CS 530: Algorithms --- Steve Homer--- Fall 2017

Homework 1 --- Due Tuesday, September 19

Reading : 1. Chapter 26, pages 708-736

2. Chapters 1-4, look over and read whatever seems new to you.

Problems:

1. (12 points). Briefly explain your reasoning in (i) - (iv).

(i) Is 2^{n+2} in $O(2^n)$?

(ii) Is 2^{2n} in $O(2^n)$?

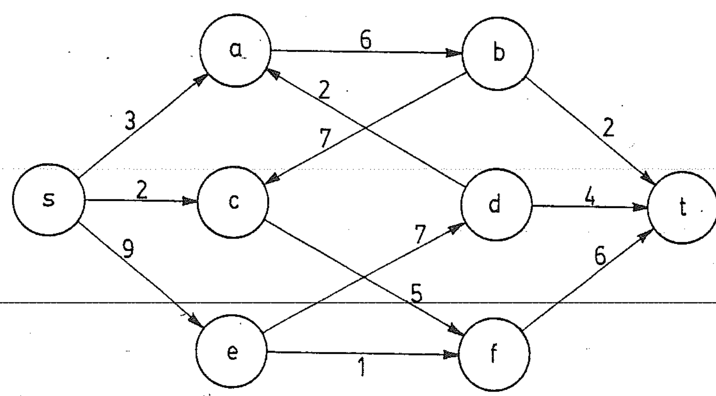
(iii) Is $(2^n)^2$ in $O(2^n)$?

(iv). Is n! in $O(2^n)$?

2. (6 points) Page 62, #3-4 parts e, and g. Justify your answers here.

3. Use the Ford Fulkerson algorithm to find the the maximum flow in the following flow graph.

Find the min s-t cut of the graph as well.



4. (5 points) Suppose you are given a flow graph G and also given a max flow f_{max} for G.

Show that you can then find a sequence of at most |E| augmenting paths which will result in the flow f_{max}.

5. (10 points)

(i). Recall the min cut problem for undirected graphs. Give an example of a graph G whose is min cut is not any single vertex of the graph. That is, for any vertex v in G the cut consisting of {v} on one side and all other vertices on the other is not a min cut.

(ii). Assume G is a 10 node graph and that G has a min cut of cost 5. Explain why G must have at least 25 edges.

(iii). Assume H is a 10 node graph which is a a simple cycle. That is H is connected and every node in H has exactly 2 neighbors. Clearly the size of H's min cut is 2. How many different min cuts does H have ? Explain your reasoning.

6. (10 points) A **centroid** of a tree with n vertices is a vertex such that its deletion leaves no subtree with more than n/2 vertices.

Is the centroid of a graph always a unique vertex ? Explain.

Write an algorithm to find a centroid of a tree.

Show how your algorithm works on the graph below. What is the complexity of your algorithm ?

You should be able to get a linear solution although partial credit will be given for a nonlinear algorithm.

Note: you are not required to prove that your algorithm is correct.

Picture: Vertex c is the centroid of graph G. It is unique in this case.

```
Graph G:
                  i ---g--h
                          |
a --- b ----- c ----- d
|\            |\
| \           | k---l
e   f         j
```