CSC 212: Data Structures and Abstractions

Spring 2019

University of Rhode Island

Weekly Problem Set #5

Due Thursday 3/28 at the beginning of class. Please turn in neat, and organized, answers hand-written on standard-sized paper **without any fringe**. At the top of each sheet you hand in, please write your name, and ID.

1 Recurrences

1. Find a closed-form equivalent of the following recurrences:

   (a) The Towers of Hanoi:

   $$T(0) = 0; T(n) = 2T(n-1) + 1$$

   (b) Merge Sort:

   $$T(1) = 1; T(n) = 2T(\frac{n}{2}) + n$$

   (c) Generic:

   $$T(0) = 1; T(n) = T(n-1) + 2^n$$

   (d) Generic:

   $$T(1) = 1; T(n) = T(\frac{n}{3}) + 1$$

2 Merge Sort

1. Determine the running-time of merge sort for a) sorted input; b) reverse-ordered input; c) random input; d) all identical input. Justify your answers.

2. Show the steps to sort the following array using Merge Sort: 6 1 7 11 4 10 2 5 9 3 8

3 Recursion

1. Write a recursive algorithm to find the maximum of a weakly unimodal array of integers given the array and its start and end indices.

```
int searchUnimodal(int* array, unsigned start, unsigned end)
```

2. For the following recursive function, determine the number of times foo is run given an initial call of foo(3)? foo(5)? What is the general formula for the number of calls; foo(n)? (Remember to count the initial call of foo also)

```
void foo(unsigned n) {
    if(n > 0) {
        foo(n/2);
        foo(n/2);
        foo(n/2);
    }
}
```