

be the minimum value among all the possible spanning trees in this graph).

## 1 Prim's Algorithm

(Goodrich2011, p.651) defines Prim's algorithm. See also <https://bit.ly/2VLz3DK>. It is an efficient algorithm; it requires  $O((m+n) \log_2 n)$  time, if we use priority queues. In this exercise you do not need to implement a priority queue; assume that you can always compute the minimums in your head and grow the MST accordingly.

## 2 Problem

We start with the graph shown in Figure 1.

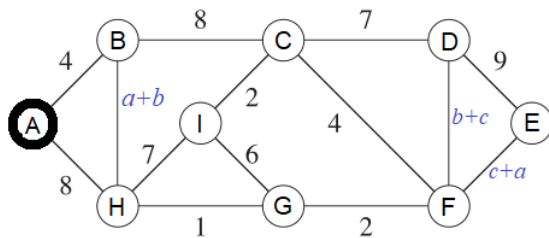


Figure 1: Graph diagram

(A) Redraw the graph Figure 1, replace the edge weights  $a + b$ ,  $b + c$  and  $c + a$  by the from your Student ID). Vertex  $A$  will be your source vertex.

(B) Draw 9 copies of the graph (no need to rewrite the edge weights all the time – just keep them before your eyes as you run the Prim's algorithm). Show how the Prim's MST (Minimum Spanning Tree grows) one edge at a time. Highlight the edges selected for MST (make them bold or color them differently). The first graph in this sequence contains no selected edges of MST, but the last graph in this sequence contains the full MST. (Goodrich2011, p.652) contains sample pictures how to run the Prim's algorithm.

(C) Add up the total weight of the obtained MST and write this in your answer (it should