Assignment 1 Solutions.

Note Title 08-09-2019

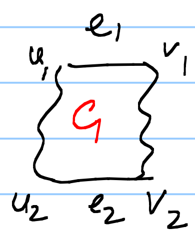
(1) $\deg(v)$ lies in the range $\{1, \dots, n-1\}$ ($\because$ the graph is connected, no vertex has degree 0).
$\because$ there are n vertices, pigeonhole principle $\Rightarrow$ there exist two vertices with the same degree.

(2) (i) Reflexive – No $\because$ self loops are not allowed
Symmetric – Yes
Transitive – No $\because$

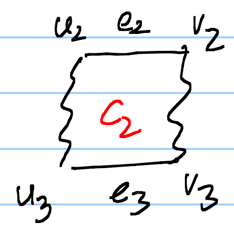
uRv, vRw but $u \not R w$.

(ii) Reflexive – No $\because$ graph may not have any cycles.
Symmetric – Yes
Transitive – Yes : Sps $e_1 = (u_1, v_1)$ R $e_2 = (u_2, v_2)$ and e_2 R $e_3 = (u_3, v_3)$

Let C_1 be the cycle containing e_1, e_2 and assume it looks like



and suppose C_2, the cycle containing e_2, e_3 looks like



Now use max-flow min-cut theorem.
If we remove any vertex, there will be a path joining e_1 and e_3. $\Rightarrow$ there are two vertex disjoint paths between them

(5) Arrange the vertices in topological sort.

$v_1, v_2, \ldots, v_n$.

Now assign them a level as follows:

for $i = 1 \ldots n$

Let $v_1', v_2', \ldots, v_k'$ be the vertices which have an edge to v_i.

All these vertices have been seen already.

Define $\text{level}(v_i) = \max(\text{level}(v_1'), \ldots, \text{level}(v_k')) + 1$.

By definition, no edge goes between vertices of the same level.

Now we show that $\text{level}(v) \leq k+1$.

Suppose not. Let v_{k+2} be a vertex in level $k+2$

$\Rightarrow \exists\, v_{k+1}$ in level $k+1$ such that $v_{k+1} \to v_{k+2}$ is an edge.

Arguing similarly we see that $\exists\, v_1, \ldots, v_k$ such that

$v_1 \to v_2,\ v_2 \to v_3,\ \ldots,\ v_k \to v_{k+1}$ are edges $\Rightarrow \exists$ path of length $k+1$. A contradiction.

(6) The algorithm maintains a set S. The pseudocode is:

Initialize $S \leftarrow$ set of all vertices

Let $G[S]$ = subgraph of G induced by S (i.e., only those edges which go between the vertices in S)

repeat {

while $G[S]$ has a vertex v of degree $< k$

Remove v from S

} until there is no such vertex in $G[S]$;
Output S;

Let O denote an optimal solution.
We will prove the following by induction of t:
Let S_t be the set S after t iterations of the repeat-until loop above.
Then, $O \subseteq S_t$.

Clearly, this will show that the set S produced by the algorithm is optimal.

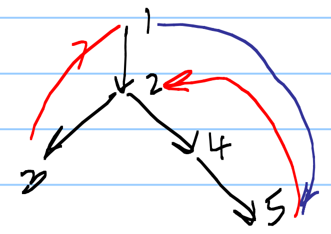
Base Case: $t=0$ is correct $\because$ $S_0 = V$ (set of all vertices).

Suppose the fact is true for S_t and let v be the vertex found in $G[S_t]$ of degree $< k$.
$\therefore$ the edges in $G[O]$ is a subset of $G[S_t]$ (by induction hypothesis)
v cannot be in $O \Rightarrow O \subseteq S_{t+1}$ as well.

3. This was done in class.

Let v_1 be the vertex in the cycle with the earliest start time.
Then v_k will be a descendant of $v_1 \Rightarrow (v_k, v_1)$ is a back edge.

It is possible that more than 1 edge becomes back edge. eg, consider the graph



Suppose DFS visits
1, 2, 3, 4, 5

then (3,1), (5,2) are back edges
and the cycle is 3, 1, 5, 2, 3