Project #4 **Due Date:**

Write a program that creates a class hierarchy that stores information about airships. Start with a general base class called **airship** that stores the number of passengers that can be carried and the amount of cargo (in pounds) that can be carried. Then create two derived classes called **airplane** and **balloon** from airship. Have airplane store the type of engine used (propeller or jet) and range, in miles; also give airplane a name. Have balloon store information about the type of gas used to lift the balloon (hydrogen or helium) and its maximum altitude (in feet). Also give balloon a name.

This program **prints a report** of airship data. It reads the data from the file "**airship.dat**". Each line of data consists of a comma-separated list. An airship is either an airplane or a balloon. The first field in each line of input determines whether the line describes an airplane or a balloon.

The **airplane** data has these fields:
airshipType, airplaneName, maxPassengerCount, maxCargoWeight, engineType, range

The **balloon** data has these fields:
airshipType, balloonName, maxPassengerCount, maxCargoWeight, gasType, maxAltitude;

The program reads each line of input and checks the airship type for the data. If the line describes an airplane, the program creates an airplane object, and if the line describes a balloon, the program creates a balloon object. The program places the airplane and balloon objects in an array of pointer type airship.

The program then iterates through the array once to generate an airplane report and once to create a balloon report. You must demonstrate the usage of **pure virtual** function and show run time **polymorphism** in this program. There is no need for user interface. Your program should take a command line argument for the input file name.

Your program should read data from the file. Data file provides five records for airplane and five records for balloon.

Data for Project #4

Use any text editor to enter the following data into a "airship.dat" file.

```
1,Blue Racer,10,10000,0,1000
0,SR-71 Blackbird,2,33000,0,80000
1,Desert Winds,20,20000,1,2000
0,U-2,4,22000,0,160000
1,Kaleidoscope,30,30000,1,3000
1,Desert Spirit,40,40000,0,4000
0,Boeing 747,300,11000,0,40000
0,McDonnel Douglas,225,11000,0,38000
0,Cessna 120,6,11000,1,10000
1,Painted Angel,50,50000,0,5000
```

Data fields Names

AirShipType {Airplane=0,Ballon=1}
AirShipName, NumberOfPassengers,AmountOfCargo
{Engine Type [Jet = 0, Propeller = 1] Gas Type [Helium = 0, Hydrogen = 1]}
Maximum Range/Altitude

Output for Project #4

Use following output format:

```
Listing of all Airplanes
Name                Engine Type     Maximum Range
-------------------------------------------------
SR-71 Blackbird     Jet             80000
U-2                 Jet             160000
Boeing 747          Jet             40000
McDonnell Douglas   Jet             38000
Cessna 120          Propeller       10000


Listing of all Balloons
Name                Gas Type        Maximum Altitude
-----------------------------------------------------
Blue Racer          helium          1000
Desert Winds        hydrogen        2000
Kaleidoscope        hydrogen        3000
Desert Spirit       helium          4000
Painted Angel       helium          5000
```

Project #4 Grading Requirements

Category	*Points Possible*	Points Received
Use of **pure virtual** functions	20	
Use of enum type with a meaningful names.	10	
Command line argument for input file	10	
Output format according to the specifications	15	
Error Checking (User-friendliness)	15	
Complete Source Code Documentation	10	
Correctness and Efficiency	10	
Meaningful variable names.	10	
Total	**100**	

Things to Remember

- Late program turn-in will have a 10% penalty per week.
- This program will not be accepted one week after the end of the current class session.
- Make sure that you do appropriate error checking. (User-friendliness)
- Do not turn in an incomplete or crashing program.