Question 1

Since mid is an instance of a numeric type, it has an upper limit on the value that it can hold. It is possible that the sum of beg and end could exceed this maximum value, leading to overflow and unpredictable results. This can occur even if beg and end are both legal values (i.e. both positive and end >= beg).

The expression mid = beg + (end - beg) / 2 will never overflow for legal values of end and beg and will always give the desired result.

Question 2

```
mid1 = beg+(end-beg)/3;
mid2 = end – (end-beg)/3;

while(beg<=end)
{
        if( a[mid1] == x )
                return mid1;
        else if( a[mid2] == x )
                return mid2;
        else if( x < a[mid1] )
                end=mid1-1;
        else if( x > a[mid2])
                beg=mid2+1;
        else
        {

                beg=mid1+1;
                end=mid2-1;
        }

}
```