

Assignment 5

1. Circle Class

Write a Circle class that has the following member variables:

- `radius`: a double
- `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 $\text{area} = \text{pi} * \text{radius} * \text{radius}$
- **getDiameter**. Returns the diameter of the circle, which is calculated as $\text{diameter} = \text{radius} * 2$
- **getCircumference**. Returns the circumference of the circle, which is calculated as $\text{circumference} = 2 * \text{pi} * \text{radius}$

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.

2. Coin Toss Simulator

Write a class named Coin. The Coin class should have the following member variable:

- A string named `sideUp`. The `sideUp` member variable will hold either "heads" or "tails" indicating the side of the coin that is facing up.

The Coin class should have the following member functions:

- A default constructor that randomly determines the side of the coin that is facing up ("heads" or "tails") and initializes the `sideUp` member variable accordingly.
- A void member function named `toss` that simulates the tossing of the coin. When the `toss` member function is called, it randomly determines the side of the coin that is facing up ("heads" or "tails") and sets the `sideUp` member variable accordingly.
- A member function named `getSideUp` that returns the value of the `sideUp` member variable.

Write a program that demonstrates the Coin class. The program should create an instance of the class and display the side that is initially facing up. Then, use a loop to toss the coin 20 times. Each time the coin is tossed, display the side that is facing up. The program should keep count of the number of times heads is facing up and the number of times tails is facing up, and display those values after the loop finishes.

3. Freezing and Boiling Points

The following table lists the freezing and boiling points of several substances.

Substance	Freezing Point	Boiling Point
Ethyl Alcohol	-173	172
Oxygen	-362	-306
Water	32	212

Design a class that stores a temperature in a temperature member variable and has the appropriate accessor and mutator functions. In addition to appropriate constructors, the class should have the following member functions:

- **isEthylFreezing**. This function should return the bool value true if the temperature stored in the temperature field is at or below the freezing point of ethyl alcohol. Otherwise, the function should return false.
- **isEthylBoiling**. This function should return the bool value true if the temperature stored in the temperature field is at or above the boiling point of ethyl alcohol. Otherwise, the function should return false.
- **isOxygenFreezing**. This function should return the bool value true if the temperature stored in the temperature field is at or below the freezing point of oxygen. Otherwise, the function should return false.
- **isOxygenBoiling**. This function should return the bool value true if the temperature stored in the temperature field is at or above the boiling point of oxygen. Otherwise, the function should return false.
- **isWaterFreezing**. This function should return the bool value true if the temperature stored in the temperature field is at or below the freezing point of water. Otherwise, the function should return false.
- **isWaterBoiling**. This function should return the bool value true if the temperature stored in the temperature field is at or above the boiling point of water. Otherwise, the function should return false.

Write a program that demonstrates the class. The program should ask the user to enter a temperature and then display a list of the substances that will freeze at that temperature and those that will boil at that temperature. For example, if the temperature is -20 the class should report that water will freeze and oxygen will boil at that temperature.

4. Trivia Game

In this programming challenge you will create a simple trivia game for two players. The program will work like this:

- Starting with player 1, each player gets a turn at answering five trivia questions. (There are a total of 10 questions.) When a question is displayed, four possible answers are also displayed. Only one of the answers is correct, and if the player selects the correct answer he or she earns a point.
- After answers have been selected for all of the questions, the program displays the number of points earned by each player and declares the player with the highest number of points the winner.

In this program you will design a `Question` class to hold the data for a trivia question. The `Question` class should have member variables for the following data:

- A trivia question
- Possible answer #1
- Possible answer #2
- Possible answer #3
- Possible answer #4
- The number of the correct answer (1, 2, 3, or 4)

The `Question` class should have appropriate constructor(s), accessor, and mutator functions.

The program should create an array of 10 `Question` objects, one for each trivia question. Make up your own trivia questions on the subject or subjects of your choice for the objects.