CPE 101
Fall 2009
Laboratory 8
Due Date

- **By Friday of week ten**
- **You must turn in your source electronically on vogon using the handin command – instructions are provided in below.**

Objectives

- To practice creating and using records/structures in C.
- To practice using and creating arrays of records.
- To practice using FILE I/O.
- To practice developing functions to work with array data.
- To develop a complete a C program, compile it, and turn it in electronically.

Resources

- **This is a pair programming assignment (you may select your own partners)**
- Your instructor, peers, texts, and your own innate capabilities and resourcefulness!

Orientation…

This lab will involve developing a small C program, which handles data about the zoo. You will be creating your own type, a record/structure, in order to store the animal information used by the zoo. You will need to develop a variety of functions to process this data.

Part 0: Creating the data type and reading in data…

Start out by defining a structure to store animal data. This structure must include:

The type of animal (80 char maximum)

The number of that type of animals currently in the zoo

You will also be writing a variety of functions, which operate on the animal data. You will read an initial zoo database from file, however, every day animals are born and die or are transferred to the zoo and the population will change and your program data must reflect this. Note that the zoo at most will ever only have a total of 30 different types of animals. Write the appropriate code to parse the zoo data from an input file and store it.

Write a function to print out the zoo data. The initial input should print out as follows:

```
animal 0:
        type: penguin
        population: 10
animal 1:
        type: elephant
        population: 2
animal 2:
        type: lion
        population: 2
animal 3:
```

```
        type: monkey
        population: 4
animal 4:
        type: lemur
        population: 3
animal 5:
        type: turtle
        population: 7
animal 6:
        type: gorilla
        population: 2
animal 7:
        type: otter
        population: 2
animal 8:
        type: fish
        population: 42
animal 9:
        type: parrot
        population: 12
.
```

Next you will be developing several functions to update the zoo population. Start by adding a dialogue loop to prompt the user to either update the zoo data or exit (a selection of 1 means the user wants to update the zoo database, while a 0 means the user wants to exit the program). If the user wants to edit the database, your program should prompt for the type of animal and then change its population. You must write code to handle multiple kinds of changes to the population:

1) either an addition or loss in the population,
2) or the addition of a new animal type
3) or the loss of the entire type of a given animal population.

In order to handle these changes you must write code to locate if that type of animal is currently in the zoo. If the animal is in the zoo then update the population. If the animal is not in the zoo, then you must add it to the database. If the current population of the animal falls to zero, you must remove that type of animal from the database. The below example run shows all of the above mentioned population changes. In order to handle the above population changes, please write the following functions

1) A function which finds a given animal in the zoo database: It should take in the array of zoo data, the current number of animals in the zoo, the animal to search for and return the index location where the animal is found. The function should set the index location of where the given type of animal is located in the zoo array or a -1 if the animal was not found.

2) A function to insert a new animal into the zoo database. The input should be the array of zoo data, the current number of animals in the zoo (but this value will change in this function, so it likewise must be what the function returns, i.e. an updated zoo population), the type of animal to add to the zoo and the population of that animal type. For example:

int InsertAnim(animD a[], int size, char in_anim[], int in_pop)

3) A function to update the zoo data. This function, should take in the array of zoo data, the current number of animals in the zoo, the index of the animal you should update and the change in the population of that type of animal. Note that this function must detect if the zoo population for the specified animal falls below zero. If so then you need to call the ShiftArray function described next.

4) Write a function, which will shift all the elements in the array of zoo data from a given index to the next position. This function will be used if a certain type of animal has no more of its type in the zoo (i.e. the number of a given type of animal falls to 0). This function should take in the array of zoo data, the current number of animals in the zoo and the index where the shift should begin. Note that again the current number of animals in the zoo should change after the shift thus use an appropriate function protype to complete this task.

Each time the database is updated, please print it out again. The following is an example run. The user input is underlined for illustration only. In addition, some text is highlighted and made bold to emphasize the change in the zoo population (you are not expected to re-create this in your output).

```
$ a.exe
animal 0:
        type: penguin
        population: 10
animal 1:
        type: elephant
        population: 2
animal 2:
        type: lion
        population: 2
animal 3:
        type: monkey
        population: 4
animal 4:
        type: lemur
        population: 3
animal 5:
        type: turtle
        population: 7
animal 6:
        type: gorilla
        population: 2
animal 7:
        type: otter
        population: 2
animal 8:
        type: fish
        population: 42
animal 9:
        type: parrot
        population: 12
```

```
      To update the zoo database, enter 1, otherwise enter 0: 1
      Enter the type of animal: otter
      Enter the change in the population as a numeric value (e.g. 5 or
-5): 1
      animal 0:
              type: penguin
              population: 10
      animal 1:
              type: elephant
              population: 2
      animal 2:
              type: lion
              population: 2
      animal 3:
              type: monkey
              population: 4
      animal 4:
              type: lemur
              population: 3
      animal 5:
              type: turtle
              population: 7
      animal 6:
              type: gorilla
              population: 2
      animal 7:
              type: otter
              population: 3
      animal 8:
              type: fish
              population: 42
      animal 9:
              type: parrot
              population: 12
      To update the zoo database, enter 1, otherwise enter 0: 1
      Enter the type of animal: fish
      Enter the change in the population as a numeric value (e.g. 5 or
-5): -12
      animal 0:
              type: penguin
              population: 10
      animal 1:
              type: elephant
              population: 2
      animal 2:
              type: lion
              population: 2
      animal 3:
              type: monkey
              population: 4
      animal 4:
              type: lemur
              population: 3
      animal 5:
              type: turtle
              population: 7
      animal 6:
```

```
                type: gorilla
                population: 2
        animal 7:
                type: otter
                population: 3
        animal 8:
                type: fish
                population: 30
        animal 9:
                type: parrot
                population: 12
        To update the zoo database, enter 1, otherwise enter 0: 1
        Enter the type of animal: turtle
        Enter the change in the population as a numeric value (e.g. 5 or
-5): -7
        All of the animals of type turtle have died.
        Removing them from the zoo database
        animal 0:
                type: penguin
                population: 10
        animal 1:
                type: elephant
                population: 2
        animal 2:
                type: lion
                population: 2
        animal 3:
                type: monkey
                population: 4
        animal 4:
                type: lemur
                population: 3
        animal 5:
                type: gorilla
                population: 2
        animal 6:
                type: otter
                population: 3
        animal 7:
                type: fish
                population: 30
        animal 8:
                type: parrot
                population: 12
        To update the zoo database, enter 1, otherwise enter 0: 1
        Enter the type of animal: capybara
        Enter the change in the population as a numeric value (e.g. 5 or
-5): 1
        Inserting new animal capybara
        animal 0:
                type: penguin
                population: 10
        animal 1:
                type: elephant
                population: 2
        animal 2:
                type: lion
```

```
        population: 2
animal 3:
        type: monkey
        population: 4
animal 4:
        type: lemur
        population: 3
animal 5:
        type: gorilla
        population: 2
animal 6:
        type: otter
        population: 3
animal 7:
        type: fish
        population: 30
animal 8:
        type: parrot
        population: 12
animal 9:
        type: capybara
        population: 1
To update the zoo database, enter 1, otherwise enter 0: 0
```

Part 3: Handing in Your Source Electronically…

1. Be sure that yours and your partner's names appear in the header of your submitted lab assignment!!!
2. Log on to vogon using the Secure Shell Client program (or your favorite equivalent).
3. Change directory (cd-command) to the directory containing the source file or files to hand in.
4. Execute the following command –

   `handin zwood csc101lab08 lab08.c`
5. You should see messages that indicate handin occurred without error. You can (and should) always verify what has been handed in by executing the following command:

   `handin zwood csc101lab08`