Requirements of finding out whether a given graph is n-partite,bipartite or non-partite

Santrupti Nerli

19 Sep 2009

1   Introduction

This software can be used to determine whether a given graph is n-partite,bipartite or non-partite.

2   General Description

An n-partite graph G(V,E) is a graph in which there will be n vertex subsets say $(V_1, V_2...V_n)$,which are disjoint (i.e,$V_1 \cap V_2 \cap ... \cap V_n$ is a null set) and there should be no edge between two vertices belonging to the same set. If n is 2, then the corresponding graph becomes bipartite, which will have two vertex subsets.The graphs are shown below.
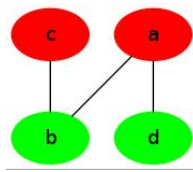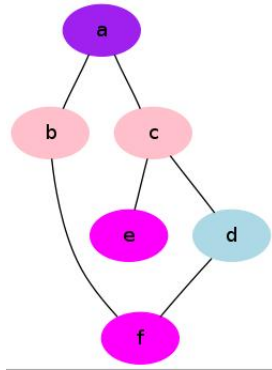

Figure 1: Bipartite Graph

Figure 2: 4-Partite Graph

3   Functionality

Random graph is taken as the input. The graph is then checked if it has no common vertices in vertex subsets. If yes then further it is checked if there is any edge which has vertices in a same set. If the graph does not have an edge with vertices in same set then it is declared as a n-partite or a bipartite graph. If any of the above mentioned conditions fail, then the graph is non-partite. The output is displayed accordingly.

4 External Interface

The program takes input from a file. The resulting output is displayed on the command line. No operations require external hardware.

5 Performance

Each entry in the adjacency matrix requires only one bit, they can be represented in a very compact way,occupying only n^2 / 8 bytes of contiguous space, where n is the number of vertices. With an adjacency matrix, an entire row must instead be scanned, which takes $O(n)$ time. The adjacency matrix of the graph requires $\Theta(n^2)$] memory,independent of number of edges in a graph.

6 Design Constraints

- The software uses C language.

- The platform used is Linux with GCC.