## **CSE 240 Homework 9: Class and Input/Output**

## **Fall 2016 (50 points)**

Due Saturday, October 22, 2016 at 11:59PM, plus a 24-Hour grace period

### Introduction

The aim of this assignment is to make sure that you understand and are familiar with the concepts covered in the lectures and input/output. By the end of the assignment, you should have understood

* Class, data members and function members
* Classes and header files
* Memory management and garbage collection
* C++ input and output

**Reading**: Textbook Chapter 3, Sections 3.1, 3.2, and 3.3 on C++ examples, class definition and memory management.

**Preparation**: Complete the multiple choice questions in the textbook exercise section. The answer keys can be found in the course Web site. These exercises can help you prepare for your weekly quiz and the exam. You are encourage to read the other exercise questions and make sure you understand these questions in the textbook exercise section, which can help you better understand what materials are expected to understand after the lectures and homework on each chapter.

You are expected to do the majority of the assignment outside the class meetings. Should you need assistance, or have questions about the assignment, please contact the instructor or the TA during their office hours.

You are encouraged to ask and answer questions on the course discussion board. However, **do not share your answers and code** in the course discussion board.

### Programming Assignment (50 points)

You are given a partially completed project containing:

3 header files:

Container.h

Pet.h

Checkup.h

4 C++ files:

Container.cpp

Pet.cpp

Checkup.cpp

hw09.cpp

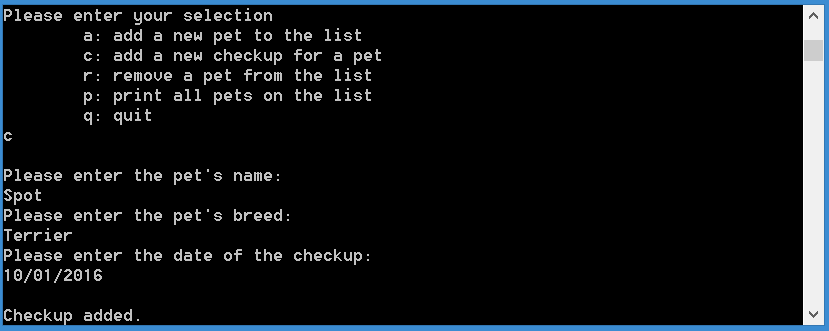
Your job is to follow the instructions given in the comments of the hw09.cpp and Pet.cpp files to complete the missing parts of the project so that the program executes properly.

### Q1: Constructor and Accessor Methods for Pet class (5 points)

You will need to write the constructor and accessor methods for the Pet class in the Pet.cpp file. The program will not compile until this part is completed. The constructor and accessor methods are already declared in the Pet.h file. *(See Pet.cpp file for details).*

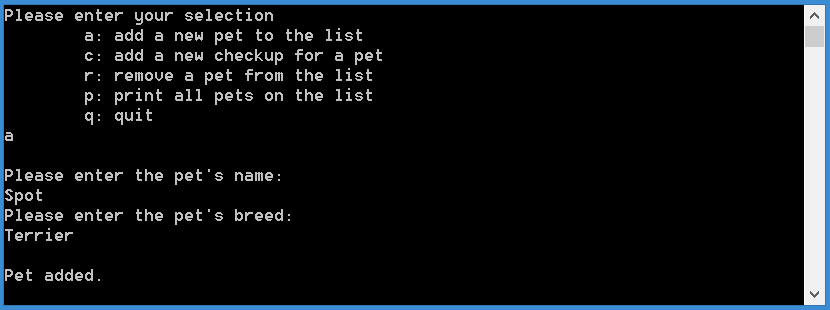
### Q2: Add Checkup and Last Checkup Methods for Pet class (10 points)

You will need to write these methods for the Pet class in the Pet.cpp file. The program will not compile until this part is completed. These methods are already declared in the Pet.h file. *(See Pet.cpp file for further instructions).*



### Q3: Add Pet Function (5 points)

When inputting a new pet, the user is prompted for the pet’s name and breed. This function will be used to add a new pet to the head of the global linked list of containers. This means that no sorting is needed for this function. You will notice that the Search Pet function is called before this function, therefore, you are to assume that the Pet that you are looking for is not already on the list. *(See hw09.cpp file for further instructions).*

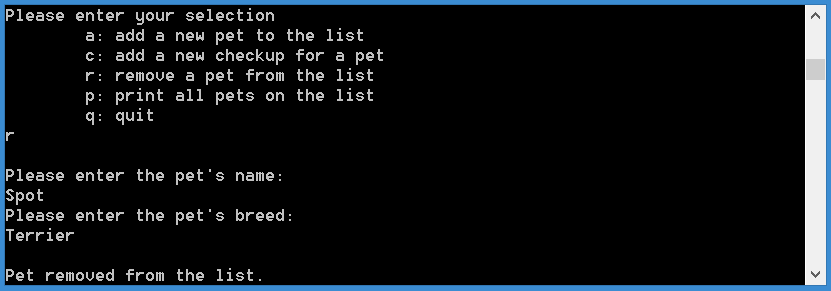


### Q4: Search Pet Function (10 points)

This function is used to check if a Pet exists in the global list of Containers. It is called before executing most of the other functionalities of this assignment. In order to find a Pet on the list, you will need to check both the name and the breed of the Pet. Pets on the list may have the same name OR the same breed. Pets on the list may not have the same name AND the same breed. *(See hw09.cpp file for further instructions).*

### Q5: Remove Pet Function (15 points)

When removing a Pet from the list, the user is prompted for the pet’s name and breed. You will use these 2 string parameters to search for the Pet on the list and remove them. You will need to do so in a way that ensures no memory leaks. You will notice that the Search Pet function is called before this function, therefore, you are to assume that the Pet that you are looking for is already on the list. *(See hw09.cpp file for further instructions).*

**

### Q6: Implement cin / cout for the 7 lines in main (5 points)

You will notice that main in hw09.cpp uses printf statements and getchar(). Update these statements to use cin and cout without changing the functionality of the statements.

### Grading of Programming Assignment

The TA will grade your program following these steps:

(1) Compile the code. If it does not compile, 20% of the points given will be deducted. For example, if there are 20 points possible, you will earn 16 points if the program fails to compile.

(2) The TA will read your program and give points based on the points allocated to each component, the readability of your code (organization of the code and comments), logic, inclusion of the required functions, and correctness of the implementations of each function.

### What to Submit?

You are required to submit your solution in a compressed format (.zip). Make sure your compressed file is label correctly - lastname\_firstname9.zip. (All lowercase, do not put anything else in the name like "hw9".)

The compressed file MUST contain the following:

hw09.cpp

Pet.cpp

No other files should be in the compressed folder.

If multiple submissions are made, the most recent submission will be graded. (Even if the assignment is submitted late.)

### Where to Submit?

All submissions must be electronically submitted to the respected homework link in the course web page where you downloaded the assignment.

### Late submission deduction policy

* No penalty for late submissions that are received within 24 hours after the deadline;
* 10% grade deduction for every day it is late after the grace period;
* No late submission after Tuesday at 11:59PM.