

KING SAUD UNIVERSITY
COLLEGE OF APPLIED STUDIES AND COMMUNITY SERVICE
GC211(Data Structures)
Tutorial 2

Question 1:

Find the running time as a function $T(n)$ of input size n in each of the following cases considering only the number of iterations of the loops (**only the number of comparison operations in the loops**)

Express the growth rate of the function in Big O notation.

(a) $s = 0;$ **1**
 $\text{for } (i = n; i \geq 1; i--)$ $(1-n+1+1)=$ **$3 \cdot n$**
 $s = s+1;$ **$2 \cdot n$**

$f(n) = -2n + 6$

$\text{big-O} = o(n)$

(b) $s = 0;$ **1**
 $\text{for } (i = 1; i \leq n; i++)$ $(n-1+1+1)=$ **$n+1$**
 $\text{for } (j = i; j \leq n; j++)$ $n \cdot (n+1) =$ **$n^2 + n$**
 $s = s+1;$ $n \cdot (n) =$ **n^2**

$f(n) = 1 + n + 1 + n^2 + n + n^2 = 2n + 2n^2 + 2$

$\text{big-O} = o(n^2)$

(c) `float power(float a, int n)`

```
{  
    float p = a;  
    float r = 1;  
    int m = n;  
    while n > 0 do  
        if n is odd  
            n--;  
            r = r · p  
        else  
            n = n/2;
```

KING SAUD UNIVERSITY
COLLEGE OF APPLIED STUDIES AND COMMUNITY SERVICE
GC211(Data Structures)
Tutorial 2

```
    p = p.p;  
end if  
end while  
return r}
```

Question 2:

- Express the following functions in terms of Big-O notation (a, b and c are constants). Which one is the Best? Which one is the worst?

- 1) $f(n) = an^2 + bn + c$ $O(n^2)$
- 2) $f(n) = 2^n + n \log n + c$ $O(2^n)$
- 3) $f(n) = n \log n + b \log n + c$ $O(n \log n)$
- 4) $3(n+1)^7 + 2n \log n$ $O(n \log n)$