Distributed Systems Assignment

s1140740

2.1

Proof: If V is a vector clock, prove that $a \to b \iff V(a) \leq V(b)$.

$a \to b \implies V(a) \leq V(b)$

There are three possibilities:

1. Event b was a local event of a process i, by definition $V(b)[i]+ = V(b)[i] + 1$
2. Event b was a send message event of a process i.
3. Event b was a receive message event of a process i.

$V(a) \leq V(b) \implies a \to b$

2.2 Inductive proof on the position of the request in the queue

Base case

Request is at position 1 in the queue and thus the process can access the resource and satisfy the request.

Induction hypothesis

If our request eventually gets satisfied at position k, it also eventually gets satisfied at position $k + 1$.

Inductive step

Our request is at position $k+1$. Let the request at first position belong to process i and let us call the request R_i. Since R_i is at the first position in the queue, all the previous processes accessing the resource are finished with it, otherwise we would have their requests in our queue before R_i (we add a request to the queue whenever we get a `REQUEST` message and remove it only once we receive a `RELEASE` message for it). Thus, there are three cases:

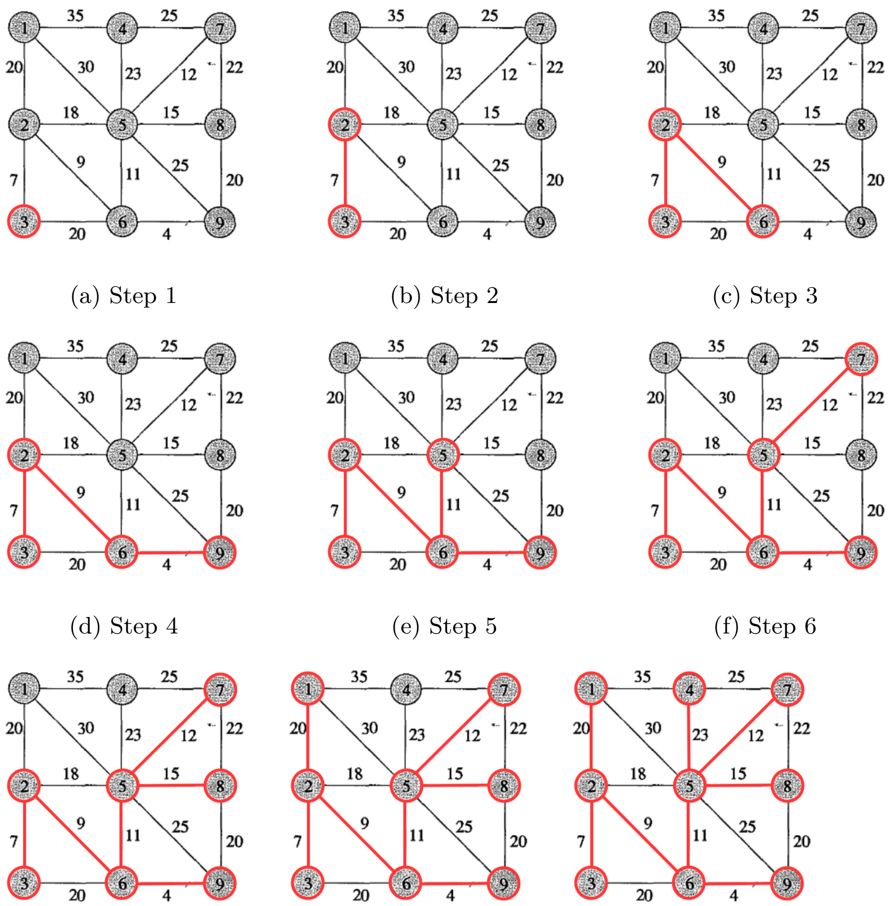
1. Process i is currently accessing the resource. Since we assume processes do not fail, this means that it will eventually finish accessing it and when it does it will send us a `RELEASE` message and R_i will be removed from our queue and our request will be in position k.
2. Process i has already finished accessing the resource. This implies we have not yet received the `RELEASE` message. Channels do not fail so we will eventually receive the message and remove R_i from our queue and our request will be in position k.
3. Process i has not started accessing the resource. This implies R_i is not yet at the top of the queue of process i (otherwise it would just access the resource). However, since we have shown that no process with a request before R_i can be accessing the resource this means that process i just has not received the `RELEASE` message from the last process accessing the resource. Once it receives this message it will pop that process's request of its queue and start accessing the resource. Logic in case 1 can then be followed to show that our request will advance to position k.

2.3

The weighted diameter of this graph is 7. The path realising this diameter is $A \to C \to E \to G \to H$.

If the graph was unweighted, the diameter would be 4 and the corresponding path would be $A \to C \to F \to G \to I$.

2.4



(a) Step 1 (b) Step 2 (c) Step 3

(d) Step 4 (e) Step 5 (f) Step 6

(g) Step 7 (h) Step 8 (i) Step 9

Figure 1: Prim's algorithm