Q:3 'O' notation

i) $(n+1)^2 \in O(n^2)$

$n^2+2n+1 \in O(n^2)$

while considering properties of tight upper bound ~~and~~ ~~tight~~ ~~for~~

it should be

$f(n) \leq c_2 \cdot g(n)$

$n^2+2n+1 \leq c_2 \cdot n^2$

as it holds the property of tight upper bound so according to mathematics, it proves the statement.

ii) $n^3 \in O(n^2)$

$f(n) \leq c \cdot g(n)$

$n^3 \leq c \cdot n^2$

So according to mathematics it does not hold the property because $O(n^2)$ is not tight upper bound for function $f(n) = (n^3)$

So it disatisfies property.

(iii) $n! \in O((n+1)!)$

$f(n)! \leq c\, g(n)$

$n! \leq (n+1)!$

it means (according to mathematics) actually it holds the property because $n!$ and $(n+1)!$ ~~can~~ hold the tight upper bound.

$n! \leq n! + 1!$

$n! \leq n!$

$n! \leq O(n!)$

So it holds the tight upper bound. So it proved.

Q:4)

```
for (i=0 ; i<n ; i++)
    if (a[i] == x) return 1
    return -1;
```

x is inserted as input

=> worse case

```
for (i=0; i<n; i++) //worst case
    if (a[i] == x) // worse case
        return 1;
```

total comparisons in worst case is $2n+1$

$O(n)$

=> best case ~~of~~ if searching at find at first index

it is $O(1)$

=> avg case

it is $O\left(\frac{n}{2}\right)$

because index may b found at middle point of array.