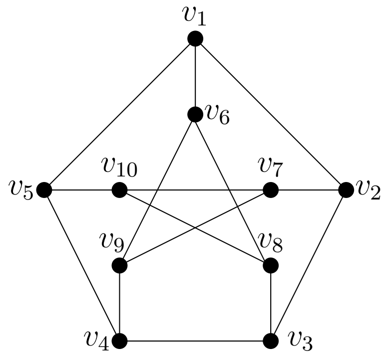
MAS210 Graph Theory Exercises 2 Solutions

Q1 Consider the following graph G.



(a) *An implementation of the basic tree growing algorithm starting at* v_7 *produces the following tree* T_5 *at the end of the fifth iteration:* $V(T_5) = \{x_1, x_2, x_3, x_4, x_5\}$ *where* $x_1 = v_7$, $x_2 = v_{10}$, $x_3 = v_5$, $x_4 = v_2$, $x_5 = v_8$, *and* $E(T_5) = \{v_7v_{10}, v_{10}v_5, v_7v_2, v_{10}v_8\}$. *List the edges of* G *which could be added to* T_5 *in the next iteration.*

$v_7v_9, v_5v_4, v_5v_1, v_2v_3, v_2v_1, v_8v_6, v_8v_3$

(b) *An implementation of breadth first search starting at* v_7 *produces the following tree* T_5' *at the end of the fifth iteration:* $V(T_5') = \{x_1, x_2, x_3, x_4, x_5\}$ *where* $x_1 = v_7$, $x_2 = v_{10}$, $x_3 = v_2$, $x_4 = v_9$, $x_5 = v_8$, *and* $E(T_5') = \{v_7v_{10}, v_7v_2, v_7v_9, v_{10}v_8\}$. *List the edges of* G *which could be added to* T_5' *in the next iteration.*

$v_{10}v_5$

(c) *An implementation of depth first search starting at* v_7 *produces following tree* T_5'' *at the end of the fifth iteration:* $V(T_5'') = \{x_1, x_2, x_3, x_4, x_5\}$ *where* $x_1 = v_7$, $x_2 = v_{10}$, $x_3 = v_5$, $x_4 = v_4$, $x_5 = v_9$, *and* $E(T_5'') = \{v_7v_{10}, v_{10}v_5, v_5v_4, v_4v_9\}$. *List the edges of* G *which could be added to* T_5'' *in the next iteration.*

v_9v_6

Q2(a) Let D *be a digraph and* v *be a vertex of* D. *Adapt the basic tree growing algorithm for (undirected) graphs to get an algorithm to construct a directed tree which contains all vertices in* D *which can be reached from* v *by a directed walk.*

We use the following iterative procedure. In the i'th step of the iteration we construct a directed tree T_i with vertices labeled $x_1, x_2, \ldots, x_i$ and with all arcs directed away from v.

Initial Step Put $x_1 := v_1$ and let T_1 be the directed tree with $V(T_1) = \{x_1\}$ and $E(T_1) = \emptyset$.

Iterative Step Suppose we have constructed a directed tree T_i with $V(T_i) = \{x_1, x_2, \ldots, x_i\}$ for some $i \geq 1$.

- If some arc e of D has a vertex x_j of T_i as its tail and a vertex y of $D - T_i$ as its head then put $x_{i+1} = y$ and $T_{i+1} := T_i + x_{i+1} + e$.
- If no arc of D goes from T_i to $G - T_i$ then STOP. Put $T = T_i$ and output T.

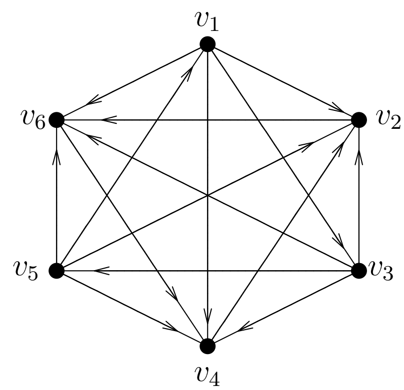
(b) *Indicate briefly why your algorithm is an efficient algorithm.*

In each iteration we check at most all the arcs whose tail belongs to T_i, and hence we check at most $|A(D)|$ arcs. Since each iteration increases the number of vertices of T_i, the number of iterations is at most $|V(G)|$. Thus the complexity of the algorithm is $O(|V(G)||A(D)|)$.

(c) *How could you use your algorithm (and the analogous algorithm for finding all vertices of D which can reach v by a directed walk) to find the strongly connected component of D which contains v?*

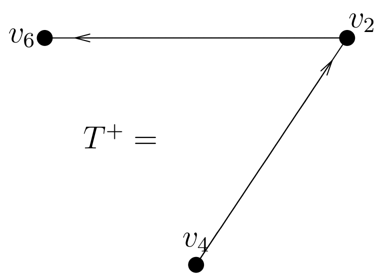
Construct a directed tree T^+ which contains all vertices in D which can be reached from v by a directed walk, and a directed tree T^- which contains all vertices in D which can reach v by a directed walk. Then $V(T^+) \cap V(T^-)$ is the set of all vertices which are linked to v by a directed walk in both directions. So $V(T^+) \cap V(T^-)$ is the vertex set of the strongly connected component H of D which contains v. We construct H by letting $V(H) = V(T^+) \cap V(T^-)$ and $A(H)$ be the set of all arcs in D whose head and tail both belong to $V(H)$.

(d) *Illustrate this algorithm by using it to construct the strongly connected component containing v_4 in the following digraph D.*



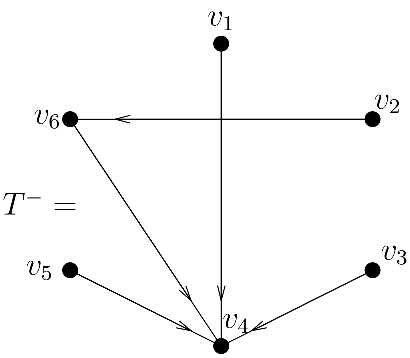
We have

$T^+ =$



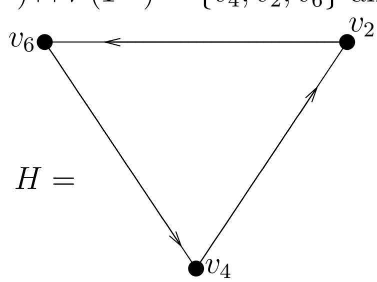
and

$T^- =$



Thus, if H is the strongly connected component which contains v_4, then $V(H) = V(T^+) \cap V(T^-) = \{v_4, v_2, v_6\}$ and

$H =$



Q3 (a) *Let G be a connected graph and e be an edge of G with end vertices u and v. Let $G - e$ be the graph obtained by deleting the edge e from G. Prove that every vertex x of $G - e$ is connected to either u or v by a path in $G - e$. (*Hint: choose a path from x to u in G.*)*

Since G is connected, x is joined to u by a path $P = v_0 e_1 v_1 \dots e_m v_m$ in G (where $x = v_0$ and $u = v_m$). If $e \notin E(P)$ then P is a path joining x to u in

$G - e$. On the other hand, if $e \in E(P)$ then $e = e_m$ and hence $v_{m-1} = v$. Thus $P' = P = v_0 e_1 v_1 \dots e_{m-1} v_{m-1}$ is a path from x to v in $G - e$.

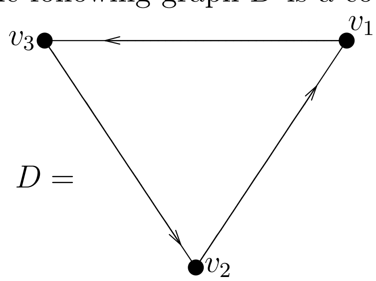
Deduce that, if e is a bridge of G, then $G - e$ has exactly two connected components, G_1, G_2, with u in G_1 and v in G_2.

By the above, every vertex of $G-e$ belongs to the same connected component of $G - e$ as either u or v. Since $G - e$ is disconnected, it follows that $G - e$ has exactly two components, each containing either u or v.

(b) *Is it true that if D is a strongly connected digraph and e is an arc of D, then $D - e$ has at most two strongly connected components? Give a proof or counterexample.*

It is false. The following graph D is a counterexample.

$D =$



D is strongly connected and $D - e$ has three strongly connected components for all arcs e of D.

Q4(a) *Let D be a digraph with no directed cycles. Prove that D contains a vertex u with $d^-(u) = 0$ and a vertex v with $d^+(v) = 0$. (*Hint: choose a longest path in D.*)*

Let $P = v_0 e_1 v_1 \dots e_m v_m$ be a directed path of maximum length in D. There are no arcs in D from any vertex of $V(T) - V(P)$ to v_0 since otherwise we could extend P. There are no arcs in D from any vertex of $V(P)$ to v_0 since otherwise we would obtain a directed cycle in D. Thus $d_D^-(v_0) = 0$. Similarly, $d_D^+(v_m) = 0$.

(b) *Prove that the edges of every loopless graph can be directed in such a way that the resulting digraph has no directed cycles. (*Hint: experiment with some small graphs.*)*

Let G be a graph. Draw G in the plane with all its vertices on a horizontal line and direct all edges from left to right. The resulting digraph D can have no directed cycles since every directed walk with at least one arc must move continuously towards the right so cannot return to its first vertex.