Fall 2014, Final Exam, Introduction to Algorithms

Name:
Section:
Email id:

11th December, 2014

This is an open book, open notebook exam.
Answer all ten questions. Each Question is worth 10 points. You have 180 minutes to complete the exam.
Happy Holidays and Have a Great New Year.

Sample Solution

1. **NP-complete and Greedy Algorithm**

   (a) Show the following problem is NP-complete by giving a polynomial time reduction from an already known NP-complete problem (see the hint). We know the problem is in NP (you do not have to give this). All you need to give is to give a reduction and show it works.

   Problem: Given an undirected graph, and integer k, testing whether G has a spanning tree such that the degree of each node is atmost k.[**5 Points**]

   Hint: Think of generlizing from a Rudrata or Hamiltionian path (Known to be NP-complete)

   Rudrata ~~cycle~~ path is polynomial time reducible to K-degree Spanning tree.

   Instance to Rudrata Path : $G(V, E)$

   ? : Does G have a rudrata path

   K-degree Spanning tree instance $G = (V, E)$, $k = 2$.

   K-degree Spanning tree has a solution iff Rudrata Path has a solutions.

   (b) Prove that vertex coloring problem can be solved in linear time using greedy coloring (with DFS), for connected regular graph of degree 2 (the degree of every vertex is 2.)

   Connected regular graph of degree 2 is a cycle (Cycle could be of even length or odd length). [**5 points**]

   1. Count the number of vertices of G
   2. do a DFS (G)
   3. If the number of vertices is odd

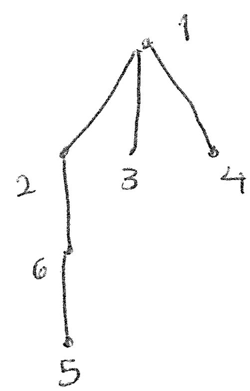
      Color(v) = pre number (v) % 3

      else Color(v) = prenumber(v) % 2

   odd cycle is 3 colorable and even cycle is 2 colorable.

2. **Approximation Algorithm** You are given a complete graph with 6 vertices and 15 edges. The 5 edges $(1,2), (1,3), (1,4), (2,6), (5,6)$ weights are 15 and the rest of them are of weight 20. Construct a 2 approximation Traveling salesman route for this graph and what is its cost. (The edge weights satisfy triangle inequality).[**10 points**]

MS



Cost is 75

TSR

1_2_6_5_3_4_1

1-2-6-5-3-4-1

Cost is 100

$4 \times 15 + 2 * 20 = 100$

3. **Graphs/DFS/Path**

(a) Consider an unweighted connected graph G. Describe a linear time algorithm to test whether the given node u is a cut node or not or not.(A node,u, is a cut node if we delete that node, the graph becomes disconnected.).[**5 points**]

H= delete (u) from G

do a DFS on ~~G~~ H
(connected)

~~if~~
if H ~~has~~ is connected then u is not a cut node
else it is a cutnode.

(b) Given a directed acyclic graph G, describe a linear time algotithm to determine whether there exists a vertex which can be reached by every other vertex. [**5 points**]

(Assume G is ~~connected (contracted)~~ ~~and~~ acylic - given)
Check if G contains more than one sink node (out degree 0).

If it contains more than one sink node then there is no vertex which can be reached from every other vertex

else the vertex of outdegree 0 is a sink node and can be reached from every other vertex.

4. **Linear Programming Formulation**

There are 2 factories which distributes myPhones to 3 stores. Every week each factory produces at most 70 myPhones and each store needs at least 40. Create a linear program which assigns how many myPhones each factory will ship to each store for a week. The distribution costs are summarized in the following table. Your objective is to minimize the distribution cost incurred by the company. Only formulate the linear program. **Do not solve it.**

Table 1: MyPhones Distribution Costs

Plant	Dist. Center #1	Dist. Center#2	Dist. Center #3
A	\$4	\$6	\$3
B	\$6	\$8	\$2

[**10 points**]

Let x_j be the number of phone from plant A to dist. center j

Let y_j be the number of phones from plant B to dist. center j

$$\min \quad 4x_1 + 6x_2 + 3x_3 + 6y_1 + 8y_2 + 2y_3$$

s.t.

$$\sum x_j \leq 70$$

$$\sum y_j \leq 70$$

$$x_1 + y_1 \leq 40$$

$$x_2 + y_2 \leq 40$$

$$x_3 + y_3 \leq 40$$

$$x_1, x_2, x_3 \geq 0$$

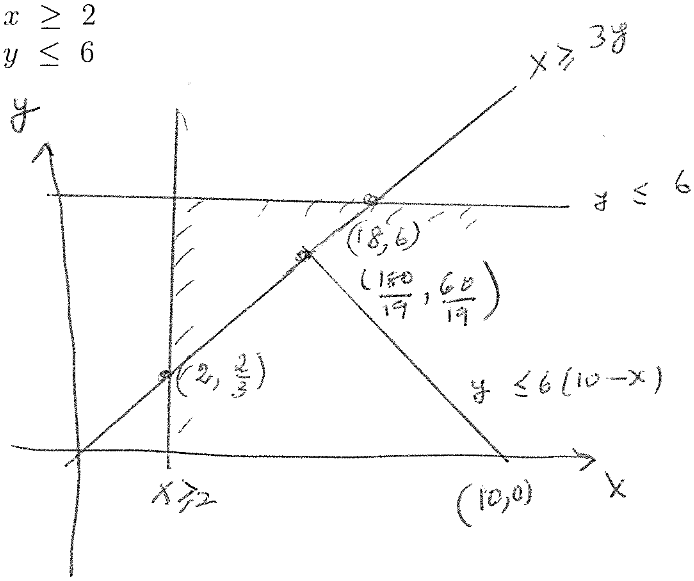
$$y_1, y_2, y_3 \geq 0$$

5. **Linear programming Geometric Solution** Solve the following linear programming Problem: (Use a geometric approach) [**10 points**]

maximize $10x + 5y$

such that

$$
\begin{aligned}
x &\geq 0 \\
y &\geq 0 \\
x &\geq 3y \\
y &\leq 6(10 - x) \\
x &\geq 2 \\
y &\leq 6
\end{aligned}
$$

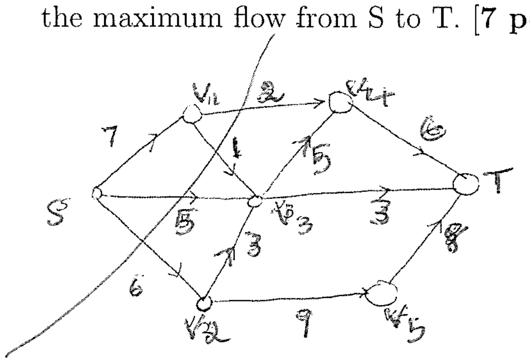


max value $\left(20,\ 20 + \frac{10}{3},\ \frac{1800 + 300}{19},\ 100\right)$

$= \frac{2100}{19}$

6. **Network Flows**

(a) Consider the following network (the numbers are edge capacities). What is the maximum flow from S to T. [**7 points**]



S - V1 - V4 - T   2

S - V3 - T   3

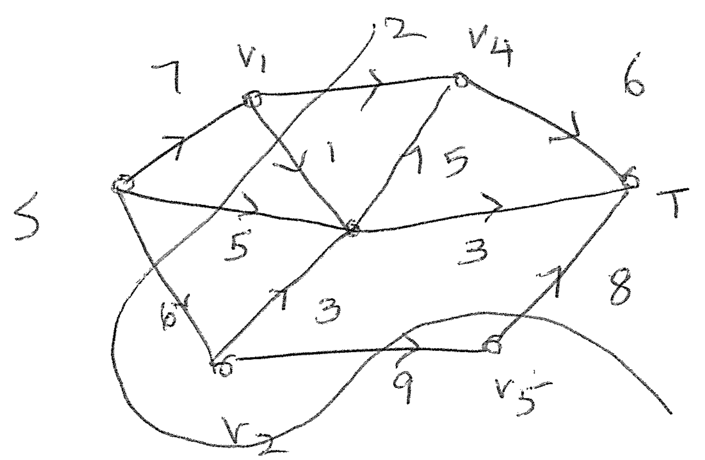
S - V1 - V3 - V4 - T   1

S - V3 - V4 - T   2

S - V2 - V5 - T   6

Max Flow is 14

(b) In the above graph, consider the cut edges $< v_1, v_4 >, < v_1, v_3 >, < S, v_3 >, < S, v_2 >, < v_2, v_5 >, < v_5, T >$ that separates S and T. What is the value of this cut? (Hint: only capacity from S to T contribute the cut value) [**3 Points**]



2 + 1 + 5 + 6 + 8

= 22

7. **Dynamic Programming** Given an unlimited supply of coins of denominations $x_1, x_2, \cdots, x_n$, we wish to make change for value v; that is, we wish to find a set of coins whose total value is v. Give an $O(nv)$ dynamic prgramming algorithm for the following problm.

   Input: $x_1, x_2, \cdots, x_n, v$.
   Question: Is it possible to make change for v using $x_1, x_2, \cdots, x_n$.

   You are allowed to use unlimited amount of coins of any denominations.

   Hint: First write the recursion and then give a pseudo code. [**10 points**]

```
c[0] = true
for i = 1 to v
  { c[i] = false }
    for i = 1 to v
    { for j = 1 to n
      { if x_j < i
            c[i] = c[i] v c[i-x_j]
      }
    }
```

$O(nv)$

8. **Extended Euclidean Algorithm** Solve the equation for integers x and y such that $21 \times x + 11 \times y = 1$ gcd(21,11)=1.[**10 points**]

21	11	1	10
11	10	1	1
10	1	10	0
1	0		

$$\begin{aligned}
1 &= 1 - 0 \\
&= 1 - (10 - 10 \cdot 1) \\
&= 11 \cdot 1 - 10 \\
&= 11 \cdot (11 - 1 \cdot 10) - 10 \\
&= 11 \cdot 11 - 12 \cdot 10 \\
&= 11 \cdot 11 - 12 \cdot (21 - 1 \cdot 11) \\
&= 23 \cdot 11 - 12 \cdot 21
\end{aligned}$$

$x = -12$

$y = 23$

9. **Algorithm Analysis** You are given an array A consisting of n integers $A[\mathbf{1}], A[\mathbf{2}], \cdots, A[\mathbf{n}]$. You would like to output a two dimensional n-by-n array B in which $B[\mathbf{i}][\mathbf{j}]$ (for $i \leq j$) contains the sum of entries $A[\mathbf{i}]$ through $A[\mathbf{j}]$ that is the sum $A[\mathbf{i}] + A[\mathbf{i}+\mathbf{1}] + \cdots + A[\mathbf{j}]$, (The values of array entry $B[\mathbf{i}][\mathbf{j}]$is left unspecified for $i > j$, so it does not matter what is the output of those values). Design the most efficient algorithm to output this matrix and Analyze the algorithm. (An inefficient correct algorithm will be worth half the points.) [**10 points**]

```
for i = 1 to n
{B[i][i] = A[i]}
for i = 1 to n-1
{ for j = i+1 to n
  {B[i][j] = B[i][j-1] + A[j]
  }
}
```

Complexity $O(n^2)$

10. **Divide and Conquer** Find the value of T(8) for the following recurrence equation. Solve it for general $n = 2^k$. That is a general solution in big Oh notation is needed.

$$T(n) \;=\; T(n/2) \;+\; 5$$

and

$$T(1) = 2$$

. **[10 points]**

$T(1) = 2$

$T(2) = 7$

$T(4) = 12$

$T(8) = 17$

$T(2^K) = T(2^{K-1}) + 5$

$Q(K) = Q(K-1) + 5$

$\vdots$

$Q(1) = Q(0) + 5$

$Q(0) = 2$

$Q(K) = 5 \cdot K + 2$

$T(2^K = n) = 5 \cdot (\log_2 n) + 2$

$= O(\log n)$