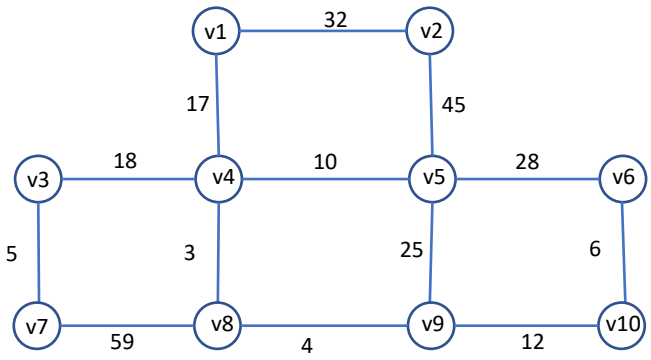
Name: Sabyasachi Sahoo
COSC 40403 - Analysis of Algorithms: Fall 2017: Homework 6
Due: 23:59:59 on November 9, 2018

Question	Points	Score
1	10	
2	10	
3	10	
4	15	
Total:	45	

1. (10 points) Use Prim's algorithm to find a minimum spanning tree for the following graph. Show the actions step by step.



Solution:
Step 1 : No self loops or parallel edges, hence do nothing.

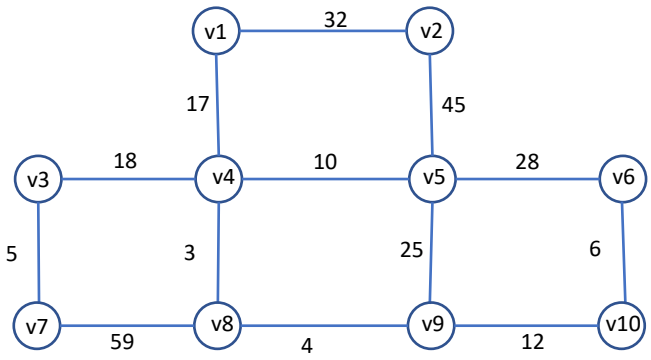
Step 2 : Choose any arbitrary note as root node.
we select v1.

Step 3 : Check outgoing edges and select the one with less cost.

Added - (v1, v4) = 17
Added - (v4, v8) = 3
Added - (v8, v9) = 4
Added - (v4, v5) = 10
Added - (v9, v10) = 12
Added - (v10, v6) = 6
Added - (v4, v3) = 18
Added - (v3, v7) = 5
Added - (v1, v2) = 32

Hence, the list of added edges is the MST produced by Kruskal's algorithm.

2. (10 points) Use Kruskal's algorithm to find a minimum spanning tree for the graph shown above. Show the actions step by step.



Solution: Step 1 : No self loops or parallel edges, hence do nothing.

Step 2 : Arrange all edges in their increasing order of weight.

(v4, v8) = 3
(v8, v9) = 4
(v3, v7) = 5
(v6, v10) = 6
(v4, v5) = 10
(v9, v10) = 12
(v1, v4) = 17
(v3, v4) = 18
(v5, v9) = 25
(v5, v6) = 28
(v1, v2) = 32
(v2, v5) = 45
(v7, v8) = 59

Step 3 : Add the edge which has the least weightage while keeping the MST property intact.

Added - (v4,v8) = 3
Added - (v8, v9) = 4
Added - (v3, v7) = 5
Added - (v10, v6) = 6
Added - (v4, v5) = 10
Added - (v9, v10) = 12
Added - (v1, v4) = 17
Added - (v3, v4) = 18
(v5, v9) and (v5, v6) aren't added as they create circuits
Added - (v1, v2) = 32
(v2, v5) and (v7, v8) aren't added as they create circuits

Hence, the list of added edges is the MST produced by Kruskal's algorithm.

3. (10 points) Use Huffman's algorithm to construct an optimal binary prefix code for the letters in the following table.

Letter	c	e	i	r	s	t	x
Frequency	0.11	0.22	0.16	0.12	0.15	0.10	0.14

Encode each word using your binary code.

1. rise
2. exit
3. text
4. exercise

Solution:
Symbol – Huffman Encoding
e – 01
c – 001
i – 111
r – 100
s – 110
t – 000
x – 101

rise: 10011111001
exit: 01101111000
text: 00001101000
exercise: 011010110000111111001

4. (15 points) Implement Huffman's algorithm using Python and run it on problem 3.

Solution: Jupiter notebook attached in the zip file.