

**20100275**

**Hamzah Baig**

**Answer 4-i:**

Order of Complexity is  $O(\log_2 n)$  because  $l$  is multiplied by 2 in every iteration.

**Answer 4-ia:**

Upper Bound:  $O(n^3)$

Lower Bound:  $O(n^3)$

Because first there is a nested loop (  $O(n^2)$  ) and in every nested loop there is a function call in which loop runs  $n$  times. Hence, total order of complexity is  $O(n^3)$ .

**Answer 4-iib:**

Sum from index 'i' to 'j' is stored in 2-D W array.

**Answer 4-iiic:**

```
int x,y,n;

/* P is a 1-D array of size n integers and W is a 2-D array of size n x n integers */

int previous; for(x= 0; x <n ; x++) {   previous = P[x];   for (y = x+1; y <n; y++) {

        W[x][y] = P[y] + previous;

        previous = W[x][y] ;

    }

}
```