# CS_601_HOMEWORK_2

Group of 2's :- 1) Aryan Bhagat - 815310
2) Moksha Shah - bp46199

Q-1 :-
```
for(int i=1; i<n; i++){
    //basic operation;
    for(int j=i; j<=n; j++){
        //basic operation;
    }
}
```

(i) Outer Loop Operations :-

it runs from $i=1$ to $i=n-1$
so, $\boxed{(n-1)}$ times

(ii) Inner loop Operations :-
For $i=1$ to $i=n-1$
it runs $j=i$ to $j=n$ (i)

$$\sum_{i=1}^{n-1} (n-i+1)$$

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

$$= \frac{2n^2-2n-n^2+n+2n-2}{2}$$

$$= \boxed{\frac{n^2+n-2}{2}} \text{ times}$$

iii) Total Operations:

Outer opr + Inner opr

$$(n-1) + \frac{n^2+n-2}{2}$$

$$= \frac{2n-2+n^2+n-2}{2}$$

$$= \boxed{\frac{n^2+3n-4}{2}}$$

Q-2

Order of growth [Non-Decreasing]:

→ $4000 < n(n^{1/3}) < \sqrt{n} < n < n\log n < n\ln(n)$

$< n^2 < n^3 < 2^{n+3} \equiv 2^n < 3^n < n! < n^n$

$4000 < n < n\ln(n) \equiv n\ln(n) < n\sqrt[3]{n}$

$n\sqrt[3]{n} < n\sqrt{n} < n^2 < n^3 < n^{\log(n)} < (1.1)^n$

$(1.1)^n < 2^n \equiv 2^{n+3} < 3^n < n! < n^n$

Q-3 By assumption and definitions of $\Theta$ and $O$, there exists $c_0 > 0$ and $n_0$, such that for $n > n_0$ we have $f(n) \le c_0 n \log(n)$, there exists $c_1 > 0$ and $n_1$ such that for $n > n_1$ we have $f(n) \geq c_1 n \log(n)$ and there exists $c_2 > 0$ and $n_2$ such that for $n \geq n_2$ we have $g(n) < c_2 n$.

For $n \geq \max(n_0, n_1, n_2, 2)$ we thus have (since then $\log(n) \geq 1$ as $n \geq 2$)

$$f(n) + g(n) \le c_0 n \log(n) + c_2 n \log(n) = (c_0 + c_2) n \log(n).$$

which shows that $f+g \in O(n \log(n))$, and also ($g > 0$) $f(n) + g(n) \geq f(n) \geq c_1 n \log(n)$ which shows $f+g \in \Omega(n \log(n))$. Hence, $f+g \in \Theta(n \log(n))$.

Q-4 ~~Estimating the run time~~

(a)

```
z ← 0
For j ← 1 to n
    q ← 0
    s ← j*j
    while q ≤ s
        q ← 5+q
        z ← z+q
```

The outer loop runs $n$ times with the $j^{th}$ iteration taking time in $\Theta(j^2)$. Since squaring is smooth function, total running time is $\Theta(n \cdot n^2) = \Theta(n^3)$.

classmate
Date
Page

(b) The outer loop iterates $4n$ times with the $j$th iteration taking time is $\Theta(\log_5(j^2))$, is smooth function (as $\log_5((2j)^2) = \log_5(4j^2) = \log_5(4) + \log_5(j^2) \le 2\log_5(j^2)$ when $j \geq 2$)

$\therefore$ Total running time is $\Theta(4n \log_5((4n)^2))$
$= \Theta(n \log n)$.

(c) In last iteration, $j$ satisfies $n/2 < j \le n$ and hence the inner loop runs in time $\Theta(n)$.

- The running time of last iteration is bigger than the total running time of all previous iterations, since $1 + 2 + \ldots + 2^{k-2} + 2^{k-1} = 2^k - 1$.

Thus, the total running time is $\Theta(n)$.