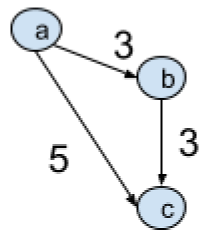
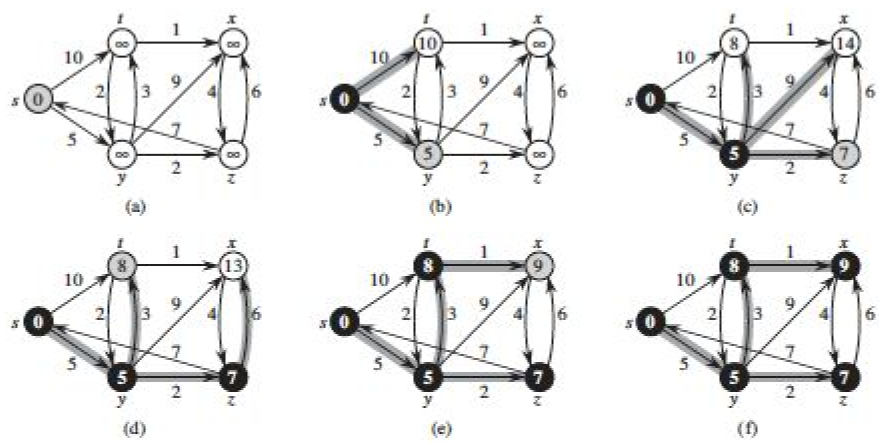
Solutions for Assignment 4

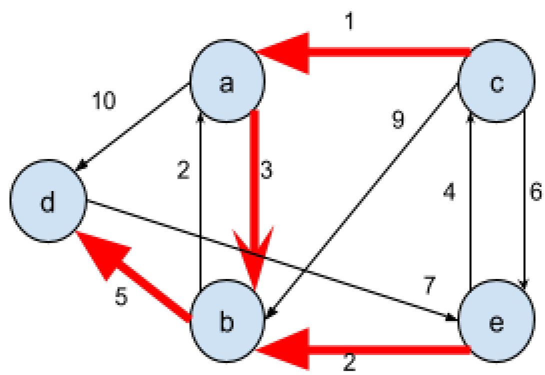
1. False! In the following graph, check the shortest path from *a* to *c* before and after the replacement.



2. A: It's easy to see if you reverse the edges, it becomes the single source shortest path destination. So the algorithm is simply reverse the graph, run Dijkstra and reverse again
The complexity would be the cost of investing, that is O(|E|) + the cost of Dijkstra, that is O(|E| lg |V|)
B) I took the example from the book, just reversed the links! So if we reverse the links of the graph given in the question, and run Dijkstra, the results would be something like this (copied from the book):



So at the end we reverse the solution, and we get the final result:



So e to d is e → b → d : 7
And c to d: c → a → b → d : 9

3. This is called Single-Linkage clustering and can be done efficiently using Kruskal's algorithm. Technically you can do it either Top-Down or Bottom-Up. In top-down, you get the MST, and start removing from the most expensive link. I will use Bottom-Up approach, sometimes referred to "agglomerative clustering":

a) Run Kruskal, stop when you have k components (you need to keep the number of components in a counter which is |V| at the beginning, and decreases with every merge!
b) Runk Kruskal, stop when the next link that you want to add has a weight equal or greater than δ
c) It reduces between two sets with possibly thousands or millions of nodes to one edge! Imagine if Victoria Beckham is in singers cluster and David Beckham in Soccer Players, one might think soccer players are very similar to Singers!