***Digital Design ||***

***1st Project Report***

***Logic Circuit Schematic Beautifier***

***Dr Mohamed Shalan***

Name: Tasneem Ismail

ID: 900114884

* I made some changes on the mul4x4.v file and erased the wires from the top and the assign from the bottom of the file, but before removing the assign, I replaced the wires by their equivalencies.

***Plan of the code:***

I read the file sample.v, saving the inputs a lone in inputs vector, saving the outputs alone in outputs vector, then reading gates and make a connection between inputs of a gate and the gate itself, and a connection between the gate and the output, this is done by assigning a value of 1 to the adjacency matrix of:

adjMatrix[map[input]][map[gate\_name]]=1

adjMatrix[map[gate\_name]][map[output]]=1

Then, applying the BFS algorithm to determine the level of each gate.

***Structure of the code:***

* I had designed a DAG class, here is the DAG.h
* #include<iostream>
* #include<map>
* #include<vector>
* #include <queue>
* using namespace std;
* #ifndef DAG\_H
* #define DAG\_H
* class DAG
* {
* public:
* struct LEVEL // This is for each gate and its level
* {
* string gate;
* int level;
* };
* DAG(int n); // constructor
* ~DAG(); //destructor
* void Read(string fileName); //sample.v
* string removeSpaces(string s); //remove spaces within string
* bool IsInput(string s); //to check if a gate is one of the inputs
* bool IsOutput(string s); //to check if a gate is one of the inputs
* void PrintV(int n); // print the vector of gates
* void PrintM(int n); //print the map
* void PrintADJ(int n); // print the adjacency matrix
* void BFS(int n); //print each gate with its level
* private:
* map<string, int> Mapping; // mapping each gate to a number
* vector <string> gates; // names of gates
* int\*\* adjMatrix; //adjacency matrix if the value is 1, then node i and j are connected
* vector<string> inputs; //for saving inputs
* vector<string> outputs; //for saving outputs
* };

#endif