University of Central Punjab

Faculty of Information Technology

Object Oriented Programming Fall 2022

Lab 3	
Topic	classes in C++, access specifiers and methods
Objective	Making students familiarize with classes and their implementations in C++,methods and access Specifiers

Instructions:

- Indent your code.
- Comment your code.
- Use meaningful variable names.
- Plan your code carefully on a piece of paper before you implement it.
- Name of the program should be same as the task name. i.e. the first program should be Task_1.cpp
- Do all the tasks in multiple files(separate interface and implementation).
- **void main() is not allowed. Use int main()**
- **You are not allowed to use system("pause")**
- **You are not allowed to use any built-in functions**
- Implement it in multiple files. Every task will contain three respective files Class.h Class.cpp and task.cpp (main().cpp)
- **You are required to follow the naming conventions as follow:**

    - **Variables:** firstName; (no underscores allowed)
    - **Function:** getName(); (no underscores allowed)
    - **ClassName:** BankAccount (no underscores allowed)

Students are required to complete the following tasks in lab timings.

Task 1:

Create a class named as **Date** having following private attributes:

1. **day(int)**
2. **month(int)**
3. **year(int)**

Now write the following for the above mentioned class:

- Write a function **init** which should receive three integers as parameters (_day, _month and _year) and initialize all the attributes with the respective values received in the parameters.
  Remember: Validate the values in the above function. For eg: month can never be negative etc.
- Write a non-returning **display** function to print the attributes of the class.

Now write a program to create two objects of Date in main(). In one you have to ask user to enter his birthDate and in other the currentDate. Display the birthdate and currentDate of the user by display function. And then calculate age of the user from the objects and print it.

Task 2:

Create a class named as **Employee** having following private attributes:

1. **employeeCode(int)**
2. **ageAtJoining(int)**
3. **currentAge(int)**

Now write the following for the above mentioned class:

- Write a function **constructor** which should receive three integers as parameters (_employeeCode, _ageAtJoining and _currentAge) and initialize all the attributes with the respective values received in the parameters.
  Remember: Validate the values in the above function. For eg: ageAtJoining can never be negative etc.
- Write a non-returning **display** function to print the attributes of the class.
- Write a function **calculateTenure** which calculates the tenure of the employee and returns it.

Write a program to create five objects of employee with different data.
Display the details of those employees whose tenure is 2 or more than 2 years.

Task 3:

Create a class named as **Triangle** having following private attributes:

1. **base (double)**
2. **height(double)**

Now write the following for the above mentioned class:

- Write a function **constructor** which should receive two doubles as parameters (_base and
  _height) and initialize all the attributes with the respective values received in the parameters.
  Remember: Validate the values in the above function. For eg: height can never be negative etc.
- Write a non-returning **display** function to print the attributes of the class.
- Write a function **calculateArea** which calculates the area of the triangle and returns it.

Write a program to create **array of five objects** of **Triangle** with different data. Display the attributes of the triangle through display function having greatest area (use function written in the class to calculate area).

Task 4:

Create a class named as **ComplexNumber** having following private attributes:

1. **realPart(double)**
2. **imaginaryPart(double)**

Now write the following for the above mentioned class:

- Write a function **constructor** which should receive two doubles as parameters (_realPart and
  _imaginaryPart) and initialize all the attributes with the respective values received in the parameters.
  Remember: Validate the values in the above function.
- Write a non-returning **display** function to print the attributes of the class in the format given below.

Write a program to create **array of five objects** of **ComplexNumber** with different data. And then display all the complex number objects in the following format.

Output Format: 3+2i

Task 5:

Using the concept of getters and setters, Create a class with name Employee that contains a

private data member with the name **salary** and take the salary of the employee from the user in the main() function and set that salary into the data member of class and the name of the data member is salary, you have to use a function setSalary(int s); and then get the salary back in the main function using a getter function with a name getSalary();

Task 6:

Use the concept of constructor overloading, write a class shape and you have to set the parameters of circle radius in the first constructor function and set the three sides of triangle using the constructor function.
Sample: **shape (float r);**
Sample: **shape(float base,float height);**
Use a member function **calculateareaofcircle();** to calculate area of a circle and use a member function **calculateareaoftriangle();**

Task 7:

Using the concept of dynamic array of pointer , Make a class with the name **student** containing the information of **science student info.** (roll no, name, chemistry subject marks) and **arts students info.** (roll no. , name , political science subject marks) create the dynamic array of size 2 of class instances and using the 1st instance, you have to enter the information of science subject student using constructor function and using the 2nd instance, you have to enter the information of arts student. Also use a **display();** function to display the information of the science student details and arts students detail.