

Name: _____ Score: _____ / 100

There are 113 points possible. Once I have everything graded, I'll send everyone a grade report. Check your school email (or whichever one you've been getting emails from me) – it's important that you let me know about any errors in the gradebook ASAP.

1. **(11 points)** Suppose you are given the Java class shown below. Finish the constructor and make an iterator which will walk through all values in the array. **(3 points)** In a separate section of code, show me how to use your iterator to print all the values *and* add the total of their mValues. You can write your Iterator class inside the Database class (just draw arrow to indicate where it would go) or outside (your choice)

```
public class Database
{
    public class Record
    {
        protected String mName;
        protected float mValue;
        public Record(String n, float v) { mName = n; mValue = v; }
    }

    protected Record[] mData;

    public Database(int initial_size)
    {

    }

}
```

2. **(15 points)** Write a generic singly-linked list class (i.e. There is no mPrev in the nodes, only mNext). I only need the constructor and two methods: addToEnd and addToBegin.

5. **(12 points, 2 each)** What are the big-O running times of each of these (you don't have to describe it in words, but if you do, I might be able to give you partial credit if you get the big-O part wrong):
- A linked list method to add after a node with a specific value.
 - A HashSet's contains method
 - Binary Search Tree's iterator
 - The brute-force circle-circle hit detection we used in the QuadTree lab.
 - A Binary Heap (this is the one we implemented) add method.
 - Bubble-sort
6. **(12 points)** Discuss (with pseudo-code and/or *detailed* English) our two methods we discussed in class for handling collisions in a HashMap/HashSet. In your answer, you should describe what a collision is and what causes it. Also discuss how we handle duplicate values (in the HashMap, this was the key; in the HashSet, it was just the value itself).

7. **(11 points)** Give me pseudo-code for the Heap's add method.
8. **(8 points)** How are DFS, BFS similar / different? You could answer this in English or you could give me pseudo-code for both and highlight the differences.
9. **(8 points)** Suppose you want to store the decimal value 426 in a Java integer type called x. What is the smallest type we could use and what would the binary representation look like? Finally, what would the result (in decimal) be if did this:
`x = x << 1;`
10. **(7 points)** Write a function called **Fibonacci** that takes one parameter (inti , which is greater than or equal to 0) and returns the i'th number in the Fibonacci sequence:
1, 1, 2, 3, 5, 8, 13, 21, ...
Each term is the sum of the previous two elements. To get full points, do this recursively.