**Family tree**

BY: Nathan almaleh 209633635 && Adam Levari 204115547

Purpose of this assignment is to build a family tree with many functions.  
Insert node: adding a new node to our tree  
Search node: searching for a node in our tree based on a given name   
Search relation between 2 nodes: based on 2 names find the relation  
Search a person based on a person name and given relation,  
Display tree, output an understanding of the tree  
Remove (delete) remove and delete all object after the end of the process **Project Function and explanation.**Our project has one class name FT  
it contains:  
 class node  
 class Family tree  
 Main function

Struct node:

Variables for node

char name[50];

short int x; // x - height of node

bool g; // g- gender

node\* fc; // Pointer to first child

node\* ns; // Pointer to next sibiling

node\* ps; // Pointer to previous sibiling

node\* p; // Pointer to parent

node\* s; // Pointer to spouse

Node functions

node(); //default constructor

void setData(string, char);//add values to the new node

~node();

|  |
| --- |
|  |
|  |
|  |

Struct Family tree;

node\* traverseDown(node\*,string); // Search functions

node\* traverseRight(node\*, string); // Search functions

node\* search(string); // Main Search functions

void addSib(node\*, node\*); // Function for adding new members

void addChild(node\*, node\*); // Function for adding new child

void addSpo(node\*, node\*); // Function for adding new spouse

void addNew(string, char);

void addNew(string, char, char, string);

string find(string, string); // Function to find relations b/w 2 indeviduals

void display(node\*); // Function to display full tree

void destroy(node\*); // Function to destroy full tree

string findRelation(string, char); // Funcion to display person relative

void show(node\*); //function information about node

Main tester  
familyTree T;  
//Display tree  
T.display(T.start);

//"Find relation with relation assign (Father/Mother/Wife/Sibling/Cousin/Uncle/Aunt/Grandparent)"

T.findRelation("adam",'f' );

//finds relation between 2 nodes in the tree

T.find(“ham”,”posher”);

Tester output:

Left main tester

Right console output

