NATIONAL UNIVERSITY OF COMPUTER & EMERGING SCIENCES ISLAMABAD

Object Oriented Programming (SE)

Spring 2025 ASSIGNMENT # 3

Due Date: 20th April, 2024

CLO-4

Instructions

Submission: Combine all your work in one .zip file. Use proper naming convention for your submission file.

Name the .zip file as SECTION_ROLL-NUM_01.zip (e.g. A_22i0412_01.zip). Your zip file should not contain

any folders or subfolders. It should only contain .cpp files for each question, e.g. Q1.cpp, Q2.cpp, …, Q3.cpp.

Submit .zip file on Google Classroom within the deadline. Failure to submit according to the above format

would result in 25% marks deduction. Submissions on the email will not be accepted.

Plagiarism: Plagiarism cases will be dealt with strictly. If found plagiarized, both the involved parties will be

awarded zero marks in this assignment, all the remaining assignments, or even an F grade in the course.

Copying from the internet is the easiest way to get caught!

Deadline: The deadline to submit the assignment is mentioned above. Late submission with marks

deduction will be accepted according to the course policy shared earlier. Correct and timely submission

of the assignment is the responsibility of every student; hence no relaxation will be given to anyone.

Important Note: Implement the main function for all questions. Evaluation will be done on the

basis of main function for each function.

Comments: Comment on your code properly. Write your name and roll number (as a block comment) at

the beginning of the solution to each problem. Implement the main function for all the functionalities'

main function.

• *You must do proper allocation and deallocation of the memory where necessary.*

• All programs must be generic.

• For timely completion of the assignment, start as early as possible.

Note: Follow the given instructions to the letter, failing to do so will result in a zero.

Q # 1:

(a) Develop a class Histogram for performing histogram calculations (i.e., counting how many values fall in each of a number of intervals). The class should satisfy the following requirements:

i. A constructor should be provided that takes a single argument specifying a std::vector of doubles containing the bounds of the histogram bins. The elements of the std::vector must be strictly monotonically increasing. For example, invoking the constructor with the argument

std::vector<double>{0.0, 3.14, 20.0, 42.42}

would create a histogram with three bins, corresponding to the intervals [0,3.14), [3.14,20), and [20,42.42).

ii. A member function clear should be provided that clears the histogram statistics.

iii. A member function update should be provided that takes a new data value to be added to the histogram statistics. This function should be able to handle out-of-range data in some appropriate manner by normalizing it to make them inside the histogram.

iv. A member function display should be provided that outputs the contents of the histogram to a given output stream (i.e., std::ostream) in some human-readable format.

v. A display function that implements some graphic library to display the contents of histogram is different colors. If we have three bins, then each bin should have separate color likewise if 4 then in 4 separate colors etc.

vi. The class should not be default constructable (i.e., no default constructor should be provided).

vii. The class should provide move and copy constructors, move and copy assignment operators, and a destructor.

viii. All data members should be private.

Q # 2:

Develop a class Integer that behaves similar to the built-in integer type int, except that:

1) the meaning of addition and subtraction are reversed; and

2) the meaning of multiplication and division are reversed.

The Integer class should satisfy the following requirements:

i. A constructor should be provided that takes a single int argument that is used to initialize the
Integer. The argument should default to zero.
ii. The class should provide move and copy constructors, move and copy assignment operators, and a destructor.

iii. The class should overload all of the following operators: addition, subtraction, multiplication, division, +=, -=, *=, and /=.

iv. A stream inserter should be provided to allow an Integer to be written to an output stream (i.e., std::ostream).

v. A stream extractor should be provided to allow an Integer to be read from an input stream (i.e., std::istream).

vi. All data members should be private.

Q # 3:

Design a class Library Management System which includes the following attributes:
· UserType: an int
· UserName: a String
· Password: a String

The class has the following member functions.

1. A constructor initializing the data members withdefault parameters.
2. Getters and Setters of the class data members.
3. Login
4. Register
5. Logout

Design a class Librarian which includes the following attributes:

· Name: a String
· ID: an int
· Password: a String

The class has the following member functions.

1. A constructor initializing the data members withdefault parameters.
2. Getters and Setters of the class data members.
3. Verify_librarian
4. Search_book

Design a class Book which includes the following attributes:

· Title: a String
· Author: a String
· ISBN: a String

The class has the following member functions.

1. A constructor initializing the data members with default parameters.
2. Getters and Setters of the class data members.
3. Show_due_date
4. Reservation_status
5. Book request
6. Renew_info

Design a class Page which includes the following attributes:

· Text: a String
· Page_no: an Int

The class has the following member functions.

1. A constructor initializing the data members with default parameters.
2. Getters and Setters of the class data members.

Design a class Account which includes the following attributes:

· no_borrowed_books: an Int
· no_lost_books: an Int
. no_reserved_books: an Int
· fine_amount: a float
The class has the following member functions.

1. A constructor initializing the data members withdefault parameters.
2. Getters and Setters of the class data members.
3. Calculate_fine

Design a class Library database which includes the following attributes:

· List_of_books: Book

The class has the following member functions.

1. A constructor initializing the data members with default parameters.
2. Getters and Setters of the class data members.
3. Add
4. Delete
5. Update
6. Display
7. Search

Design a class User which includes the following:
· Name: a String
. ID: an int

The class has the following member functions.

1. A constructor initializing the data members with default parameters.
2. Getters and Setters of the class data members.
3. Verify

4. Check_account
5. get_book_info

Design a class Staff which includes the following:
· Dept: a String
. Designation: a String

The class has the following member functions.
1. A constructor initializing the data members with default parameters.
2. Getters and Setters of the class data members.

Design a class Student which includes the following:

· Batch: a String
. Designation: a String

The class has the following member functions.

1. A constructor initializing the data members with default parameters.
2. Getters and Setters of the class data members.

You may add other attributes and functions to the classes if needed.

Now check the following scenario

1. Create a new admin i.e. Library Management System object
2. Admin should create a new Librarian
3. Librarian should be able to Login and add 10 books to the Library database
4. Create 2 students and 1 staff members
5. Student 1 should be able to issue 2 books, student 2 should be able to issue 3 books. Same book can’t be issued twice.
6. Staff member should be able to issue 2 books.
7. Due date for Students should be 14 days and the deadline for staff members should be 16 weeks.
8. Display return date of all the books issued by Student 1.
9. Return the books issued by student 2. Display fine for the book returned late
10. Update the status of returned book & user.