the week that this document is due).

6) Detailed plan. What do you plan to accomplish each week between now and the final week of the semester? Include **all** of the following for each week:
    a) What will you turn in for each class due date? This should be a detailed account and should include what a user can expect to interact with/see/be able to do at that point.

(when the rest of the class is turning in homework 3, homework 4, and for the final project checkpoint, and the final project due date (see github for dates associated with these homeworks)). This plan can change over the course of the semester, but should not change wildly.

   i) You will be required, at each due date, to turn in a detailed plan of what you are planning to accomplish for the next due date.
   ii) The plan that you propose now for the homework 3 due date will be your plan for that due date.

b) What knowledge do you think that you will need to accomplish the goals for the week?
c) Of that knowledge, what will you need to learn?