**The following source codes are provided to you:**

The body of the Animal, Bird, Fish, and AnimalRescueCenter class.

The Application class which contains as a member a static instance of the class AnimalRescueCenter. the static displayMenu() method is already implemented. You are not allowed to modify the body of the displayMenu() method. There are also 4 static methods declared and defined without source code. In step 2 of the exam, you will need to complete the source code for these methods later during the exam.

**Technical guidelines and specifications**

**Step 1 - Creating the classes:**

You need to create 4 classes:

You must create the Animal, Bird and Fish classes. You must also complete the source code for the AnimalRescueCenter class that was provided to you before the exam.

An Animal class which contains:

Animal identifier (integer numeric value)

The name of the animal of type string or array of char (your choice).

The type of animal (of a numerical value or enumeration value, your choice).

A variable that contains information that indicates the status of whether the animal has been adopted or not.

At least one constructor, to initialize each of the data members via the formal parameters, except for the data member containing the adopted status of the animal, which will be automatically set to not adopted (false).

A constant getter for each of the data members.

One setter only for the data member of the adopted status.

A virtual constant display method that displays all data member values.

A Fish class which inherits from the Animal class and contains:

At least one parameterized constructor to initialize the data members of the inherited class.

A move() method that displays a message following the format below. The name displayed is the value found in the data member of the Animal base class:

name the fish swim in the aquarium!

A Bird class which inherits from the Animal class and contains:

The wingspan of the bird.

At least one parameterized constructor to initialize the data member of the class as well as the data members of the inherited class.

A redefinition of the Animal display method to display Animal and Bird member data.

A crawl() method that displays a message following the format below. The name displayed is the value found in the data member of the Animal base class:

name the bird fly in the bird sanctuary!

A **AnimalRescueCenter**class containing:

A **static constant** data member that should contain the size of the Animal array, i.e. 25.

An array of 25 Animal . Use the constant created previously.

The current number of animals in the array.

A constructor without parameters only. The constructor structure is already provided in the **AnimalRescueCenter** class, you only need to complete the source code to initialize the Animal Rescue Center without any animal at the start.

A method named **addAnimal()** that will add an Animal created with dynamic allocation (new) in the Animal Rescue Center. The form for entering a new Animal must be either in this method or in a static method of the Animal class (you are allowed to add this method in the Animal class if you want). You must take into consideration that the maximum size of the array is 25.

A method named **searchAvailableAnimal()** must be declared as private. This method is used to search among animals that have not yet been adopted in the Animal Rescue Center according to an identifier that will be entered by a user following a dialogue with him in this method. The method will return**the index where this Animal is in the array, or -1 if it is not there.**

A method named **giveAnAnimalToAdoption()** that will call the **searchAvailableAnimal()**method to ask the user to enter the identifier of the animal to adopt. If the animal is available, modify the state of its data member to indicate it is adopted and display the complete information of this Animal. If no animal were found, display a message indicating that the animal is not available.

A method named **displayAllAvailableAnimals()** that need to displays the information of all the animals in the Animal Rescue Center that have not been adopted yet.

A method named **makeTheAdoptedAnimalsMove()** that will make the fish swim and the birds flies. Only animals that have been adopted will move.

**Step 2 -** Complete the source code of the methods of the Application class

In the Application class provided to you, you must complete the source code for the following 4 methods:

The **saveAnAnimal**method which:

Call the Animal Rescue Center's **addAnimal()** method.

A **adoptAnAnimal**method which:

Call the **giveAnAnimalToAdoption()** method of the Animal Rescue Center'.

A **displayAnimals** method which:

Calls the **displayAllAvailableAnimals()** method.

A **makeTheAnimalsMove**method which:

Call the appropriate method for making the animal that has been adopted move.