#### BFS Algoritması

Bu algoritma bir S düğümünden T düğümüne en kısa yolu ve uzaklığı bulur. Algoritmadaki L etiketlenmiş düğümler kümesidir. A düğümünün önceli, L kümesinde A ‘yı etiketlemek için kullanılan düğümdür*.* Algoritma aşağıdaki adımlardan oluşmaktadır.

***Adım 1 ( S ‘ i etiketle )***

1. ***S’ e 0 etiketini ver , S’in önceli yok olsun .***
2. ***L={S} ve k=0***

***Adım 2 ( Düğümleri etiketle)***

***repeat***

***Adım 2.1 ( etiketi artır)***

***k=k+1***

***Adım 2.2 (etiketlemeyi genişlet)***

***While ( L’de k-1 etiketli, L’de olmayan bir W düğümüne komşu olan bir V düğümü var)***

***a)W düğümüne K etiketini ata***

***b)V düğümünü W’nin önceli olarak ata***

***c)W düğümünü L kümesine ekle***

***endWhile***

***until (T, L kümesinin içinde yada L’deki düğümlere ,L’de olmayan düğümlerden hiç biri komşu değil)***

***Adım 3( T’ye en kısa yolu oluştur.)***

***if(T,L kümesinde)***

***S düğümüne erişene kadar T düğümünün önceli, öncelinin önceli v.b. üzerinden en kısa yolu oluştur. T ’nin etiketi S ’den T ’ye uzaklığı gösterecektir.***

***Otherwise***

***S ’den T ’ye yol yoktur.***

***Endif***

Bir düğümü etiketlemek ve bir kenarı, komşu kenarı bulmakta kullanmayı, algoritmanın analizinde elemanter işlem olarak düşündüğümüzde, n düğümlü ve e kenarlı bir grafta , her düğüm bir kez etiketlenmekte ve her kenar komşu düğümü bulmak için en çok bir kez kullanılmaktadır. Buna göre en çok n+e işlem yapılacaktır.

n+e<= n+C(n,2)=n+1/2(n.(n-1)) olduğuna göre algoritma en çok n2 mertebesindedir.

Bu algoritmayı kullanarak aşağıdaki grafın B düğümünden G düğümüne en kısa yolunu bulunuz. Etiketli grafın son halini çiziniz ve en kısa yolu grafta da gösteriniz.

