CS 310 (sec 20) - Spring 2005 - Final Exam (solutions)

SOLUTIONS

1. (Functions) For each of the following functions determine whether it is one-to-one, onto or a bijection. If it is a bijection find its inverse.

1. $f : \mathbb{Z} \to \mathbb{Z}$, $f(x) = 2x + 1$.
2. $f : \mathbb{R} \to \mathbb{R}$, $f(x) = 2x + 1$.
3. $f : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$, $f(x, y) = 2x^2 + y$.
4. $f : \mathbb{Z} \times \mathbb{Z}^* \to \mathbb{Q}$, $f(x, y) = x/y$.
5. $f : \mathbb{N} \to \mathbb{N} \times \{0, 1, 2\}$, $f(x) = (\lfloor x/3 \rfloor, x - 3 \cdot \lfloor x/3 \rfloor)$, where $\lfloor x \rfloor =$ greatest integer less than or equal to x.

Solution:

1. One-to-one. It is not onto because the image contains only odd integers.
2. Bijection, $f^{-1}(x) = (x - 1)/2$.
3. Onto. It is not one-to-one because, e.g., $f(0, 2) = f(1, 0) = 2$.
4. Onto. It is not one-to-one because, e.g., $f(2, 3) = 2/3 = 4/6 = f(4, 6)$.
5. Bijection, $f^{-1}(x, y) = 3x + y$.

2. (Algorithms) Let $f(n)$ be the function defined with the following pseudocode:

```
procedure f(n)
  if n = 0 then
    return 1
  else
    return (n * f(n-1))
end f
```

1. Find the *exact value* of $f(n)$ for every integer $n \geq 0$.

2. Find the slowest growing function $g(n)$ among the following ones such that $f(n) = O(g(n))$:

$$1\,, \quad \log\log n\,, \quad \log n\,, \quad n\,, \quad n\log n\,, \quad n^k \ (k \in \mathbb{Z}^+,\ k \geq 2)\,, \quad 2^n\,, \quad n^n\,.$$

3. Among those same functions find the fastest growing one such that $f(n) = \Omega(g(n))$.

4. Is $f(n) = \Theta(g(n))$ for any of those functions? If yes, which one? If not, why not?

Solution:

1. $f(n) = n!$

The answers to the next two questions are based on the double inequality $2^n < n! < n^n$ for every $n \geq 4$.

2. $g(n) = n^n$, i.e., $f(n) = O(n^n)$.

3. $g(n) = 2^n$, i.e., $f(n) = \Omega(2^n)$.

4. No. Among the functions shown only n^n and 2^n are "candidates" for the $g(n)$ such that $f(n) = \Theta(g(n))$. We need $C_1\, g(n) \leq n! \leq C_2\, g(n)$ for some positive constants C_1, C_2, and for every n large enough. We already have $C_1\, 2^n \leq n! \leq C_2\, n^n$, but we cannot have $C_1\, n^n \leq n!$ or $n! \leq C_2\, 2^n$ because $n^n/n! \to \infty$ and $n!/2^n \to \infty$ as $n \to \infty$.

3. (Induction) For which positive integers n is $n < (n-1)!$? Prove your answer using mathematical induction.

Solution:

We can see by inspection that the claim is false for $n = 1, 2, 3$, but we will prove by induction that it is true for every $n \geq 4$.

1. *Basis Step*: For $n = 4$ we have $4 < 6 = 3!$, so the proposition is true for $n = 4$.
2. *Inductive Step*: We must prove that for any $n \geq 4$,

$$n < (n-1)! \Rightarrow n + 1 < n!$$

So, assume that $n \geq 4$, and $n < (n-1)!$ Then for $n+1$ we have

$$n + 1 < n \cdot n \underset{\uparrow}{<} (n-1)! \cdot n = n!$$

(induction hypothesis)

This completes the Inductive Step.

Hence the claim is true for every integer $n \geq 4$.

4. (Graphs/Counting)

1. How many simple graphs are there with 10 vertices and 3 edges?
2. How many simple graphs are there with 10 vertices (and any number of edges)?
3. How many multigraphs are there with 10 vertices and 3 edges?
4. How many pseudographs are there with 10 vertices and 3 edges?

Assume the vertices are labeled $1, 2, 3, \ldots$, so for instance graph G_1 with set of edges $E_1 = \{(1,2), (2,3), (3,1)\}$, and graph G_2 with set of edges $E_2 = \{(2,3), (3,4), (4,2)\}$, are considered different even though they are isomorphic.

Recall that in a *simple graph* multiples edges and loops are not allowed. In a *multigraph* multiples edges are allowed, but loops are not. In a *pseudograph* both multiples edges and loops are allowed.

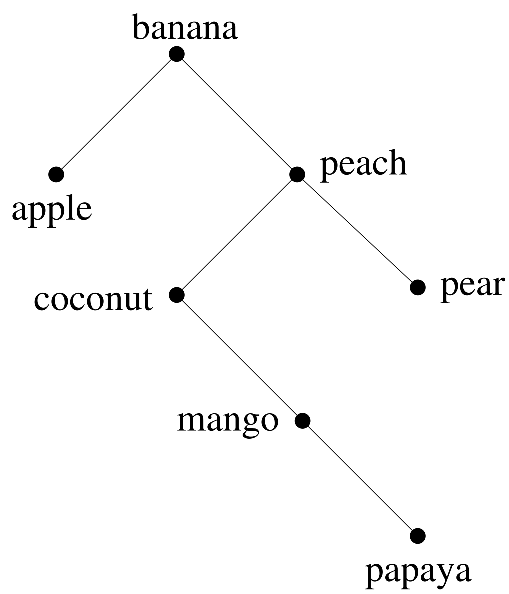
Solution:

The problem deals with different ways of choosing edges among all possible (unordered) pairs of 10 vertices. If the vertices must be different (no loops) then there are $\binom{10}{2} = 45$ such pairs. Otherwise (loops allowed) there are $\binom{10}{2} + 10$, or equivalently $\binom{11}{2} = 55$ such pairs. Also recall that he number of combinations of n objects taken r at a time with repetition is $\binom{n+r-1}{r}$.

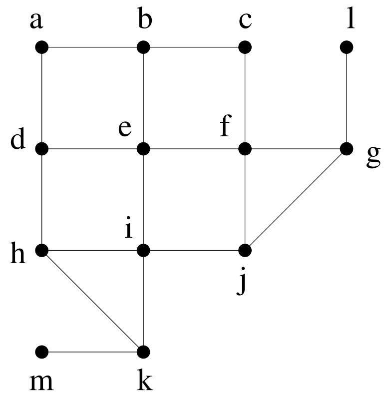
1. Number of simple graphs with 10 vertices and 3 edges $= \binom{\binom{10}{2}}{3} = \binom{45}{3} = 14190$.
2. Number of simple graphs with 10 vertices and any number of edges $= 2^{\binom{10}{2}} = 2^{45} = 35184372088832$.
3. Number of multigraphs with 10 vertices and 3 edges $= \binom{\binom{10}{2} + 3 - 1}{3} = \binom{47}{3} = 16215$.
4. Number of pseudographs with 10 vertices and 3 edges $= \binom{\binom{11}{2} + 3 - 1}{3} = \binom{57}{3} = 29260$.

5. (Search Trees) Build a binary search tree for the words *banana*, *peach*, *apple*, *pear*, *coconut*, *mango*, and *papaya* using alphabetical order.

Solution:

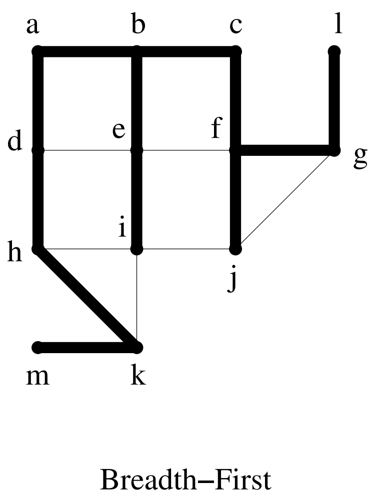


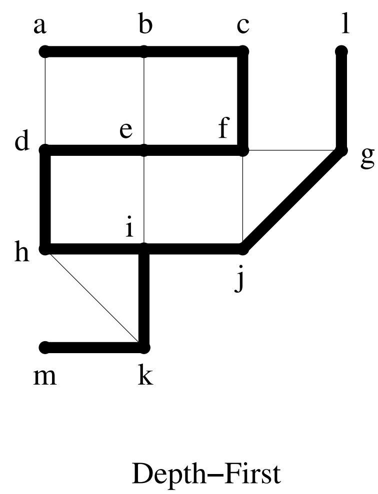
6. (Spanning Trees) Find the spanning trees of the following graph with its vertices sorted in alphabetical order:



1. Using Breadth-First Search.
2. Using Depth-First Search.

Solution:





7. (Boolean Algebras) Find Boolean expressions (as simple as possible) for the following Boolean functions with three arguments:

1. Majority Voting, i.e., $f(x, y, z) = 1$ if the algebraic sum $x + y + z \geq 2$, and $f(x, y, z) = 0$ otherwise.
2. Parity: $f(x, y, z) = 0$ if the algebraic sum $x + y + z$ is even, and $f(x, y, z) = 1$ if it is odd.
3. All or Nothing: $f(0, 0, 0) = f(1, 1, 1) = 1$, otherwise $f(x, y, z) = 0$.

Solution:

1. Majority Voting:

$$f(x, y, z) = x \cdot y + x \cdot z + y \cdot z$$

2. Parity:

$$\begin{aligned} f(x, y, z) &= x \cdot y \cdot z + x \cdot \overline{y} \cdot \overline{z} + \overline{x} \cdot y \cdot \overline{z} + \overline{x} \cdot \overline{y} \cdot z \\ &= x \oplus y \oplus z \end{aligned}$$

3. All or Nothing:

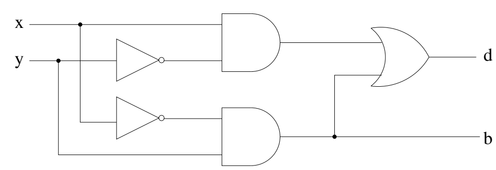
$$f(x, y, z) = x \cdot y \cdot z + \overline{x} \cdot \overline{y} \cdot \overline{z}$$

8. (Logic Gates) Construct a circuit for a half subtractor using AND gates, OR gates and NOT gates. A half subtractor has two bits x, y as input and produces as output a difference bit $d = x - y$ and a borrow b, according to the following table:

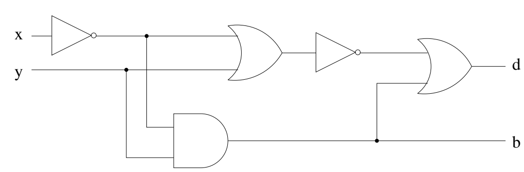
x	y	d	b
1	1	0	0
1	0	1	0
0	1	1	1
0	0	0	0

Solution:

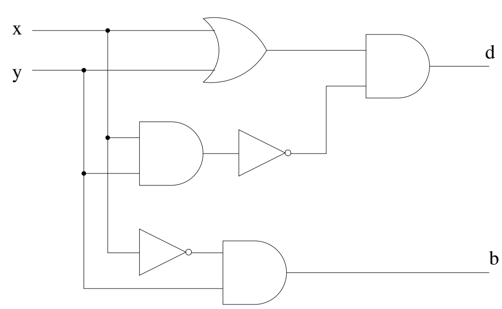
The difference bit is $d = x \oplus y = x \cdot \overline{y} + \overline{x} \cdot y$. The borrow is $b = \overline{x} \cdot y$.



This is another design similar to the previous one but with $x \cdot \overline{y}$ replaced with $\overline{\overline{x} + y}$.



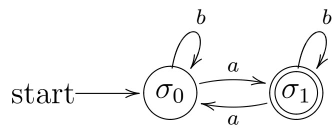
Another design, using $d = x \oplus y = (x + y) \cdot \overline{x \cdot y}$.



Other designs are possible.

9. (Languages and Automata)

1. Find the language recognized by the following automaton:



2. Design (give the transition diagram of) an automaton that recognizes the language over $\{a, b\}$ defined by the regular expression $a^* + b^*$, consisting of all strings (including the empty string) containing only a's, or only b's, but not both: $L = \{\lambda, a, b, aa, bb, aaa, bbb, \dots\}$.

Solution:

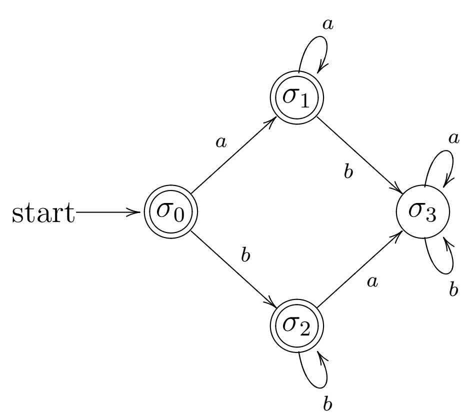
1. Possible answers:

   (a) Strings over $\{a, b\}$ with an odd number of a's.

   (b) Language defined by the following regular expression: $b^*ab^*(ab^*ab^*)^*$.

   (Other descriptions of the language are possible.)

2. The following automaton recognizes the desired language:



10. (Grammars) A palindrome is a string that reads the same forward and backward.

1. Define a grammar for all palindromes over $\{a, b\}$ with an even number of symbols (including the empty string), e.g., λ, 'aa', 'bb', '$aaaa$', '$abba$', '$baab$, '$bbbb$', '$aabbaa$', etc.

2. Define a grammar for all palindromes over $\{a, b\}$ with an odd number of symbols, e.g., 'a', 'b', 'aaa', 'aba', 'bab', 'bbb, '$aaaaa$', '$aabaa$', etc.

Solution:

1. $S \to aSa, \quad S \to bSb, \quad S \to \lambda.$

2. $S \to aSa, \quad S \to bSb, \quad S \to a, \quad S \to b.$