Student Name: __Solutions_____________ Student Number: ___________________________	CISC 203 Discrete Mathematics for Computing Science Test 5 Fall 2010 Professor Mary McCollam

Please write in pen and only in the box marked "Answer".
This is a closed-book exam. No computers or calculators are allowed.

Question 1: [10 marks]

(a) Show that the relation *R* on the set of all bit strings such that *s R t* if and only if *s* and *t* contain the same number of 1s is an equivalence relation.

Answer:

R is an equivalence relation because:

1. R is reflexive, since sRs for all bit strings s: s contains the same number of 1s as itself
2. R is symmetric, since if sRt then tRs: If a bit string s has the same number of 1s as a bit string t, then t has the same number of 1s as s
3. R is transitive, since if sRt and tRv, then sRv: If a bit string s has the same number of 1s as a bit string t and a bit string t has the same number of 1s as a bit string v, then s has the same number of 1s as v.

(b) List the ordered pairs in the equivalence relation on the set { 0, 1, 2, 3, 4 } formed by the partition { 0, 1, 2 }, { 3, 4 }.

Answer:

{ (0,0), (1,1), (2,2), (3,3), (4,4), (0,1), (1,0), (0,2), (2,0), (1,2), (2,1), (3,4), (4,3) }

Question 2: [10 marks]

(a) Assume S is the set of all people in the world. Consider (S, R), with $(a,b) \in R$, where *a* and *b* are people and either a = b or a is a descendant of b.

Answer:

i) Is (S, R) a poset? Why or why not?

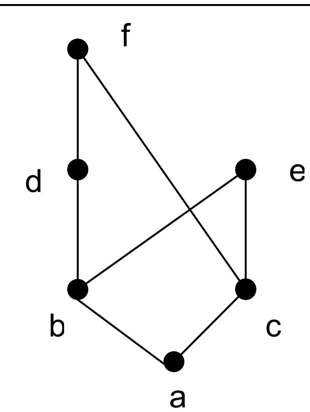
a = b or a is a descendant of b: This is a poset since:

1. R is reflexive, since every person is equal to themselves
2. R is antisymmetric, since if a is a descendant of b, then b is not a descendant of a, so aRb only if a = b
3. R is transitive, since if a is equal to or a descendant of b and b is equal to or a descendant of c, then a is equal to or a descendant of c.

ii) Is (S, R) a totally ordered set? Why or why not?

No, it is not a totally ordered set, since not every pair of elements is comparable, i.e., it's not always true that either a R b or b R a, for every a and b that are elements of S. For instance, you and I are not the same person and neither of us is a descendant of the other.

(b) In the poset represented by the following Hasse diagram, identify the:



Answer:

i) maximal and minimal elements
   maximal: f, e   minimal: a

ii) greatest element, if it exists: none

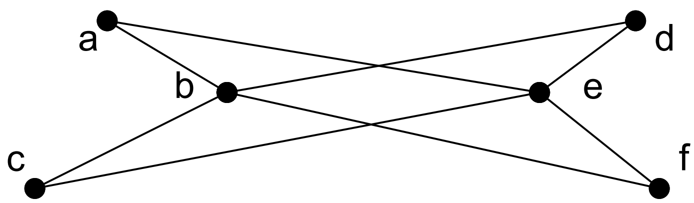
iii) least element, if it exists: a

iv) lower bounds of { b, f }: b, a

v) greatest lower bound of { b, f }, if it exists: b

Question 3: [10 marks]

(a) Determine whether the following graph is bipartite. Justify your answer.



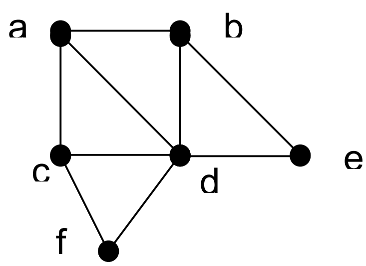
Answer: This graph is bipartite.

The vertices can be divided into two disjoint sets such that the vertices in each set are not adjacent to any other vertex in the same set:

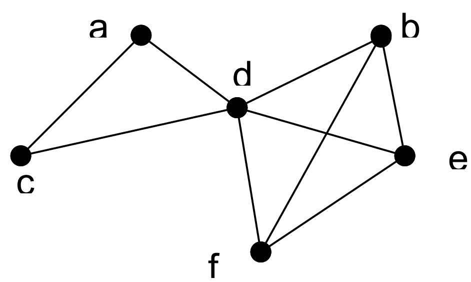
Vertices a, c, d, and f in one set
Vertices b and e in the other set

(b) Determine whether the following pair of graphs are isomorphic. Exhibit an isomorphism or provide a rigorous argument that none exists.

G



H



Answer: These two graphs are not isomorphic.

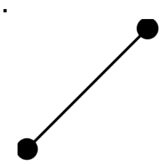
They have the same number of vertices and the same number of edges.

They also have the same number of vertices of each degree:

2 vertices of degree 2, 3 vertices of degree 3, and one vertex of degree 5.

However, if we check the sub-graphs consisting of all vertices of a certain degree and edges directly connecting those vertices, not all of those sub-graphs are isomorphic.

For instance, the sub-graph for vertices of degree two in the right graph is:



Whereas the two vertices of degree two in the left graph are unconnected.

Question 4: [10 marks] Use Huffman coding to encode these symbols with given frequencies: A: 0.10, B: 0.25, C: 0.05, D: 0.15, E: 0.30, F: 0.07, G: 0.08.

Answer:

In what follows, the heavier tree is always to the left (edges going left are labeled with a 0, while edges going right are labeled with a 1).

Combine F and C into a tree T_1 of weight 0.12.

Combine A and G into a tree T_2 of weight 0.18.

Combine D and T_1 into a tree T_3 of weight 0.27.

Combine B and T_2 into a tree T_4 of weight 0.43.

Combine E and T_3 into a tree T_5 of weight 0.57.

Finally, combine T_5 and T_4

This gives the following code: A: 110, B: 10 , C: 0111, D: 010, E: 00, F: 0110, G: 111