

Design Document

1. Employee Record Program, Ritesh Sood, 105977937, October 29 2016.
2. This program is a database that holds records of individuals. Each record holds 15 different fields of information in each record. The program allows the user to add, delete, search or modify any record.
3. Overall Software Architecture
The main functions are the Search/SubSearch, UpdateNode, deleteAnode, and AddaNode. The Search and Sub search both use the naïve algorithm to perform a contains search. They perform a different search for each field depending on what the user requires. The update node function uses the search functions to find the node. Then it asks the user which field to update. Takes the new criteria for the chosen field and puts it in a new node. The old node is deleted and this new node is passed into the tree. The delete node function uses the delete provided in the BSTree.cpp file. AddaNode simply creates a new Node, prompts the user for the fields and passes it into the tree. The Write to File function is very important because it holds different sorts for all of the fields. All the fields use the same sort algorithm which is the selective sort.
4. Input Requirements
The Program takes a file input of records to store the data in. Many of the inputs after the file read are from the user accessing different menu options. An int is used for all of the menu option inputs. There are inputs from the user when modifying or adding a new entry. All fields except for the I.D., which is a double, are made up of the strings. As far as updating or adding a new entry the program mostly takes in strings with the exception of a double for the I.D.
5. Output Requirements
The output is up the user. When the user is ready to write to file, the program asks how they would like the file to be sorted. The user can choose any field to sort it by except for the affiliates. Once the sort is chosen, the database is written out to a file called dataOut.txt.
6. Problem Solution Discussion
There are tree traversal algorithms used throughout the program. If it is not tree traversal for the search then the search is being done by a naïve algorithm. The naïve algorithm was the simplest way I found of doing a contains search. This algorithm is only used with vectors in the program. A selective sort is used for the Write to file function. This sort was the simplest for me to implement without using the built in sort.
7. Data Structures
For my sub-containers I only used vectors. I found them to be simple to work with

for this program. The use of more binary tree may have been more efficient, but I went ahead and chose vectors for the simplicity. There is a function which copies the BSTree into a vector. This function is called before any search that is not an ID search. A vector is used for every field that is not the ID field.

8. User Interface Scheme

The main menu lists 8 options. They are Read from file, Add a Node, Update a Node, Search an Entry, Delete an Entry, Display the database, Write to File/Sort, and Exit Program. Then there is The search menu which allows the user to search 9 different fields. Search and Sub-search share the same menu. User interface scheme should show the menu items at top level and items in sub menus and how to navigate through menus. There is also the update menu which just prompts the user to enter which field they would like to edit. Last is the sort menu which asks the user which field they would like to use for the sort.

