

Data Structures

Fall 2019, Homework #1

Date: Oct. 14, 2019

1. (35 pts) For the following pairs of functions, indicate whether the ?? could be replaced with O , Ω , or Θ . If more than one may be correct, indicate all that apply.

Pr.	$f(n)$	$g(n)$	$f(n)$ is ?? ($g(n)$)
a)	$.00001n^4$	$375 \log_2 n$	
b)	$24n$	$2n^2 + 1$	
c)	$3n^3 + 2n$	2^n	
d)	$6n$	$\log_2 n$	
e)	$n!$	$5n^{20} + n^2 \log_2 n$	
f)	$3n$	$3 \log_2(2^n)$	
g)	$95n + 2$	$3n + \log_2 n$	

2. (30 pts) Analyze the running time of each of the following three functions (namely, A, B, C). Each of the functions takes as input an integer array arr and its length n . For each function you should give the running time $T(n)$ as a function of n and then give the appropriate Big-O classification.

NOTE: For the purposes of this problem, integer addition and multiplication are the only basic operation that you should count when determining the running time $T(n)$ (i.e. you can ignore array accesses, assignments, etc.). You can assume for simplicity that n is divisible by 2.

```
int A(int arr[], int n)
{
    int result = 25;

    int i = n;
    while (i > 1) {
        for (int j = 0; j < n; j++) {
            result += 1;
            result += 3 * arr[j];
        }
        i = i / 2;
    }

    return result;
}
```

```

int B(int arr[], int n)
{
    int result = 0;

    for (int i = 0; i < n/2; i++) {
        result += 3 * arr[i];
    }

    for (int i = n/2; i < n; i++) {
        result += 4 * arr[i];
    }

    for (int i = n-1; i >= 0; i--) {
        result += 3 * arr[i];
        result += 5 * arr[i];
    }

    return result;
}

int C(int arr[], int n)
{
    int result = 0;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < 5; j++) {
            for (int k = 0; k < n; k++) {
                result += j * j + arr[k];
            }
        }
    }
    return result;
}

```

3. (20 pts) List the following functions in order of their asymptotic growth. That is if $f_i(n) = O(f_j(n))$ then $f_i(n)$ should come before $f_j(n)$ in your list. If $f_i(n) = \Theta(f_j(n))$ then the two functions should be given the same rank (in this case, group the two functions together by circling them).

- $f_1(n) = n^3$
- $f_2(n) = n!$
- $f_3(n) = n \log_2 n$
- $f_4(n) = 1$
- $f_5(n) = 2^{\log_2 n}$
- $f_6(n) = 10n \log_{10} n$
- $f_7(n) = (n + 1)!$
- $f_8(n) = 2^{\log_{50} n}$
- $f_9(n) = 4^{\log_2 n}$
- $f_{10}(n) = n^{\log_2 \log_2 n}$

4. (15 pts) Prove or disprove the following:

If $f(n) = O(h_1(n))$ and $g(n) = O(h_2(n))$, then $\frac{f(n)}{g(n)} = O\left(\frac{h_1(n)}{h_2(n)}\right)$, where f, g, h_1, h_2 are functions of positive values.