Name KEY

Quiz - Module 9: Max-Flow and Min-Cut

1. [7 points] Answer the following True/False.

The *flow-value lemma* states that *the value of any valid flow* in a graph can be defined terms of *the capacity* of any cut. **True** **False**

The main loop in the *Ford-Fulkerson* algorithm may execute more often than is really needed to find the max-flow if it happens to find augmenting paths that only account for a small part of the total max-flow. **True** **False**

When the *Ford-Fulkerson* algorithm completes, each back-flow edge from v back to u in the residual graph G_f represent the final flow values for edge (u, v) in the flow-graph G. **True** **False**

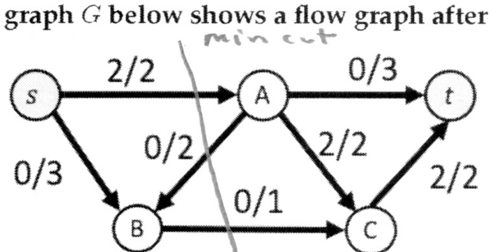
Any two proper disjoint subsets of the vertices (such that every node is present in one of the two subsets) in a flow-graph represent a valid *cut* of that flow-graph. **True** **False**

In a standard network flow-graph, under certain conditions a vertex v that is not the source or the sink can have total in-flow that has a different value than its total out-flow. **True** **False**

In *Ford-Fulkerson*, for a pair of vertices u and v connected in the residual capacity graph G_f, the sum of the values for the back-flow and the residual capacity edges between that vertex-pair must always equal the capacity of the edge between them in the flow-graph G. **True** **False**

When *Ford-Fulkerson* finds an augmenting path that includes a back-flow edge from v back to u in the residual graph G_f, this indicates *undoing* flow assigned earlier from v to u for an edge (u, v) in the flow-graph G. **True** **False**

The graph G below shows a flow graph after Ford-Fulkerson has found one augmenting path, $s - A - C - t$.



2. [2 points] Are there any augmenting paths in the residual flow graph G_f corresponding to the drawing of G shown above? If so, list the vertices in a valid augmenting path in the box below. If there is not, write "none."

Vertices in path: s B C A t

Flow of 1, uses back flow edge C → A

3. [2 points] Look for or find a min-cut in the graph G shown above and list the vertices on the source side of this cut in the box. (Include vertex s in your answer.) s B

4. [1 points] Which is the best explanation of why the time-complexity of *Ford-Fulkerson* is $\Theta(|E| \times |f|)$?

- To find how $|f|$ units of flow can move through the flow-graph G, at each step we must examine the current flow values for each edge in G.
- It costs $\Theta(|E|)$ to find an augmenting path, and in the worst case finding one path will add one unit of flow to the solution.
- We must update values for every edge in G_f every time we find an augmenting path.

5. [4 points] Below is an implementation of ford-fulkerson using residual graph G_f. Fill in (or complete) the blank lines to complete the implementation. *Note: This is python-esque but not exactly.*

```
//Given residual graph Gf, return max-flow
Max-Flow(Gf):
        flow = 0

        /* Loop while an augmenting path still found */
        while augPath=DFS(Gf) not null:
                minCost = infinity
                for e in augPath: //e=(e.u,e.v)
                        minCost = min(minCost, Gf[e.u][e.v])  //line 1

                for e in augPath:
                        Gf[e.u][e.v] += -= minCost  //line 2   decrease
                        Gf[e.v][e.u] = += minCost  //line 3   increase
                flow = flow + min Cost  //line 4

        return flow
```

This question is about the proof of the max-flow min-cut theorem. Below are the three short proofs from the theorem, but one has an error. Read the three proofs and then answer the questions below.

- **If there is no augmenting path in G_f, then there exists a cut in G whose capacity equals f**: Assume there is no augmenting path in G_f. Run DFS to find the nodes reachable from the source and put these in A (source side of the cut) and everything else in B (sink side of the cut). This cut has a capacity of f because DFS could not get across it (it is at capacity) and the flow going across the cut is f (flow-value lemma).
- **If there exists a cut (C) in G whose capacity equals f, then f is maximum flow**: Assume cut C exists, but f is not maximum. This means the flow can increase by at least 1. Via the flow-value lemma, we know the flow across the C will decrease, which cannot happen when the overall flow increased (contradiction!).
- **If the flow f is maximum flow, then there is no augmenting path in G_f**: Assume f is maximum but there IS an augmenting path. We can use this augmenting path to increase the flow by at least one unit, contradicting the claim that f was maximum.

6. [1 points] Circle the proof above that is NOT a valid proof.

7. [2 points] In a couple of sentences, describe the error in the proof you circled.

In the 3rd sentence, it's wrong to say the flow will decrease here. It would have to increase, but that violates what we know about capacity. (See slides or text on this!)

Name Key

Quiz - Module 10: Reductions

1. [8 points] Answer the following True/False.

yes!

When defining a reduction $A \leq_P B$, we must define 3 algorithms: one that transforms an input for A to an input for B; $\boxed{\text{a second that solves } B}$; and a third that transforms the solution for B to a solution for A. — True / **False**

no! not part of reduction!

yes!

In our reduction of an instance G of the *vertex-disjoint path* problem to an instance G' for *edge-disjoint path* problem, it's impossible for a pair of paths to share a vertex in G unless two paths share an edge in G'. — **True** / False

If we prove two reductions, $A \leq_P B$ and $B \leq_P C$, and each reduction is polynomial, then we can solve A indirectly in the time required to do the two reductions plus the time required to use an algorithm to directly solve C. — **True** / False

In a circulation network, there is not a single source node or a single sink node, but every node has the potential to have an positive overall outflow or to have a positive overall inflow. — **True** / False

In our reduction from a circulation network to max-flow, the circulation is feasible if and only if the maxflow value $|f|$ is equal to the sum of the positive demand nodes in the circulation network. — **True** / False

In *maximum bipartite matching* to find the edges in the set of "matching" edges M, we can reduce the problem to network-flow, solve for max-flow, and then count the number of edges from a node in L to a node in R that have flow value equal to $|f|$. — True / **False**

In the problem where we modeled an airplane scheduling problem, each flight in the schedule was modeled using two nodes connected by an edge with a lower-bound equal to the flight's duration. — True / **False**

In our example illustrating a perfect bipartite matching, every person was matched with exactly one adorable doggie. — **True** / False

2. [1 points] In a circulation network G with lower-bounds, if there is a directed edge (u, v) and it has a lower-bound l, which best describes how we transform this to a circulation network G' that does not have lower bounds? (Circle the bullet-symbol for your choice.)

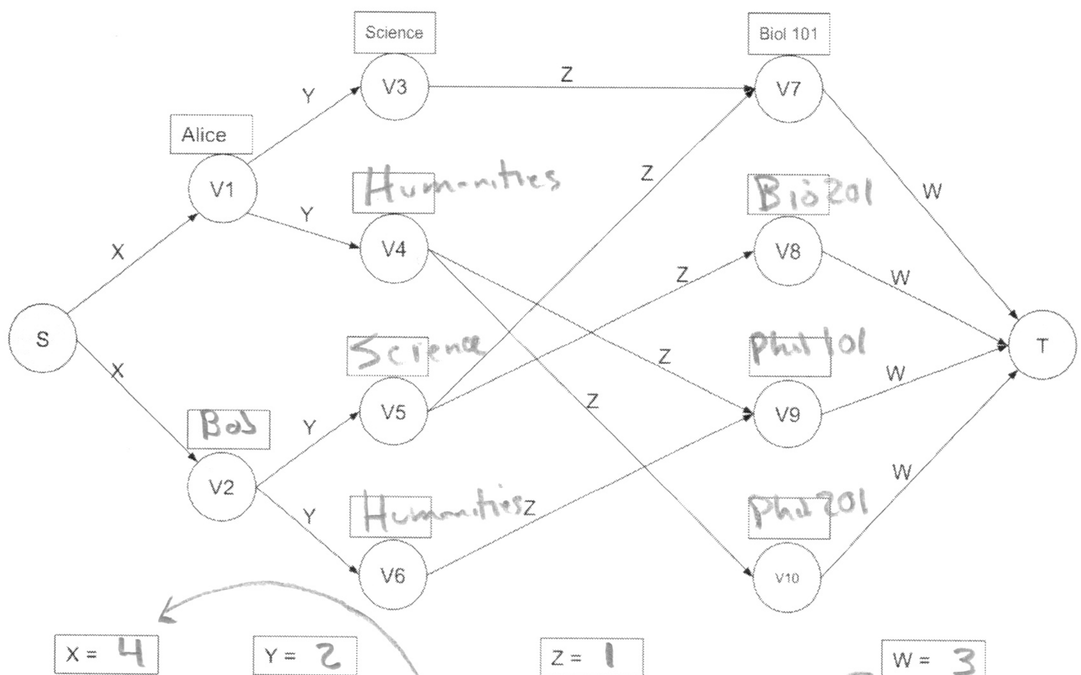
- The demand for v is increased by l, the demand for u is decreased by l, and the capacity of the edge is decreased by l.
- The demand for v is decreased by l, the demand for u is increased by l, and the capacity of the edge is decreased by l.
- The demands for v and u are changed in some way but the capacity of the edge does not change.

3. [2 points] Suppose we have problem A. Floryan thinks $\Theta(2^n)$ time is optimal, but Horton finds a reduction to problem B (which is solvable in $\Theta(n^3)$ time). If Horton's reduction takes $\Theta(n^5)$ time to convert problem A into problem B, what does this say about Floryan's claim and why?

Floryan is wrong. (A rare occurence!)
Using the reduction and the direct solution, we can solve A in $\Theta(n^5 + n^3)$, which is $o(2^n)$

In this problem, we will ask you to do a reduction similar to the one on homework 10 (scheduling). We are given a list of students and classes they are requesting. Each class has a capacity and each student must sign up for some number of courses. However, this time each student needs to sign up for a set number of science courses and humanities courses. For example, perhaps each student must take exactly 2 science courses and 1 humanities course.

The flow network below is the beginning of a reduction to max-flow. The questions below will ask you to label the graph to finish the reduction and answer a question about the reduction.



4. [3.5 points] Suppose you have this input: *The classes are Bio101, Bio201, Phil101, and Phil201 (Bio classes are all science and Phil are humanities). The students are Alice and Bob. Every class has a capacity of 3, and each student MUST take exactly 2 science and 2 humanities (4 courses total).* *Alice is interested in Bio101, Phil101, and Phil201. Bob is interested in Bio101, Bio201, and Phil101.* Given this input, label each node with what it represents. Some of these labels have been given to you to get you started. (Note: this example may not be satisfiable.)

5. [2 points] Additionally, assign a value for each edge (by filling in each box at the bottom of the graph) with the correct capacity that can be used to model this problem.

6. [2 points] In a sentence or two, explain how to take the max-flow of the network above to figure out how to enroll the students. Specifically, which edges are of particular interest here?

[IF $|f| = 4 \times$ number of students, all students can be enrolled as required.]

The edges labeled Z tell us which courses each student is enrolled in — look for Z edges with Flow == 1.