**Analisis Pengunaan Algritma**

**BFS (Pencarian Melebar) dan DFS (Pencarian Mendalam) pada Graf**

Pemecahan masalah dengan representasi graf dilakukan pertama-tama dengan membuat representasi objek masalah sebagai simpul dalam graf serta membuat representasi hubungan antar objek dengan garis yang menghubungkan simpul-simpul tersebut. Setelah itu, setiap simpul dalam graf dikunjungi secara sistematis (traverse).

Dalam graf terdapat dua macam algoritma traversal, yaitu:

1. Algoritma pencarian melebar (breadth-first search / BFS)
2. Algoritma pencarian mendalam (depth-first search / DFS)

**Algoritma BFS (Pencarian Melebar)**

Algoritma pencarian melebar (BFS) mengunjungi setiap simpul dalam graf mulai dari simpul akar, lalu dilanjutkan dengan mengunjungi simpul-simpul yang bertetangga dengan simpul akar tersebut. Setelah itu, untuk setiap simpul yang sudah dikunjungi tersebut, dikunjungi simpul-simpul yang bertetangga dengannya dan belum dikunjungi, demikian seterusnya sampai seluruh simpul berhasil dikunjungi.

Dalam algoritma BFS, simpul anak yang telah dikunjungi disimpan dalam suatu antrian. Antrian ini digunakan untuk mengacu simpul-simpul yang bertetangga dengannya yang akan dikunjungi kemudian sesuai urutan pengantrian. Untuk memperjelas cara kerja algoritma BFS beserta antrian yang digunakannya, berikut langkah-langkah algoritma BFS:

1. Masukkan simpul ujung (akar) ke dalam antrian
2. Ambil simpul dari awal antrian, lalu cek apakah simpul merupakan solusi
3. Jika simpul merupakan solusi, pencarian selesai dan hasil dikembalikan.
4. Jika simpul bukan solusi, masukkan seluruh simpul yang bertetangga dengan simpul tersebut (simpul anak) ke dalam antrian
5. Jika antrian kosong dan setiap simpul sudah dicek, pencarian selesai dan mengembalikan hasil solusi tidak ditemukan
6. Ulangi pencarian dari langkah kedua

**Algoritma DFS (Pencarian Mendalam)**

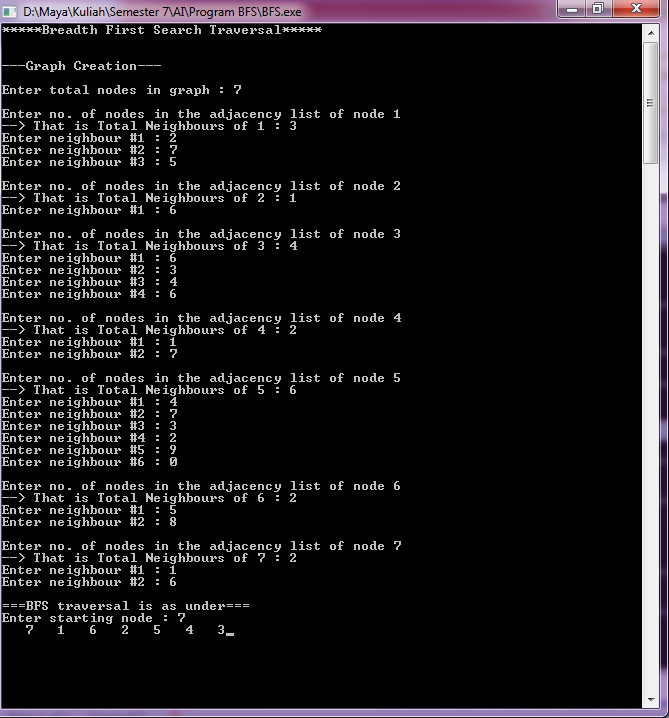
Algoritma pencarian mendalam (DFS) mencari solusi dengan mengunjungi simpul akar, lalu simpul-simpul yang bertetangga dengan simpul akar (setingkat di bawahnya), terus sampai simpul paling dalam pada bagian tersebut. Setelah itu, dicari simpul yang telah dikunjungi pada tingkat terdekat. dan terdalam, lalu simpul yang bertetangga dengan simpul ini dikunjungi, demikian seterusnya sampai seluruh simpul telah dikunjungi.

Dalam algoritma DFS, simpul yang telah dikunjungi disimpan dalam suatu tumpukan (stack). Antrian ini digunakan untuk mengacu simpul-simpul yang akan dikunjungi sesuai urutan tumpukan (masuk terakhir, keluar pertama) dan mempermudah proses runut-balik jika simpul sudah tidak mempunyai anak (simpul pada kedalaman maksimal). Untuk memperjelas cara kerja algoritma DFS beserta tumpukan yang digunakannya, berikut langkah-langkah algoritma DFS:

1. Masukkan simpul ujung (akar) ke dalam tumpukan
2. Ambil simpul dari tumpukan teratas, lalu cek apakah simpul merupakan solusi
3. Jika simpul merupakan solusi, pencarian selesai dan hasil dikembalikan.
4. Jika simpul bukan solusi, masukkan seluruh simpul yang bertetangga dengan simpul tersebut (simpul anak) ke dalam tumpukan
5. Jika tumpukan kosong dan setiap simpul sudah dicek, pencarian selesai dan mengembalikan hasil solusi tidak ditemukan
6. Ulangi pencarian dari langkah kedua

**Analisis Program**

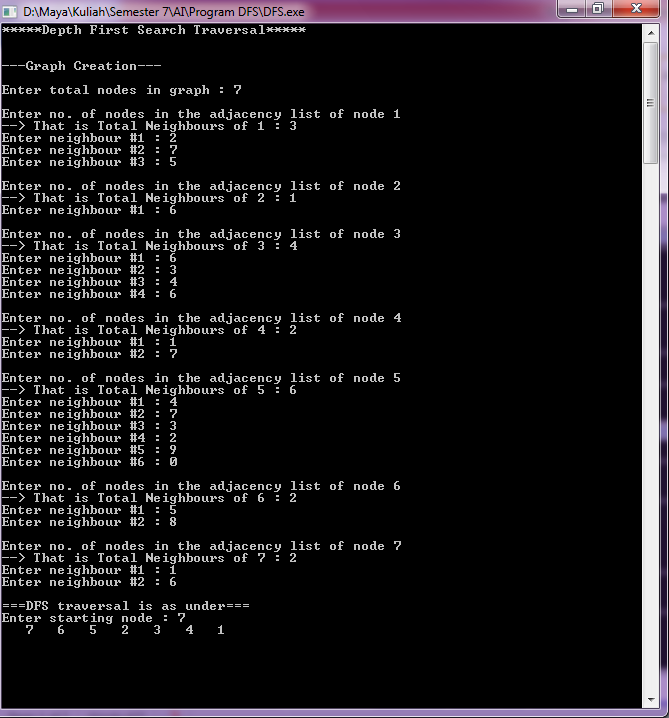
* Program BFS



Berdasarkan program BFS diatas dapat dilihat bahwa pada algoritma BFS pencarian akan dimulai dari akar (starting node) yang dalam contoh diatas adalah node 7. Node yang dikunjungi setelah node 7 adalah node yang bertetangga dengan node 7 yaitu node 1 dan node 6. Node yang dikunjungi selanjutnya adalah node 2 (node yang bertetangga dengan 6), setelah itu node 5 (node yang bertetangga dengan 2), setelah itu node 4 (node yang bertetangga dengan 5) dan terakhir adalah 3 (node yang bertetangga dengan 4. Sehingga urutan node yang dikunjunginya adalah:

7 1 6 2 5 4 3

* Program DFS



Berdasarkan program DFS diatas dapat dilihat bahwa pada algoritma DFS pencarian akan dimulai dari akar (starting node) yang dalam contoh diatas adalah node 7. Node yang dikunjungi selanjutnya adalah node 6, setelah itu node 5, node 2, node 3, node 4, dan terakhir node 1. Sehingga urutan node yang dikunjunginya adalah:

7 6 5 2 3 4 1