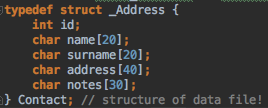
## 

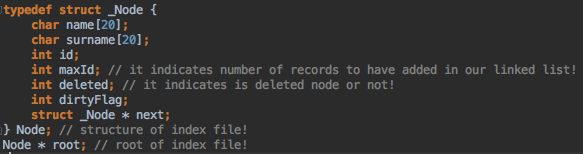
Abdulsamet İleri

2014510091

* My data object is Contact. it has



* I used **fixed length structure** to store data objects. So the data must be at most **114bytes.**



* To index data objects, I used linked list as a structure. Because it easy to add, delete, and update. Also, it easy to maintain.
* Program starts with **loadIndexFile2Ram()**. Thanks to this method, if any file named as indexfile.dat, it opens and reads context and puts all data to linked list. If there is no file as named as indexfile.dat, it does nothing.
* After the menu shows on the screen, all operations opens for the user command. All inputs will be taken from user is inside those methods. And in order to take inputs, we use scanf function as first parameters **“ %[^\]s”**, thanks to this regular expression there is no control I wrote down. This expression works like “**keeps taking in values from command line it encounters newline and spaces get saved as well.**”
* In case of add, update, delete I assigned a flag as named dirtyFlag in my linked list node 1 and as soon as I write, i store this value in created index file! And if program encouters run time error, and restarted, in **loadIndexFile2Ram() controls this flag if it set it create data and index file again(createNewIndexFile()), if it is 0, it continues normally.**
* In **updateRecord()** firstly, i want name and surname which to be updated from user, after i took this values, i want to new name, surname, address and notes.
* In **deleteRecord()** is very similar to updateRecord function, firstly i want name and surname which to be deleted after it starts delete operation.
* In **concat(char \* s1, char \* s2),** in order to match primary keys in the system(name and surname), I have to combine these two. So this function provides this.
* In **strlower(char \* ar)**, this function takes name and surname and it converts lowercase of these. So I matches primary keys in different record one another.
* In **writeDataIndexFile(),** if user enter 6, it means exit. Dirty flag is reset and all context within linked list in our ram written in the index file.
* In **sortLinkedList(Node \* newData),** it sorts records based on their value of primary keys.
* In **printList**, it tells us our structure of the ram(linked list).
* ***All operations are completed successfully!***