School of Computer Science
University of Guelph

CIS*3490 The Analysis and Design of Algorithms

Winter 2021
Instructor: Fangju Wang

Assignment 1

1. Question 3 on page 59 in the text. (10%)

2. Question 5 on page 60 in the text. (10%)

3. Question 1 on page 67 in the text. (10%)

4. Question 2 on page 67 in the text. (10%)

5. Question 3 on page 67 in the text. (10%)

6. Consider the following algorithm and answer the questions. (10%)

```
ALGORITHM X($A[0..n-1]$)
    // Input: $A$ contains $n$ real numbers
    for $i \leftarrow 0$ to $n-2$ do
        for $j \leftarrow i+1$ to $n-1$ do
            if $A[j] > A[i]$
                swap $A[i]$ and $A[j]$
```

   1. What does this algorithm compute?
   2. What is the input size?
   3. What is the basic operation?
   4. How many times is the basic operation executed? (Calculate the function expressing the number of repetitions of the basic operation.)
   5. What is the efficiency class of this algorithm?

7. Solve the following recurrences and find the efficiency class for each of them. (10%)

   1. $A(n) = 3A(n-1)$ for $n > 1$, $A(1) = 4$

2. $A(n) = A(n-1) + 5$ for $n > 1$, $A(1) = 0$

3. $A(n) = A(n-1) + n$ for $n > 0$, $A(0) = 0$

4. $A(n) = A(n/5) + 1$ for $n > 1$, $A(1) = 1$ (solve for $n = 5^k$)

5. $A(n) = 2A(n/2) + n - 1$ for $n > 1$, $A(1) = 0$ (solve for $n = 2^k$)

8. Consider the following algorithm and answer the questions. (10%)

```
ALGORITHM $Y(n)$
        // Input: $n$ is a positive integer
        if $n = 1$ return 1
        else return $Y(n-1) + n * n$
```

1. What does this algorithm compute?

2. What is the input size?

3. What is the basic operation?

4. Set up a recurrence and an initial condition and find the number of times the basic operation is executed.

5. What is the efficiency class of this algorithm?

9. Question 4 on page 77 in the text. (10%)

10. Consider the following algorithm and answer the questions. (10%)

```
ALGORITHM W($A$, $l$, $r$, $K$)
        // Input: $A$ is an array of sorted integers,
        //        $l$ and $r$ are the leftmost and rightmost indexes of the
        //        array elements to be processed,
        //        $K$ is an integer
        if $l > r$ return $-1$
        else
                $m \leftarrow \lfloor (l+r)/2 \rfloor$
                if $K = A[m]$ return $m$
                else if $K < A[m]$ return W$(A, l, m-1, K)$
                     else return W$(A, m+1, r, K)$
```

1. What does the algorithm compute?

2. How is the input size n expressed in terms of the parameters?

3. Assume that after comparison of K with $A[m]$, the algorithm can determine whether K is smaller than, equal to, or larger than $A[m]$. Set up a recurrence (with an initial condition) for comparison in the worst case of this algorithm. Solve the recurrence for $n = 2^k$, and determine the Θ efficiency class.

4. What is the Θ efficiency class when $n \neq 2^k$? Why?

Due time: 23:59, Monday, Feb 1, 2021. Please submit an e-copy to Moodle.