Project 1: Course Matching System

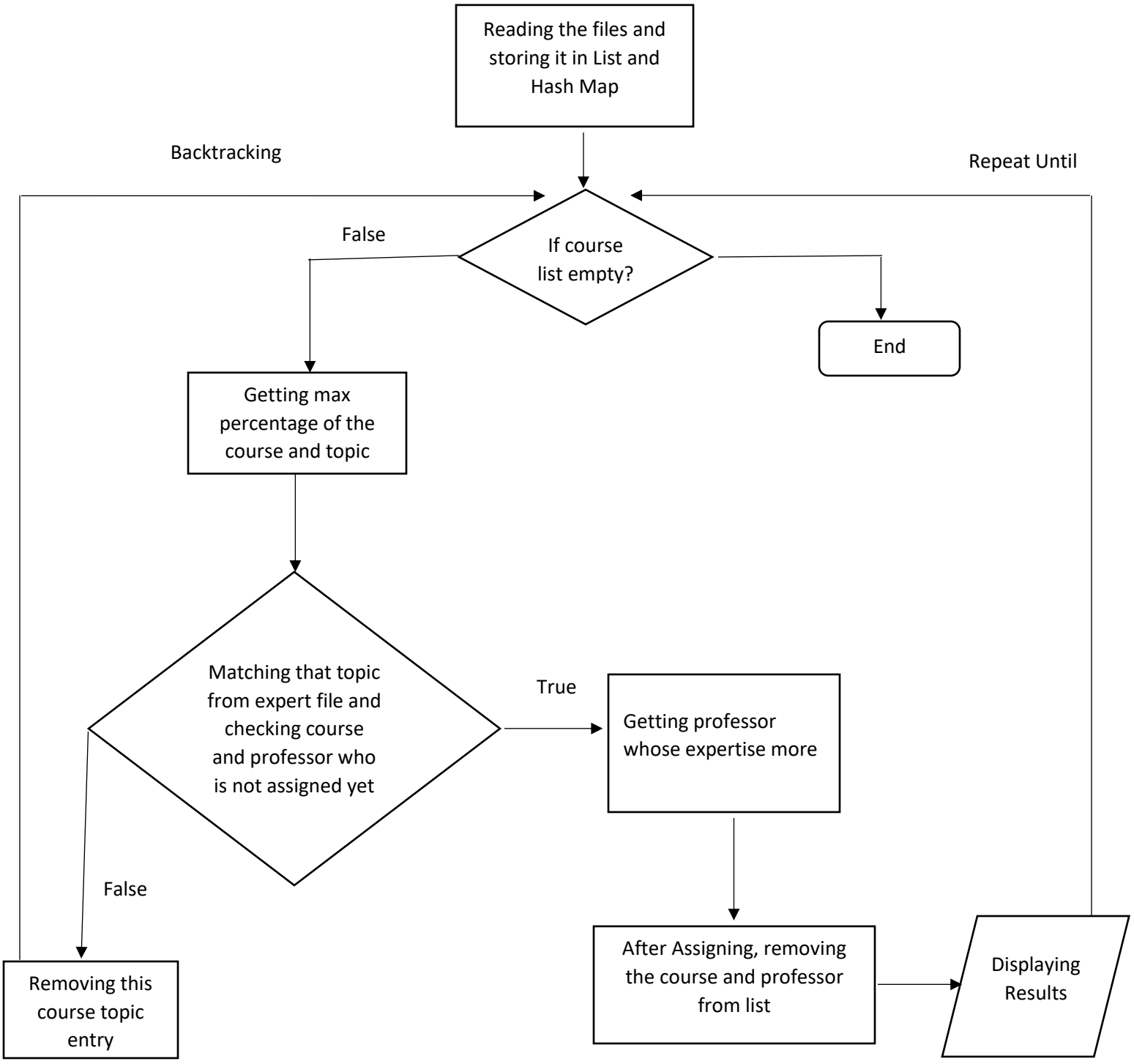
Name: Nageswara Rao Nandigam
16244177

Assumptions:

Assumed that the number of professors and courses is the same.

Taken consideration of every course topic should be teach by some professor

Brief Explanation of the Idea:



1) **Reading files at run time and storing courses and professors in list and courseTopic in Hashmap**
2) **Repeating the course until it is empty (i.e. until it is assigned to some professor)**
3) **Extracting the course which more topic percentage.**
4) **Using that topic for finding professor whose expertise is more**
5) **Matching that topic from expertise file and checking courses and professors whose is not yet assigned and having more expertise among those professors who teaches that topic**
6) **If it is matching with above conditions we are displaying the results and remove those professor and course from respective list**
7) **If it is not matching I am removing the entry from course topic and repeating the same steps from 2 until all courses are assigned**

• **How would you modify your algorithm if a professor must teach two courses instead?**

I will just keep a counter on how many times each professor allocated. That is something we can do it by adding Hasmap to store key as professor and value as how many times it assigned. That's how can look up in O(1).

Efficiency of Algorithms:

I am using three parametrs here

N= no of courses

M= no of entries in courseTopic file

K= no of entries in expert file

If I write in terms of Big o notation then it will be O(N*M*K)

References:

http://www.geeksforgeeks.org/backtracking-algorithms/

https://stackoverflow.com/

Github:

https://github.com/Nagumkc/DAA_Project