CS/Math 371
Guidelines for the Final Project Report

Purpose & general information

Your project report must describe **your project** (the one you finished, not the one you proposed) as outlined below. This should be written as a stand - alone document, that means someone not familiar with your project and your work (that is, someone who has not read prior progress reports) should be able to read and understand this report.

The final report is mainly a description of **what** you have done and **how** your code solves all aspects of your project - together with other information we need to understand your approach. We need to judge how well your (electronic) project solves the problem you set out to solve. For that you need to tell us

        * what your project is
        * what you had to learn about before you could get started (research info)
        * how you represent data/information in your program or in files (files/folders and data structures)
        * how you put things together (overview, relationship of functions and their purposes)
        * if you **used code which was developed by someone else.**
        * how to execute your code and
        * how we can see and be impressed with the breadth the depth of your completed project.

Grading:
As already specified with the original project guidelines, the final project report will be graded using the following criteria.
        • Progress, overcoming obstacles, and project completion
        • Difficulty of the project
        • Quality of the general solution: Design
        • Quality of the specific solution: Implementation
        • Documentation provided with the solution

Turn in a **hard - copy** of this report (without code) to the main - office on Wednesday, May 13th, not later than 4:00 pm.

Also turn in a electronic submission on blackboard by that same deadline. The electronic submission needs to include **all** the supporting files (if we cannot run your project because you did not submit all the needed files your grade will reflect that you turned in a non - working project) and an electronic version of your actual report. The code should be organized using *Mathematica*'s cell structure - so it's easy find your way around all of your code.

Make sure you **follow the instructions** in 1 to ensure that **we can run the code without problems on a PC and a Mac**. As before put all your files in a folder and then zip up the folder. The file name of the folder **and** the name of the zipped file must contain your last names, do not turn in a folder (or file) called finalreport (or finalreport.zip), but something like ernstzieglerfinalreport.zip.

If you **give your presentation prior to turning in the project report,** make sure that you include a paragraph in your project report if there is code which did not run during your presentation but which works when you turn in your report (make it item 1.5).

Description of content

There are **10 required pieces** for this project report. Please address all in the order and with the label (= number) given below. If some items of 3, 4 or 5 do not apply to you then you need to still address them and explain why they do not apply. Only the signed statements may be turned in separately. The entire report should be typed - including the algorithms.

1) An **explanation of how to get your code to run**

An explanation of **how to get your code to run** and what it shows e.g."Evaluate the notebook Whatsit-sname.nb, and then execute the function call PlayGame[ ] at the end of the Mathematica file. You will see.... " Make sure that you (electronically) **include code to demo all major parts of your program** and include how we can start/see those demos as part of this section of your report. Remember that you need to make sure that we can run the code effortlessly. So use SetDirectory[NotebookDirectory[ ]] and use relative file paths with / (and no\ or \\) - as done in PortIt.zip.
If there are **several notebooks**, make sure you specify which one to evaluate or in which order they need to be evaluated. If you know that **your code will take a long time to run** make sure you mention this here. If you know there are errors that cause an infinite loop or a crash, state that as part of 8.
If you give us **wrong instructions** you will **loose up to 10% of the report grade** even if we somehow figure out what the correct instructions would've been. If we cannot figure how to run you project out then your report cannot earn more than 50%.

2) A **detailed description**

A **detailed description** of your project (not your proposal, but your actual project). It must satisfy the original requirements for the project proposal: It must be in complete sentences, start with a description, and include enough detail such that someone who does not know about your project understands what the project accomplishes. It must include
  • Several screen shots of what can be seen on screen;
  • explanations of the screen shot (s) as needed;
  • explanations and examples of how a user influences what happens.
Note: The screen shots mentioned above should actually be screen shots taken from your running project.

3) An **explanation of algorithms/formulas**

An **explanation of algorithms/formulas which are the result of research & development**. A number of projects are using theoretical models, algorithms, and formulas (Examples: determining the movement of a DNA fragment in a gel, the movement of planets and stars, the strategy for an AI for whatever game you are developing, the way rational tangles are drawn, music theory you are using, the flight path of an object, etc ...) which needed to first be researched or studied or developed. Include one or more examples in your explanations which makes it easier for us to follow your description. Make sure you use your own words in your explanation. Provide a reference for it (and if you have an electronic copy of the reference, please include it with the material you submit electronically).

Algorithms should be explained in full sentenced and you may include a description in pseudo-code, but do not turn in *Mathematica* code for this section.

Some projects are not research based. If your project did not require and ' outside' research, state that.

4) A **description of any files and/or file structure**

A description of **any files and/or file structure** your project uses. This refers to how you arrange your code or the data external to your code. Be specific about what information is stored where and why. Make sure that when your program reads and writes files, the names of the files are given relative to your project folder. Also provide an explanation of the format of the information in the file - preferably

with a short example. Some projects use no files, others use many. If your project does not use any, state that.

5) A **description of any major data structures**

A description of **any major data structures (lists, etc) your project uses**. This refers to how you arrange the data in your code. Be specific about **what** information is stored and **how** the information is stored and specify the variable **name** associated with the data structure (e.g blocklist) For the 'how' describe the structure of the list (e.g. A list of pairs of points, where each pair specified the lower left and upper right corner of a black surface in cubeworld....). An example might be helpful - e. g. copy it from your code and write an explanation. Some projects use no data structures, others use many. If your project does not use any, state that.

6) An **overview of how your code approaches the problem** (with list of functions and their info)

An **overview of how your code approaches the problem**. This should be written in complete sentences and be written such that it can be understood without knowing anything else about your project other than the detailed description from 2) and 4) and 5). You must include a **bulleted list** of the i**mportant functions** you wrote. (Note: There are very few reasons for your code not to be in the form of functions - even if some functions only initialize global variables.) For each function specify
> * its purpose
> * the input/information it requires (parameters, global variables, user input)
> * the output/information it generates (return values, changes on screen and to global variables)
> * a short description of its algorithm

Note: unimportant functions are functions which are trivial and don't need to be mentioned. If you have LOTS of important functions, then use good judgment on what to include.

7) An **explanation of externally used code**

If you have included code from the Internet, friends, the book, or blackboard, you need to clearly state this in this section. If you wrote all your code on your own, then state that as the answer to this item. Provide the reference where you found it and clearly state what the code you used accomplishes. If you found code and modified it to fit your purposes, clearly state that, too (also with reference, of course). Make sure it is clear to us how much of the original code is actually used and how much is your work. (If you found code that helped you learn about something for your project, you need to include that as references 9.)

8) A **description of the limitations**

A **description of the limitations** of your code, i.e for what kind of inputs does it work and for what kind does it not work (if there are such cases). Include any known bugs of your program in this section. It is better for you to be upfront about what doesn't t work rather than letting us find out. You **may** also include in this part of your project report a description about the difficulties you had in finding solutions and developing the code. However, you must also make sure that your description is accurate and consistent with our experiences in you seeking help from us.

9) A **list of references**

Any significant references you used to learn about issues related to your project and/or how to approach it (other than the Documentation Center).

10) A **signed stateme**nt from each group member

A **signed stateme**nt from each group member (separate statement/page for each member) with a **clear** description (qualitative and quantitative) of the contributions s/he made to the project. The description should clearly establish **what** was contributed and the **level** of contribution. Signed statements may be turned in independently of the project report as in the past.