UNIVERSITY OF ALBERTA
CMPUT 204 (Session A2)

Term Test 3, November 2004

Duration: 50 min.

Student Number: |_|_|_|_|_|_|_|_|_|

Last Name: _____

First Name: _____

---

*Do **not** turn this page until you have received the signal to start.*
(In the meantime, please fill out the identification section above.)

Aids allowed: NONE

1: _____/15

2: _____/25

TOTAL: _____/40

Good Luck! PLEASE HAND IN

Question 1. [15 MARKS]

Consider the following undirected graph $G(V, E)$:

$$\begin{aligned} V &= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\} \\ E &= \{(1,5), (1,6), (1,9), (3,5), (3,9), (3,11), (4,8), (4,10), \\ &\quad (5,6), (6,11), (7,8), (7,10), (7,12)\} \end{aligned}$$

Part (a) [6 MARKS]

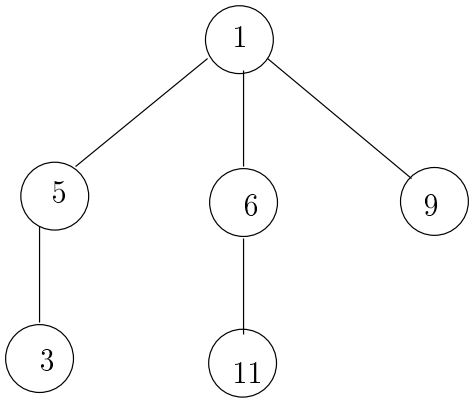
How many connected components does it have? List the vertices of each connected component separately.

Solution: It has 3 connected components with vertex sets: $\{2\}$, $\{1, 3, 5, 6, 9, 11\}$, and $\{4, 7, 8, 10, 12\}$.

Part (b) [9 MARKS]

Draw the Breadth-First-Search (BFS) tree rooted at vertex 1, assuming that the edges incident with every vertex are examined in ascending numerical order.

Solution:



Question 2. [25 MARKS]

In an undirected graph $G(V, E)$, a vertex-cover is a set $S \subseteq V$ of vertices such that every edge of G has at least one end-point in S (so the vertices in S "cover" all the edges). Trivially, $S = V$ is always a vertex-cover.

Part (a) [5 MARKS]

Find a vertex-cover in the graph of Question 1 of small size (as small as you can).

Solution: $S = \{1, 3, 4, 6, 7\}$.

Part (b) [10 MARKS]

Explain (clearly) an algorithm that when given $G(V, E)$ and $S \subseteq V$ outputs whether or not S is a vertex-cover of G.

Solution: We assume that for every vertex v, membership $v \in S$ can be checked in constant time. For instance, S can be given as a 0/1 array of size $|V|$ such that $S[u] = 1$ if and only if u is in the set.

```
Provedure TestVertexCover (G, S)
for each vertex v ∈ V do
    if v ∉ S then
        for each neighbor u of v do
            if u ∉ S then
                return "S is not a vertex-cover"
            end if
        end do
    end if
end do
return "S is a vertex-cover"
```

Part (c) [5 MARKS]

What is the running time of your algorithm in terms of $|V|$ and $|E|$ if the graph is given by adjacency list and why?

Solution: For each vertex $v \in V$, the second "for loop" is iterated at most $degree(v)$ times and each test of membership in S takes constant time. Therefore, each edge of the graph is examined at most twice (once from each end-point). Thus the running time is at most $O(|V| + |E|)$.

Part (d) [5 MARKS]

What is the running time of your algorithm in terms of $|V|$ and $|E|$ if the graph is given by adjacency matrix and why?

Solution: If we have adjacency matrix, then to find neighbors of a vertex v we have to check all the $|V|$ vertices. Therefore the two nested loops take (at worst case) $O(|V| + |V|^2) \in O(|V|^2)$ time.

Total Marks = 40