Object-Oriented Programming and Design - Monsoon 2024

Input from shell, assert statements and STL

This assignment carries a total of 60 marks. Q1 carries a total of 25 marks, Q2 carries a total of 15 marks and Q3 carries a total of 20 marks.

1. Suppose you are asked to write a program that needs to keep track of the publications based on research conducted by an institute. Such records are often required by ranking agencies. A single publication consists of the following:

   (a) Title

   (b) Venue of publication

   (c) A number of authors

   (d) Each author has some affiliation. At least one author needs to have the institute's affiliation.

   (e) Optionally a Digital Object Identifier, which gives a URL

   (f) Year of publication

   You need to model the above using the concepts of object-oriented programming. Write a program to design the appropriate data structure.

2. The publication data is often given in the form of a bib file (a sample bib file is attached). Write a program to parse this bib file, and construct the appropriate data structure based on the content. Assume that our institute's affiliation is "IIIT-Delhi". The way the data should be stored should allow an easy search whenever a particular author name's publications are requested. In case an invalid bib file is given, it should raise an assert statement saying it is invalid. Cases of invalid bib files are:

   (a) Formatting mistake, with incorrect braces or commas

   (b) Same author repeated twice in any bib entry

   (c) No author belonging to the institute's affiliation.

   You can separately take the list of faculty from a csv file.

3. Write a different main function that takes the name of any one or more author as an input **via arguments**, and show the list of their publications. Then, calculate and print the average number of co-authors for each paper by this author.

What and How To Submit

- The C++ program sources. Classes and inheritance must be used wherever appropriate.
- The bash script files to create the directory and file structures.
- `Makefile` to compile the sources and generate the running binary for the shell. The Makefile should generate two versions of the binary – one for debugging and another for optimized execution.
- Utilization of multiple files to run the entire program is compulsory.
- A readme text file, explaining the commands needed to build the file, and the format of the input files. If code is copied from anywhere else (not that copying from any other student is plagiarism, but using textbook or open-source code is allowed), that should be mentioned here.
- At least 4 significant commits on a **private** github repository, with proper descriptions of the commits. You may have as many commits as you wish.
- Make the assigned TA the admin of the github repository, **and** submit the same code in zipped form on Google Classroom by the due date.

Late Submission Policy

Late submissions will not be allowed for this assignment.