**Final Project  
CS 5800, Fall 2024  
Dr. Alan Jamieson**

**DUE: December 9th, 1pm eastern via Canvas**

**Tl;dr version - you and your team will implement the project, create a writeup, and present your result during the final period.**

For this final project, you and your team will select and implement an algorithm from the list below.  
  
List of potential algorithms:

* k-Nearest Neighbors for continuous variables (input: data file)
* Johnson’s Algorithm using Fibonacci Heaps (input: an integer n denoting the number of vertices in the graph; graphs are to be generated randomly, with edge weights between 1-20; initial graphs have to be output in the visualization)
* Knight’s tour calculations (input: two integers for board size; output: a single example tour if one exists, error message if it does not exist)
* Simultaneous Localization and Mapping in a 2D space (no input needed, you should set up 3 static environments and show how the algorithm “solves” that space)
* Knuth’s Algorithm X as applied to a hyper sudoku puzzle (input: puzzle file)
* Christofides Algorithm (input: an integer n denoting the number of vertices in the graphs; graphs are to be generated randomly, with edge weights between 1-20; initial graphs have to be output in the visualization)
* Karger’s Algorithm (input: an integer n denoting the number of vertices in the graph; graphs are to be generated randomly; initial graphs have to be output in the visualization)
* ChaCha20 Cipher (input: 32 character string)

In addition to the above, feel free to propose your own algorithm that you find interesting!

It will be your task to implement the project and develop a paper about your project and your process. During the final period, you and your team will present your work in a 15 minute presentation. See below for more details on the paper and the presentation.

You are required to turn in all deliverables **before** the final period. In addition, you will need to submit your presentation slides. Only one set of submissions will be required per group.

In addition, each individual is also responsible for completing and submitting via email a team evaluation form. That form is available on the assignment page, and is due **before** the final period.

**Teams:** Teams will be 2-3 members. You may select your own teams, but those selections need to be in to me by **11:59pm eastern on November 8th**. To accomplish this, one member of your team should email me, copying the other team members, with your team information. Feel free to email me with incomplete teams as well. 3-member teams may have an additional member added. I will complete teams on the 9th.

**Algorithm selection:** Each team will meet and choose one of the topics above (or get approval for a different algorithm). Once selected, send that information to me **via email by 11:59pm eastern, November 22nd.**

**A note about help:** Due to the nature of this project, neither me nor the TAs will be walking through your selected algorithm. It is up to you and your team to look at available pseudocode and figure out how the algorithm works! Also, depending on how you are tackling the implementation, we may be limited help with programming issues.

**Implementation:** For your program, you will implement your selected algorithm including any caveats that were attached (for instance, 3D mazes). Once implemented, you will need to build a visualization of your algorithm in action. Inputs are required if noted, and clear instructions on how inputs are to be formatted should be part of your paper. Inputs could be handled as part of your visualization front-end, but not required. You are not allowed to “fix” your program to a set of predetermined inputs (unless otherwise noted). See the algorithm list on what is required for inputs for each algorithm. Language is the team’s choice, but clear instructions to compile and run your programs need to be provided as part of your paper. Any libraries, compilers, or languages used must be freely available.

The visualization should clearly show step-by-step execution of your algorithm along with a representation of whatever is being manipulated as part of your algorithm (graph, data set, puzzle, etc.). A GUI is not required, but may make the most sense for your algorithm.

**IMPORTANT NOTE:** that’s probably unnecessary - your team must create your algorithm *from scratch*. Math functions, data structures, and other support elements of your algorithm could come from libraries, but the algorithm itself must be your team’s creation. If there’s ever a question about a function or data structure, please ask me.

**Paper:** As part of this project, your team will also need to create a paper about your algorithm and your project. There is no minimum or maximum page length, just a content requirement. Your team should include the following components:

* An overview of the algorithm, including history.
* An illustrative example.
* Information about the technical parts of your project including language and library choices, with reasoning.
* Instructions on how to run your program including any 3rd-party libraries required.
* Instructions on required formatting of input.
* A reflection on the project - what went well, what didn’t, and what you’d do differently.
* Citations, using a standard format.

**Presentation:** During the final period, each team will be required to present their project during a 15-minute presentation. In that presentation, teams should present, at least, the following components:

* A broad overview of the algorithm.
* The project as implemented, including a demonstration.
* Details in regards to the creation of the project, including technical challenges.

A rubric will be provided on the assignment page. A pdf or pptx file of your final presentation is required to be submitted with the rest of your project deliverables.

**Slip day note:** Because of this being the final and that there is a presentation component to the project, it is not eligible for slip day use. The project must be completed with all deliverables submitted by the start of the final period on December 9th.

**Grading**: Grades will be split into the following categories and percentages:

50% Project - grading will be focused on the completeness of the project, if it meets the proposal, and how well it demonstrates the topic

20% Paper - grading will be focused on inclusion of all required components, completeness of explanation of components, grammar and spelling

20% Presentation - rubric based

10% Team review - rubric based

A non-exhaustive list of deductions:

-100 No show during the final period

-10 Missing team review at deadline

-10/each Missing deliverable (proposed deliverables, presentation slides)

-----------------------------------------------------------

**Teams info due:** November 8th, 11:59pm eastern, via email. Team assignments will be posted to Canvas on November 9th.

**Algorithm finalized:** November 22nd, 11:59pm eastern, via email.

**Due:** December 9th, 1pm eastern, via Canvas; presentations during the final period on the 9th.  
**Format:** team reviews via template posted to the assignment page via email. Presentation via pdf or PowerPoint file, and all other deliverables should be compressed and submitted via Canvas.

-----------------------------------------------------------