CMPT 1020 - Assignment # 1

Warning: You must submit three .cpp files as indicated in the following questions. If you submit file(s) with other names or formats, you will receive 0 for this assignment.

Question #1:

Write a Circle class that has the following member variable:

- radius : a double

In your program define a named constant:
pi : a double initialized with the value 3.14159

The class should have the following member functions:

- Default Constructor. A default constructor that sets radius to 0.0.
- Constructor. Accepts the radius of the circle as an argument.
- setRadius. A mutator function for the radius variable.
- getRadius. An accessor function for the radius variable.
- getArea. Returns the area of the circle, which is calculated as

area = pi * radius * radius

- getDiameter. Returns the diameter of the circle, which is calculated as

diameter = radius * 2

- getCircumference. Returns the circumference of the circle, which is calculated as

circumference = 2 * pi * radius

Write a program that demonstrates the Circle class by asking the user for the circle's radius, creating a Circle object, and then reporting the circle's area, diameter, and circumference.

Name the program file circle.cpp

Question #2:

Define a class for complex numbers with two double member variables, real and imaginary.
Your class definition must include accessor, mutators, input, output, and three constructors (one default, one which receives one argument, and one receives two arguments.)
overload +, -(binary and unary) , * , +=, -=, *=, = =. Overloaded operators must be defined as members of the class.

Name the program file complexnum.cpp

Question #3 – Name the program file Q3.cpp

Step 1- Define a class named Student with the following member variables:

- firstName (a string)
- lastName (a string)
- studentID (a string)
- phoneNumber (a string with this format 604-555-5555)
- gpa (a double number)

Member functions include:

- 5 accessor and 5 mutator functions
- the default constructor which sets all string member variables to a string with one space in it (" ") and gpa to 0
- a constructor with all five parameters
- input function
- output function that displays the member variables of a student

No other function is necessary for this class.

Step 2- Define a class named Course with the following member variables:

- code: Course code (a string like CMPT1020)
- section: Section number (an integer)
- capacity: Max number of students (an integer)
- numStudents: Actual number of students (an integer)
- list: An array that stores the info for the students in the course , as a pointer to a Student object (defined in part 1)

Member function include:

- 4 accessor functions (for the first four member variables)
- 2 mutator functions (for code and section)
- Default constructor that sets: code=CMPT1020, section=1, capacity=35, numStudents=0 and allocates memory for list (35 students)
- A constructor that accepts: code, section number, and capacity. It set numStudents=0 and allocate memory for list (capacity students)
- A destructor to free memory
- A function named add which adds a student
- A function named display (to display all students), this function has no input argument

- A function named void displayByFirst(string): that accepts a string for firstName and displays the entire info for the student with given firstName
- A function named void displayByLast(string): that accepts a string for lastName and displays the entire info for the student with given lastName
- A function named void displayByID(string): that accepts a string for studentID and displays the entire info for the student with given studentID
- A function named void displayByPhone(string): that accepts a string for phoneNumber and displays the entire info for the student with given phoneNumber
- A function named remove that accepts one string for phone number, this function removes one of the students with given phone number.
- A function named sortByFirstAsc: sorts the list based on first names in ascending order
- A function named sortByFirstDes: sorts the list based on first names in descending order
- A function named sortByLastAsc: sorts the list based on last names in ascending order
- A function named sortByLastDes: sorts the list based on last names in descending order
- A function named sortByIDAsc: sorts the list based on IDs in ascending order
- A function named sortByIDDes: sorts the list based on IDs in descending order
- A function named sortByPhoneAsc: sorts the list based on phone numbers in ascending order
- A function named sortByPhoneDes: sorts the list based on phone numbers in descending order
- A function named sortByGPAAsc: sorts the list based on GPAs in ascending order
- A function named sortByGPADes: sorts the list based on GPAs in descending order

Step 3- Write a main function to define a Course object and test the functions you defined in the previous steps.