

Lab Exam 1:

BEFORE YOU GET STARTED:

- Please read through Questions 1-3 before you start coding.
- The input for this file will come from the “inputfile.txt”
- For questions 1 and 2, I give the MINIMUM requirements. You are free to add any additional functions and variables that you need.
- For question 3, refer to my samplemain.cpp and sampleoutput.txt for what I expect to see.
- Keep in mind, Questions 1,2, and 3 are VERY similar to labs 1 and 2.
- You're free to use your past homework and labs. The internet is also allowed.

Question 1 (40%):

Create an **Animal** class.

Minimum requirements for the Animal class:

- A default constructor.
- 3 Private Variables: **species, color, number of legs**
- Overloaded << operator that outputs the species, color and number of legs.

Recommended:

- Functions to get and set the private variables.
- Overloaded comparison operator (for sorting).
- Overload your Constructor to initialize the private variables.

Question 2 (35%):

Create a **Zoo** class.

Minimum requirements for the Zoo class:

- A default constructor (It can be blank if you don't need to initialize anything)
- A **vector OR array** of Animal objects as a PRIVATE variable.
- A **function to insert (append, add)** an Animal object to the end of the vector/array.
- A **function to delete (remove)** an Animal object at the end of the vector/array.
- A **function to sort** the vector/array by number of legs for each animal (It doesn't matter if you sort from least to greatest or greatest to least)

Recommended:

- A function that takes either a stream or string as an input that returns an Animal object. This will help when you read from the input file and store the Animal object into the vector/array.
- A function that can get rid of dummy lines.
- Write a function that can print all the Animal objects in the array to a file. (Either overload the << operator or write a function).

Question 3 (10%):

Write your **main()** function.

- Declare a Zoo object in your main().
- Read from the file “importfile.txt,” and store each line into the vector/array of your Zoo object. Print the results to an output file. (Your code must be able to handle the dummy line).
- Append your favorite animal to the vector/array in the Zoo object. Print the results to an output file.
- Remove the last element from the vector/array in the Zoo object. Print the results to an output file.
- Sort the elements in the vector/array in the Zoo object. Print the results to an output file.

Finish the Questions 1-3 before working on the questions below

Question 4 (5%):

Find the **median** in the vector/array from your Zoo class based on the number of legs of the Animal objects.

- Write a function in your Zoo class that finds the median. Call the function in your main. Print the result to an output file.

Question 5 (5%):

Find all the Animals in the vector/array in your Zoo class that have **less than 4 legs**.

Hint: This is easier if you write it with 1 function that takes ofstream as a parameter.

- Write a function in your Zoo class that finds all the animals with less than 4 legs, call the function in your main and print it to an output file.

Question 6 (5%):

Sort the animals in the vector/array in your Zoo class **alphabetically**.

Hint: There is a reason I made sure every animal began with a different letter.

Hint: Look up Ascii code. Notice that the character 'A' is less than 'B'

- Write a separate sort function in your Zoo that sorts the Animals in the vector/array alphabetically.
- Call this function in your main and print the results to an output file.