

### Homework assignment 3:

**Suggested due date:** Friday, September 29 2017 at 03:30pm

- 1- Suppose a machine on average takes  $10^{-8}$  seconds to execute a single algorithm step. What is the largest input size for which the machine will execute the algorithm in 2 seconds assuming the number of steps of the algorithm is  $T(n) =$ 
  - a.  $\log n$
  - b.  $\sqrt{n}$
  - c.  $n$
  - d.  $n^2$
  - e.  $n^3$
  - f.  $2^n$
- 2- For the machine in the previous example, how long will it take to run the algorithm for an input of size 1,000, assuming the time complexities from the same example?
- 3- An algorithm takes 0.5 seconds to run on an input of size 100. How long will it take to run on an input of size 1000 if the algorithm has a running time that is *linear*? *quadratic*? *log-linear*? *cubic*?
- 4- An algorithm is to be implemented and run on a processor that can execute a single instruction in an average of  $10^{-9}$  seconds. What is the largest problem size that can be solved in *one hour* by the algorithm on this processor if the number of steps needed to execute the algorithm is  $n$ ,  $n^2$ ,  $n^3$ ,  $\log n$ ? Assume  $n$  is the input size.
- 5- Determine the asymptotic running time for the following piece of code, assuming that  $n$  represents the input size.
  - a.

```
sum = 0;
for(i=0; i < n; i++)
    sum++;
```
  - b.

```
sum = 0;
for(i=0; i < n; i++)
    for(j=0; j < n; j++)
        sum++;
```
  - c.

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

- d.     `sum=0;`  
           `for(i=0;i<n;i++)`  
               `for(j=0; j< i;j++)`  
                   `sum++;`
- e.     `sum = 0;`  
           `for(i=0; i < n; i++)`  
               `for(j=0; j < i*i; j++)`  
                   `for(k=0; k < j; k++)`  
                       `sum++;`
- f.     `sum = 0;`  
           `for(i=0; i < n/2; i++)`  
               `for(j=0; j < (i*i)/2; j++)`  
                   `sum++;`

6- Sort the below numbers using: ([Show the work](#))

- a. Counting Sort
- b. Radix Sort
- c. Insertion Sort
- d. Bubble Sort
- e. Selection Sort
- f. Merge Sort
- g. Quicksort

- 1, 2, 0, -3, 5, -7, 10
- 0, 2, 3, 8, 9, 16

7- Sort the below numbers using: ([Show the work](#))

- a. Merge Sort
- b. Quicksort

- 8, 0, 2, -1, -2, 2, 3, 7, -6, -9
- 19, 7, 6, 3, 2, -1, -7, -18

8- Perform the partitioning algorithm on the below array using the median-of-three heuristic.

- 1, 2, 6, -3, 20, -61, 7, 8, 19, 100
- 0, 7, -6, 23, 12, 30, -71, 19

9- What is the *largest* value of  $n$  such that an algorithm whose running time is  $10n^2$  runs faster than an algorithm whose running time is  $50n$  on the same machine?

10- What is the running time of Insertion Sort if all elements are equal? Explain.