**Assignment 1: Visualize Subnetworks**

You are free to choose whatever programming language you like, but in general you may not use any bioinformatics toolkits or libraries. You are allowed to use libraries that implement data structures (lists, arrays, etc), but you must fully cite the source. For example, if you are programming in C++, the use of the Standard Template Libraries is allowed. If you are programming in python, you may use scipy and the numpy multidimensional array library. If there are any questions as to what is allowed, please send an email to mazen.alborno@ucdenver.edu.

This assignment is to test of your individual algorithmic design and programming skills. If you want clarification about anything else, you may email [mazen.alborno@ucdenver.edu](mailto:mazen.alborno@ucdenver.edu). Answers to clarifying questions will be sent out to all students.

You must submit your exam, which consists of the following components by the due date:

1. Written report  
2. Algorithm pseudocode  
3. Documented program code and program outputs.

Provide in your submission a README.txt file that includes detailed, concise instructions on how to install your program and its dependencies within a basic environment.

Grading

Your work will be graded based upon three components: a written report (25%), algorithm pseudocode (25%), and implementation (50%). The written report should describe and justify your strategy, define the input files, present the algorithm overview, define any scoring methods, and detail the expected output files. The report should also contain an analysis of the final results and discussion. You should cite the appropriate scientific literature where appropriate. The pseudocode should describe your algorithm in a code-agnostic manner. The implementation must include a working program with extensive comments describing each step, as well as describing all inputs and outputs.

Description

We provide 12 disjoint gene sets for Fanconi Anemia (FA) as input in the tab-delimited file ‘Input.gmt’ in the Broad Institute’s Gene Matrix Transformed (GMT) format (https://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data\_formats/). Also provided is a version of the STRING database of known and predicted protein-protein interactions (http://string-db.org/) in the tab-delimited file ‘STRING.txt’ where each line represents an edge in the network between two genes, weighted by the strength of their functional similarity. Your task is to display the functional network among Fanconi Anemia genes. Your program must output a file that will be read in Cytoscape for network visualization <https://cytoscape.org>. The supported file formats to Cytoscape can be found here: http://manual.cytoscape.org/en/stable/Supported\_Network\_File\_Formats.html

In your written report, you must have descriptions for the following sections: Motivating Problem from domain, Computational Problem formulation, Specific Approach to the problem (i.e. choice of algorithm), Specific Implementation of approach. Here’s an example of how this can be done for a different (but similar) task:

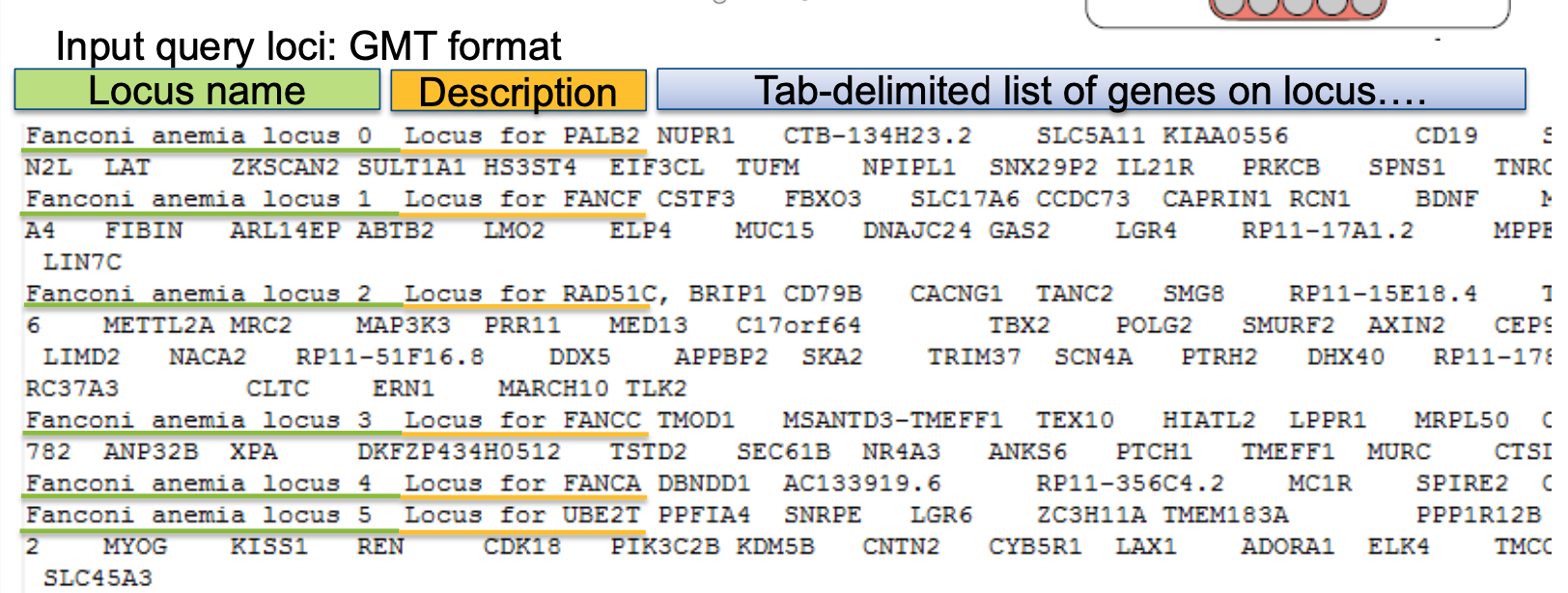
Motivating Problem from domain  
Fanconi anemia (FA) is a disease caused by genomic instability, with clinical features of developmental abnormalities in major organ systems, early-onset bone marrow failure, and a high predisposition to cancer. Studies have found 12 genes associated with FA, many of them involved in the FA/BRCA pathway. A functional network of all FA genes may reveal new mechanisms and pathways.

Computational Problem formulation  
Given a network and a set of query nodes, visualize subnetwork of all nodes and edges joining query nodes.

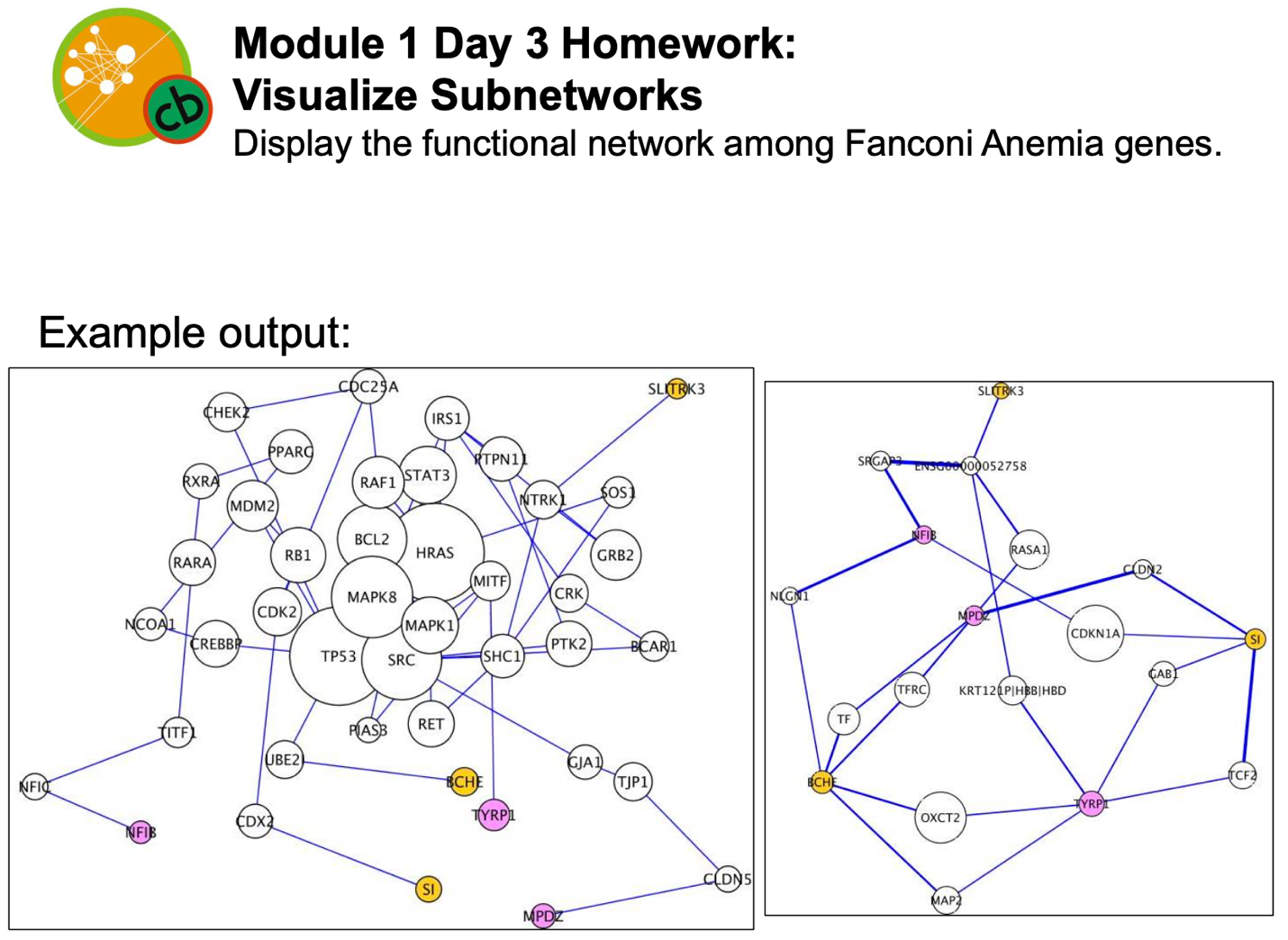
Specific Approach to the problem (i.e. choice of algorithm)  
Given a network of functional linkages and a set of disease genes (potentially with annotations), implement shortest path algorithm between query nodes and visualize all nodes and edges along the union of all pairwise shortest paths between query nodes.

Specific Implementation of approach  
Given the STRING network of functional linkages and the set of FA disease genes from OMIM, implement Dijkstra’s shortest path algorithm using priority queues and visualize all nodes and edges among all pairwise shortest paths between query nodes using Cytoscape, where non-query nodes are white, the original query nodes are each colored by a different color. The size of each node is proportional to its original degree in the network and the width of each edge is proportional to the number of shortest paths between pairs of query nodes in which it participates.

The input GMT file will be specified in the following format:



In this format, the list of FA genes can be found in the second column. In the STRING database of known and predicted protein-protein interactions, the first two columns represent interacting proteins and the third column represents the strength of their functional similarity.

Here are example outputs to your program: