Arushi Sahai
as5976
COMS W3137
February 10, 2020

Honors Data Structures and Algorithms HW 1 Written Assignment

1. **(6 pts) Weiss, Exercise 2.1**
   Order the following functions by growth rate: *N*, √*N*, *N*1.5, *N*2, *N*log*N*, *N* log log *N*, *N* log2 *N*, *N* log(*N*2), 2/*N*, 2*N*, 2*N*/2, 37, *N*2 log *N*, *N*3. Indicate which functions grow at the same rate.
   2/N < 37 < √N < N < Nlog(logN) < NlogN == Nlog(N^2) < Nlog^{2}N < $N^{1.5}$ < N^2 < N^2logN < N^3 < $2^{N/2}$ == 2^N.

2. **(6 pts) Weiss, Exercise 2.6**
   In a recent court case, a judge cited a city for contempt and ordered a fine of $2 for the first day. Each subsequent day, until the city followed the judge's order, the fine was squared (that is, the fine progressed as follows: $2, $4, $16, $256, $65, 536, . . .).
   a. **What would be the fine on day *N*?**
      $2^{2^\wedge N-1}$.
   b. **How many days would it take the fine to reach *D* dollars? (A Big-Oh answer will do.)**
      O(log(logD))) days.

3. **(6 pts) Weiss, Exercise 2.7 part a (the rest of this problem is part of the programming section)**
   For each of the following six program fragments:
   a. Give an analysis of the running time (Big-Oh will do).

```
(1)  sum = 0;
   for( i = 0; i < n; i++ )
         sum++;
```

O(N).

```
(2)  sum = 0;
   for( i = 0; i < n; i++ )
      for( j = 0; j < n; j++ )
         sum++;
```

O(N^2).

```
(3)  sum = 0;
```

```
   for( i = 0; i < n; i++ )
       for( j = 0; j < n * n; j++ )
           sum++;
```

O(N^3).

```
  (4)  sum = 0;
   for( i = 0; i < n; i++ )
       for( j = 0; j < i; j++ )
           sum++;
```

O(N^2).

```
(5)  sum=0;
       for( i = 0; i < n; i++ )
            for( j = 0; j < i * i; j++ )
                 for( k = 0; k < j; k++ )
                      sum++;
```

O(N^5).

```
(6)  sum=0;
for( i = 1; i < n; i++ )
       for( j = 1; j < i * i; j++ )
          if( j % i == 0 )
              for( k = 0; k < j; k++ )
                  sum++;
```

O(N^4).

4. **(6 pts) Weiss, Exercise 2.10 (part a)**

   Determine, for the typical algorithms that you use to perform calculations by hand, the running time to do the following:

   a. Add two *N*-digit integers.

   O(N).

5. **(8 pts) Weiss, Exercise 2.11**

   An algorithm takes 0.5 ms for input size 100. How long will it take for input size 500 if the running time is the following (assume low-order terms are negligible):

   a. Linear

   2.5 ms

   b. *O*(*N*log*N*)

   0.5 ms = 100log100 * x ; x = 0.0025.

   3.37 ms = 500log500 * 0.0025 ; answer: 3.37 ms.

   c. quadratic

   0.5 ms = 100^2 * x ; x = 0.00005.

12.5 ms = $500^2 * 0.00005$; answer: 12.5 ms.

d. Cubic

0.5 ms = $100^3 * x$; $x = 0.0000005$.

62.5 ms = $500^3 * 0.0000005$; answer: 62.5 ms.

6. **(10 pts) Weiss, Exercise 2.15**

Give an efficient algorithm to determine if there exists an integer *i* such that $A_i = i$ in array of integers $A_1 < A_2 < A_3 < \cdots < A_N$. What's the running time of your algorithm?

One possible algorithm which is O(logN) is as follows:

Find the index of the midpoint of the array. Check to see if the index matches the element at that index. If it does, then there does exist such an integer; return true. If the element is < the index, then find the index of the midpoint of the subarray from the midpoint to the end. If the element is > the index, find the index of the midpoint of the subarray from the beginning to the midpoint. Repeat the check and keep cutting down the array by half until either an integer i is found or the elements to check in the array run out.

7. **(8 pts) Weiss Exercise 3.1**

You are given a list, L, and another list, P, containing integers sorted in ascending order. The operation printLots(L,P) will print the elements in L that are in positions specified by P. For instance, if P = 1, 3, 4, 6, the elements in positions 1, 3, 4, and 6 in L are printed. Write the procedure printLots(L,P). You may use only the public Collections API container operations. What is the running time of your procedure?

The runtime is O(N).

Testing: the program below, when run, should give the output “10 30 40”.

```
import java.util.*;

public class WeissExercise3.1 {

        public static void main(String[] args) {
                createTestLists();
                printLots(L, P);
        }

        private static void createTestLists() {
                L.add(1);
                L.add(10);
                L.add(20);
                L.add(30);
                L.add(40);
                L.add(50);
                P.add(1);
                P.add(3);
                P.add(4);
                P.add(6);
        }

        public static void printLots(List<Integer> L, List<Integer> P) {
                Iterator<Integer> iteratorL = L.iterator();
                Iterator<Integer> iteratorP = P.iterator();
                int count = -1;
                Integer position = iteratorP.next();

                while (iteratorP.hasNext() && iteratorL.hasNext())
                {
                    Integer element = iteratorL.next();
                    count++;

                    if (position.equals(count)) {
                        System.out.print(element + " ");
                        position = iteratorP.next();
                    }
                }
        }

        private static List<Integer> L = new ArrayList<Integer>();
        private static List<Integer> P = new ArrayList<Integer>();
}
```