CS 461 – ARTIFICIAL INTELLIGENCE

HOMEWORK #4 (5% OR 10 POINTS)

Assigned: Mon 16 Nov 2020

Due: Wed 25 Nov 2020 ** 2 pm **

You can do this homework in groups of 5 (or less). Your group for this homework should normally coincide with your term project group. Do not forget to indicate clearly the students who are submitting this homework (i.e., write at most five names on the submission).

You must submit your homework (including all the code) to our TAs. Just a single submission per group! Our TAs may send you a note explaining the specifics.

Any programming language can be used as long as you have it available on a portable computer. Any group member should be prepared to give a demo (individually and using that portable computer) when requested to do so by our TAs.

The usual late policy applies.

In this homework you are expected to obtain the class-precedence list for a given class hierarchy. The normal way of doing this is to implement the 'fish hook algorithm' given by Winston (chapter 9). But reuse of any topological sorting program is also permitted, provided that you clearly state where you found it. You must also tell, in precise terms, how you modified it to fit the needs of this assignment.

You should test your program with the examples (class hierarchies) given in the following pages to make sure that it reports class-precedence correctly. The input file (data structure) should be manually prepared using the class hierarchy pictures depicted in the sequel. Khan Academy can be consulted about the alternatives:

> https://www.khanacademy.org/computing/computer-science/algorithms/graph-representation/a/representing-graphs

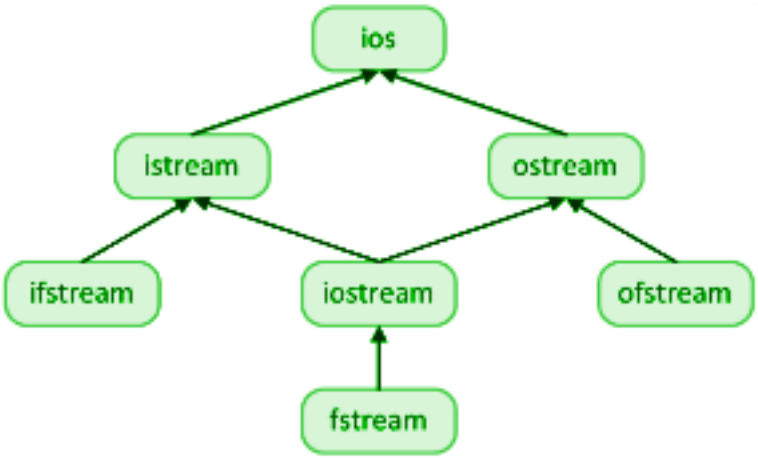
The output of your program should simply be a linear list, with the highest and lowest precedence ends clearly identified. (In test examples 1 and 2, feel free to add, for the sake of observing the convention of chapter 9, the universal class called **Everything**.) Do not forget to submit the entire code (even if it is not yours).

Your program should have a simple control for 'single stepping' (tracing your code) so that you and the TAs can inspect the intermediate stages of the problem-solving process in an incremental fashion.

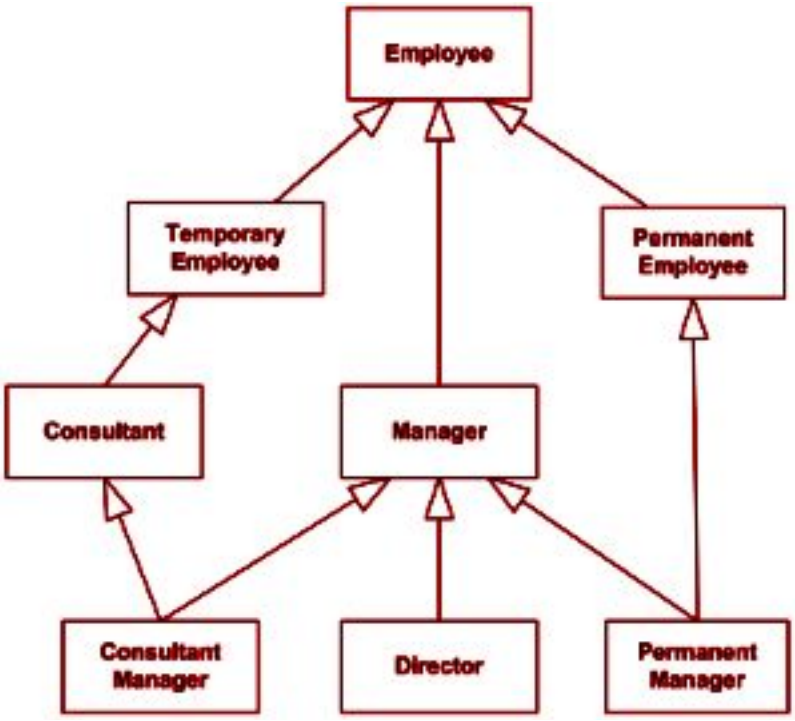
TEST EXAMPLES AND GRADING

The first and the second examples are worth 3 points each. The third example is worth 4 points.

1 (Compute three lists: one for **ifstream**, one for **fstream**, one for **ofstream**)



2 (Compute three lists: one for **Consultant Manager**, one for **Director**, one for **Permanent Manager**)



3 (Compute two lists: one for **Crazy**, one for **Jacque**)