Exam 2
Computer Engineering 251
Santa Clara University
Thursday, 23 April 1998

NAME (Printed) ______________________________

NAME (Signed) ______________________________

Login ______________________________

Day of Lab Session ______________________________

Please show all your work and circle your answers. Your grade will be based on the the work shown.

Question 1 (10 points)	
Question 2 (10 points)	
Question 3 (10 points)	
Question 4 (10 points)	
Question 5 (10 points)	
Question 6 (10 points)	
Question 7 (20 points)	
Question 8 (20 points)	
TOTAL	

Useful Formulas

$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$ $\sum_{i=1}^{n} x^i = \frac{x^{n+1}-1}{x-1}$

$\sum_{i=1}^{n} \frac{1}{i} = \ln n + O(1)$ $\sum_{i=0}^{\infty} ix^i = \frac{x}{(1-x)^2}$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots$ $\lim_{n\to\infty}(1 + \frac{x}{n})^n = e^x$

$n! = \sqrt{2\pi n}(\frac{n}{e})^n(1 + \Theta(\frac{1}{n}))$ $n! = o(n^n)$

$n! = \omega(2^n)$ $\lg(n!) = \Theta(n \lg n)$

$\binom{n}{k} = \frac{n!}{k!(n-k)!}$ $(\frac{n}{k})^k \leq \binom{n}{k} \leq (\frac{en}{k})^k$

$E[X] = \sum_x xPr[X = x]$ $Var[X] = E[(X - E[X])^2] = E[X^2] - E^2[X]$

–Put gates/formulas here–

1. (10 points) 5 short answers (morning) or 10 T/F (evening)

   (a) ___ In a flow network, the flow on an edge is always larger than the capacity of the edge.

   (b) ___ The longest simple path in an undirected graph has at most $V - 1$ edges.

   (c) ___ Greedy algorithms always find the optimal solution to a problem.

   (d) ___ Every logical gate has fan-in 2 (that is, has exactly 2 inputs).

   (e) ___ Prim's Algorithm and Kruskal's Algorithm solve the same problem.

   (f) How many different minimal spanning trees are there on a triangle with edge weights 1 (that is, the graph $G = (\{1,2,3\}, \{(1,2),(1,3),(2,3)\}$ and $w(1,3) = w(1,3) = w(2,3) = 1$)?

   ____

   (g) What is the most number of edges possible in a matching on a graph $G = (V, E)$, where the number of vertices is even?

   ____

   (h) In a flow network, what is the capacity of a path, p, if the capacity of each edge (u, v) is $c(u, v)$?

   ____

   (i) How can the number of strongly connected components in a graph change if a new edge is added?

   ____

   (j) If an undirected graph has E edges and V vertics, how many edges does a spanning tree of the graph have?

   ____

2. (10 points) Huffman encoding

   (a) Given the following character set and the frequency each letter occurs, construct a Huffman code for the character set:

	a	b	c	d	e	f
Frequency (in thousands)	45	13	12	16	9	5

   (b) Given the following character set and the frequency each letter occurs, construct a Huffman code for the character set:

	a	b	c	d	e	f
Frequency (in thousands)	5	11	1	4	9	1

3. (10 points) Give algorithm and example
   (a) (morning)
      i. Write Dijkstra's Algorithm.
      ii. Give a simple example of a directed graph with negative-weight edges for which Dijkstra's Algorithm produces incorrect answers.
   (b) (evening)
      i. Write Kruskal's Algorithm
      ii. Suppose that all the edge weights in the graph are integers in the range from 1 to $|V|$. How fast can you make Kruskal's Algorithm run?
4. (10 points) Analyze graph algorithms
   (a) Analyze the following algorithm:

```
G is a directed graph with weight function w and s is a
vertex of G.

BELLMAN-FORD(G,w,s)
INITIALIZE-SINGLE-SOURCE(G,s)
for i <- 1 to |V[G]| - 1
   do for each edge (u,v) in E[G]
      do RELAX(u,v,w)
for each edge (u,v) in E[G]
   do if d[v] > d[u] + w(u,v)
      then return FALSE
return TRUE

INITIALIZE-SINGLE-SOURCE(G,s)
for each vertex v in V[G]
   do d[v] <- infinity
      p[v] <- NIL
d[s] <- 0

RELAX(u,v,w)
if d[v] > d[u] + w(u,v)
  then d[v] <- d[u] + w(u,v)
       p[v] <- u
```

   (b) Analyze the following algorithm. You may assume that all flows are integer-valued (that is, f[u,v] is always an integer) and $|f*|$ is the value of the maximal flow.

```
G is a directed graph with s and t vertices in G.
(Note: Due to typesetting

FORD-FULKERSON(G,s,t)
for each edge (u,v) in E[G]
   do f[u,v] <- 0
      f[v,u] <- 0
while there exists a path p from s to t in the
      residual network Gf
   do cf(p) <- min{cf(u,v) : (u,v) is in p}
      for each edge (u,v) in p
         do f[u,v] <- f[u,v] + cf(p)
            f[v,u] <- - f[u,v]
```

5. (10 points) Simple circuit question

   (a) i. Draw a circuit that takes **3 bit inputs** x, y, z and return 1 if their sum is odd and 0 if their sum is even.

   ii. Draw a circuit that takes **4 bit inputs** x, y, z, w and return 1 if their sum is odd and 0 if their sum is even.

   (b) i. Draw a circuit that takes **3 bit inputs** x, y, z. The circuit should have 2 outputs: c and s. The output c should be 1 iff at least 2 of the inputs are 1. The output s should be 1 if the sum of the inputs is odd and 0 if sum of the inputs is even.

   ii. Use your circuit from above to design a circuit that will add two 4-digit binary numbers. (You may represent the circuit from above as a "black box" instead of redrawing it for this part of the question.)

6. (10 points)

   (a) Draw an example of flow network with a flow that is not maximal. Label the edges of your graph with the flow and capacity of the edge. Also indicate an augmenting path for your flow.

   (b) Show that for any graph $G = (V, E)$, there exists a minimal spanning tree that contains the minimal weight edge.

7. (20 points) Solve recurrences using generating functions

   (a) Solve the following recurrence:

$$T_{n+2} = 2T_{n+1} - T_n \text{ for } n \geq 0,\ T_0 = 0, T_1 = 1$$

   (Hint: let $\mathcal{T}(x) = \sum_n T_n x^n$, solve for T(x), and use formulas about sums to decide the coeffients).

(b) Solve the following recurrence:

$$T_n = \sum_{k=0}^{n} T_k \text{ for } n \geq 1,\ T_0 = 1$$

(Hint: let $\mathcal{T}(x) = \sum_n T_n x^n$, solve for T(x), and use formulas about sums to decide the coeffients).

8. (20 points) Design an algorithm...

(a) (20 points) Design a greedy algorithm for storing web pages in memory. The total size of memory is M. As each page arrives, you should place it memory if space is available. If space is not available, you should remove another page (or pages) from memory and insert your page. The pages will be specified by their size only. For example, if $M = 10$ and the sequence of page sizes is $1, 5, 8, 1$, you would insert the first 2 pages, then need to remove at least the second page to insert the third page. After that you, you can insert the fourth page.

Please specify the data structures you used and give an analysis of the program.

(b) (20 points) **Arbitrage** is the use of discrepencies in currency exchange rates to transform one unit of currency into more than one unit of the same currency. For example, suppose that 1 US dollar buys 0.7 British pounds, 1 British pound buys 9.5 French francs, and 1 French franc buys 0.16 dollars. By converting currencies, a trader can start with 1 US dollar and end up with $1 \times 0.7 \times 9.5 \times 0.16 = 1.064$ dollars, and then returning a profit of 6.4%.

Suppose that we are given n currencies $c_1, c_2, \ldots, c_n$ and an $n \times n$ table R of exchange rates such that one unit of c_i buys $R[i, j]$ units of currency c_j.

Give an efficient algorithm for deciding whether or not there exists a sequence of currencies that will yield a profit. Specify any data structures you use and give an analysis of the program.