CMPSC 465 Data Structures & Algorithms
Fall 2021 Antonio Blanca and David Koslicki

HW 5

Due October 5, 10:00 pm

Instructions: You are encouraged to solve the problem sets on your own, or in groups of three to five people, but you must write your solutions strictly by yourself. You must explicitly acknowledge in your write-up all your collaborators, as well as any books, papers, web pages, etc. you got ideas from.

Formatting: Each part of each problem should begin on a new page. Each page should be clearly labeled with the problem number and the problem part. The pages of your homework submissions must be in order. When submitting in Gradescope, make sure that you assign pages to problems from the rubric. You risk receiving no credit for it if you do not adhere to these guidelines.

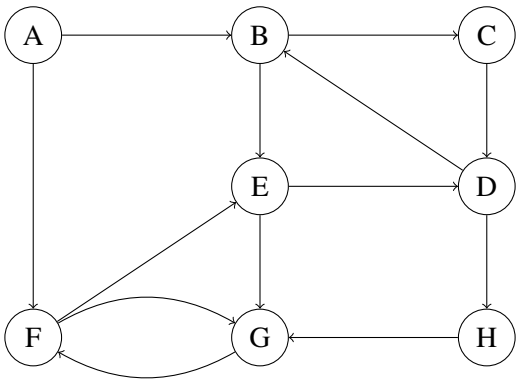
Late homework will not be accepted. Please, do not ask for extensions since we will provide solutions shortly after the due date. Remember that we will drop your lowest two scores.

This homework is due Monday, October 5, at 10:00 pm electronically. You need to submit it via Gradescope (Class code 6PERPR). Please ask on Piazza about any details concerning Gradescope.

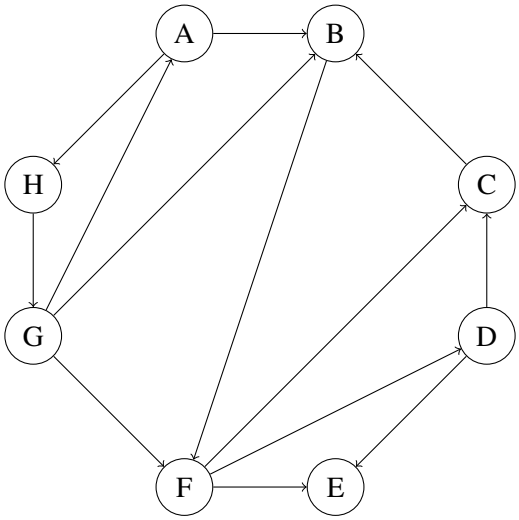
Extra credit. Five extra credits will be added to your homework score this time if you use Latex to typeset your solutions.

1. (20 pts.) Perform depth-first search on each of the following graphs; whenever there's a choice of vertices, pick the one that is alphabetically first. Classify each edge as a tree edge, forward edge, back edge, or cross edge, and give the pre and post number of each vertex.

(a)



(b)

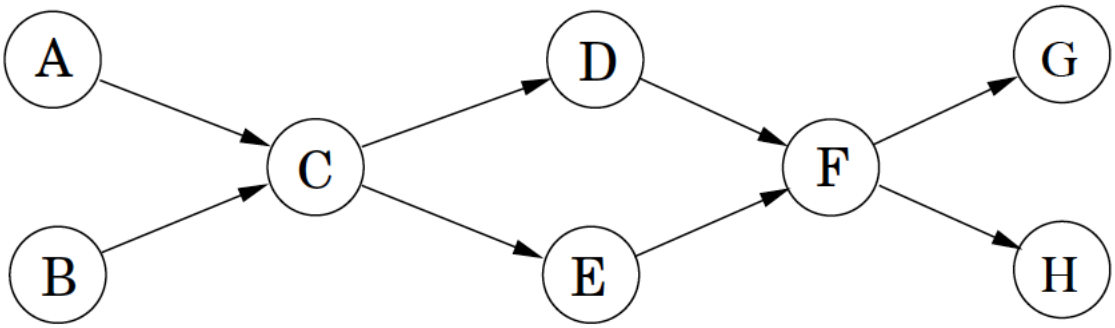


2. (20 pts.) Finding for cycles in undirected graphs

Design an algorithm to check whether an undirected graph $G = (V, E)$ has a cycle. Your algorithm should run in $O(|V|)$ time, independent of $|E|$. (Hint: Try to modify the Explore procedure and make sure you stop once you find one cycle.)

3. (20 pts.) Topological Sort

Run the DFS-based topological ordering algorithm on the following graph. Whenever you have a choice of vertices to explore, always pick the one that is alphabetically first.



(a) Indicate the *pre* and *post* numbers of the nodes. Please put your answers in the format of "$A : 1, 2$" for each vertex, with 1 as A's *pre* number you found and 2 as A's *post* number. Note this is only an example for the illustrative purpose that has nothing to do with the correct solution.

(b) What are the sources and sinks of the graphs? Please put your answers in the format of "Sources: ...; Sinks: ...".

(c) What topological ordering is found by the algorithm?

(d) How many topological orderings does this graph have? Why?

4. (20 pts.) Funny Money

You are in charge of the United States Mint. The money-printing machine has developed a strange bug: it will only print a bill if you give it one first. If you give it a d-dollar bill, it is only willing to print bills of value $d^2 \pmod{600}$ or $d^2+1 \pmod{600}$. For example, if you give it a \$5 bill, it is willing to print \$25 and \$26 bills, and if you then give it a \$26-dollar bill, it is willing to print bills of value \$76 or \$77. ($76 \equiv 26^2 \pmod{600}$.)

You start out with only a \$1 bill to give the machine. Every time the machine prints a bill, you are allowed to give that bill back to the machine, and it will print new bills according to the rule described above. You want to know if there is a sequence of actions that will allow you to print a \$10 bill, starting from your \$1 bill.

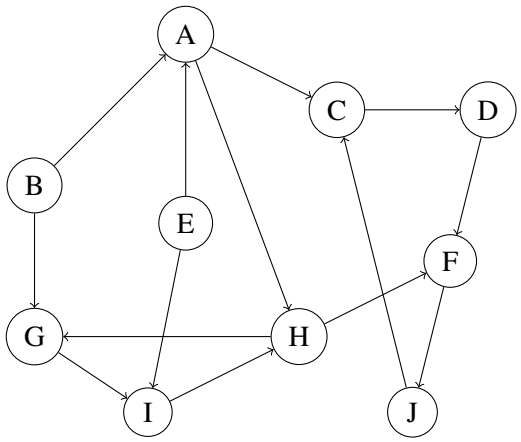
(a) Model this as a graph problem: give a precise definition of the graph involved and state the specific question about this graph that needs to be answered.

(b) What algorithm should be applied to solve the problem?

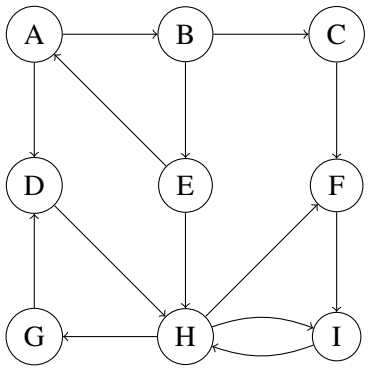
5. (20 pts.) Strongly Connected Components (SCC)

Run the strongly connected components algorithm on the following directed graphs. When doing DFS on the reverse graph G^R: whenever there is a choice of vertices to explore, always pick the one that is alphabetically first.

(i)



(ii)



In each case answer the following questions.

(a) In what order are the strongly connected components (SCCs) found?

(b) Which are source SCCs and which are sink SCCs?

(c) Draw the "metagraph" (each "meta-node" is an SCC of G).

(d) What is the minimum number of edges you must add to this graph to make it strongly connected?