1. In the binary search algorithm, It is suggested to calculate the mid as beg + (end_beg)/2 instead of (beg + end)/2. why is it so.

Because (beg + end) may overflow which then means you get a result that is less than beg or for into the -ve if you are using signed integer
So, instead they take the distance b/w beg and end and add half of that to beg
This is only a single extra operation to make the algorithm more robust

Q write the algorithm/fun for ternary search

Ans:-
```
Int ternary-search(int l, int r, int x)
{
 if (r>=1)
 {
 int mid1 = l+(r-l)/3;
 int mid2 = r-(r-l)/3;
 if (ar[mid1]==x)
  return mid1;
 if (ar[mid2]==x)
  return mid2;
if (x < arr [mid1])
return ternary_search(l, mid1-1, x);
else if (x > arr [mid2])
return ternary-search(mid2 + 1, r, x);
else return ternary-search (mid1+1, mid2-1,x)
}
return -1;
}
```