**Data structures 2017**

**Family Tree Assignment**

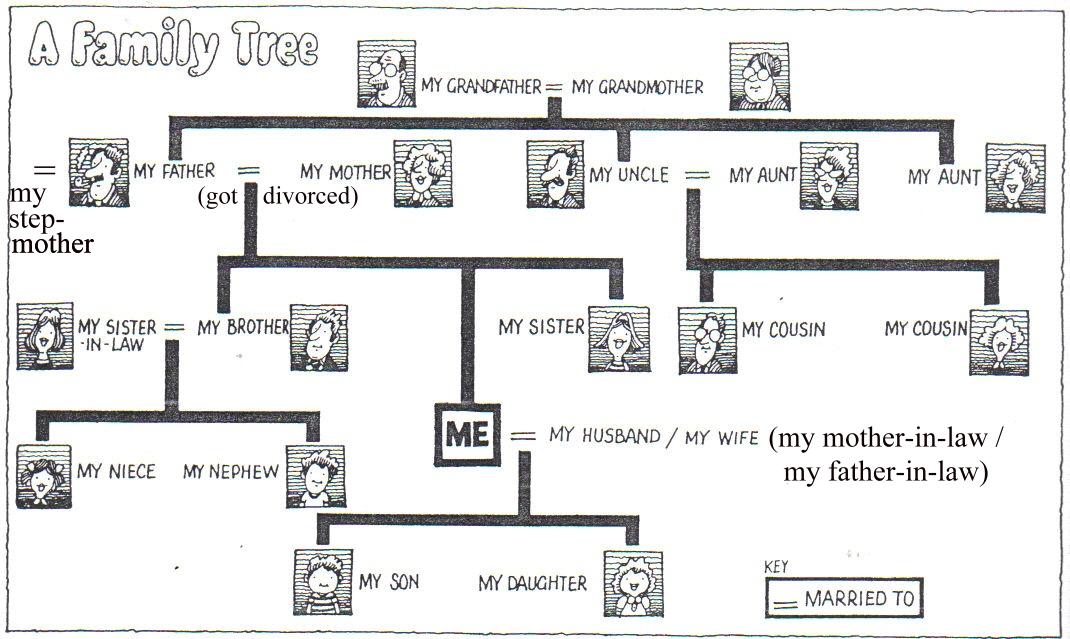
By Rachel Lawton

20072872

Introduction

In this report, I will be talking about how I went about planning, prototyping and implementing my family tree assignment. My assignment was all done on the Eclipse IDE.

A family tree also known as a Family genealogy, is usually a graph or tree picture that contains all past family members arranged in a specific way that is easy to read. Most family trees start with grandparents and then proceed to go to children who then become parents and have children of their own and so on and so forth



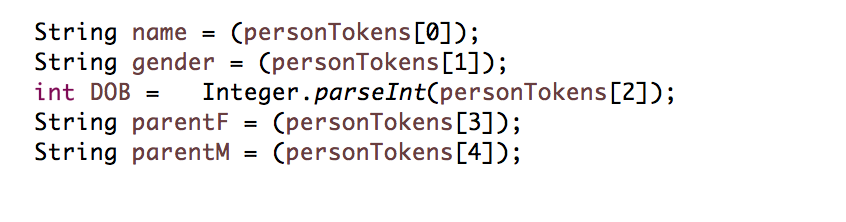
Planning

I planned this assignment all on paper before I went near my computer. The main topics I planned where:

* Data requirements.
* What data structures I will use.
* What relationships will I cater for.
* How will I test the code.
* What GUI will I use.
* What methods or operations will I need.

*Data requirements:* The requirements that I needed from the data were Name, Gender, DOB and mother and/or father. These requirements where given to use in the assignment spec

*What data structures will I need:*

The data structures that I choose where the most efficient and accurate in my mind. I chose an array list for reading in the file of names, I chose this because it allowed me to have multiple tokens which contained each piece of data I needed.

I also chose an array list for the children in the person class this was the most efficient because it was just a temporary holding space for the children of a tree.

Another data structure I used was the tree data structure. This is what made up the family tree itself. I used this data structure because it was required in the spec.

*What relationships will I cater for:*

My assignment caters for the basic relationships which are mother, father, children and siblings. I chose this because I wanted to be able to have the basics before I moved on to more complicated features.

*How will I test my code?*

I didn’t want to be coding “in the dark”. So, I used Junit testing, debugging as well as printing things so the console as my way of seeing what was going right or wrong. I used Junit testing because we learnt a lot about it last year in algorithms and was very handy back then. The use of the console using the “system.out.println” was useful for seeing what is happening at a certain point. The debugger was used for also seeing why something wasn’t working at a certain point by using the break point feature.

*What GUI will I use?* I decided on using Jswing like we used from lab 01 the calculator because I would have had enough knowledge about it for what I wanted. It also meets the requirements on the spec.

*What methods/operations will I need:*

The methods/operations I chose to use was

* Add function
* Reading in the file
* Find function
* To string
* Siblings

Prototyping

This was my plan at the start. This changed when it comes to the method and operation’s and GUI.

When I started to prototype I found out after a while that trying to make the family tree with an actual tree data structure is very difficult. I tried so many different methods for adding to a tree. But I just couldn’t wrap my head around it (all my attempts are commented out in my code).

The first problem I encountered was how to make a tree for one family since the root could only be one thing. I tried looking up different tree data structure like n-array trees but to no avail. I was confused on who I would put in the tree. Do I put the mother or the father? if I put the children on one do I not put it in the other tree?

The second problem I found was when trying to merge the trees or create a forest of trees. I did a lot of research on this but kept coming back with disjoint sets. Which I looked a lot into but didn’t think it was the right one to use.

In the end I just used user input to input a family tree into an arraylist using the console as a gui.