CS 215 – Study Questions 3
C++ structs and Sorting

1. What is an istream? How did we use an istream in the first part of Lab 3?

2. Describe the purpose/function of the loop:
   ```cpp
   while (input >> variable)
   {
       // do something with variable
   }
   ```

3. We created a function called isvowel for this lab. What is the purpose of this function? Why did we not need to create functions for isalpha and isdigit? Why did we not need a function called isconsonant?

4. What is a struct?

5. What is a constructor?

6. The array
   ```cpp
   string words [100];
   ```
   is a static array. What does this mean?

7. What is sorting?

8. The Swap function has 2 variables. Each variable name is proceeded by an &. What does the & indicate? Why does the programmer need to use the & for this Swap?

9. In this lab, we implemented a version of the insertion sort. Describe the insertion sort algorithm.

10. In addition to insertion sort, what other sorts do you know of?

11. We implemented the insertion sort to sort in ascending order. How would you modify the insertion sort to sort in descending order?

12. What is the output of the statement:
   ```cpp
   cout << left << setw(20) << “Hello World!”
   << right << setw (5) << 25 << endl;
   ```
   (use an underscore (_ _) to indicate a space.)
Use the following struct for a fraction type to answer questions 13 – 16.

```
struct fraction
{
    fraction ();
    int whole;
    unsigned numerator;
    unsigned denominator;
};
```

13. Write the implementation of the constructor for fraction:

```
fraction::fraction ()
{
}
```

14. Given the following array of fraction objects, write a C++ loop that outputs to the console the whole number portion of the first 15 fractions in the array.

```
fraction somefractions[20];
```

15. Given the following array of fraction objects, write a C++ loop that calculates the product of all 20 fractions in the array. Store the product in the fraction variable called product.

```
fraction somefractions[20];
fraction product;
```

16. Given the following array of fraction objects, write a C++ loop that calculates the sum of all 20 fractions in the array. Store the sum of the fractions in the fraction variable called sum.

```
fraction somefractions[20];
fraction sum;
```