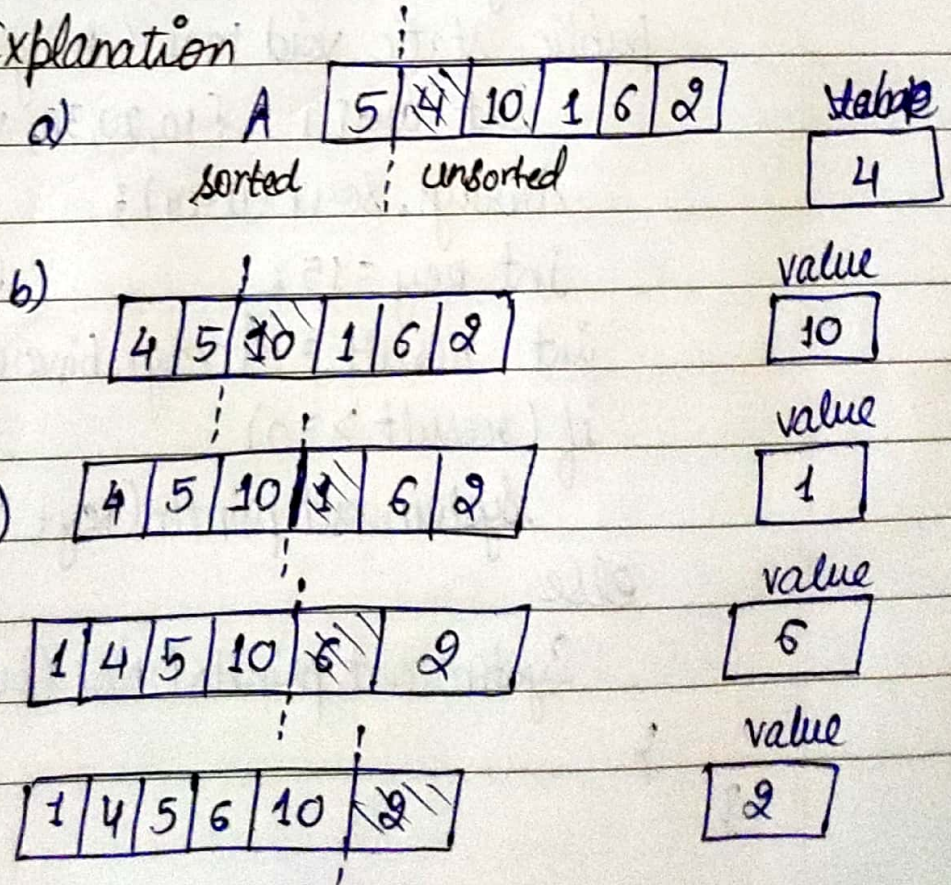
Assignment

Ques 1. Insertion Sort

```
InsertionSort(A,n)
{   for i ← 1 to n-1
    value ← A[i]
    hole ← i
    while(hole ≥ 0 && A[hole-1] > value)
    {   A[hole] ← A[hole-1]
        hole ← hole -1
    }
    A[hole] ← value
  }
}
```

Code Diagram Explanation



f) | 1 | 2 | 4 | 5 | 6 | 10 | Sorted array.

(ii) Selection Sort

```
SelectionSort (A, n)
{   for(i ← 0 to n-2
    {iMin ← i.
     for j ← i+1 to n-1
     {
       if (A[j] < A[iMin])
          iMin ← j
     }
     temp ← A[i]
     A[i] ← A[iMin]
     A[iMin] ← temp
    }
}
```

Code Diagram Explanation

0	1	2	3	4	5
7	4	10	8	3	1

sorted | unsorted

Pass 1:

1	4	10	8	3	7

Pass 2:

1	3	10	8	4	7

Pass 3:

1	3	4	8	10	7

Pass 4:- | 1 | 3 | 4 | 7 | 10 | 8 |

Pass 5:- | 1 | 3 | 4 | 7 | 8 | 10 | sorted array.

Ques 2. Merge Sort

```
MergeSort (A, p, r)
 if p<r
 q = [(p+r)/2]
  MergeSort(A, p, q)
 MergeSort (A, q+1, r)
  Merge(A, p, q, r)

Merge (A, p, q, r)
    n1 = q-p+1
    n2 = r-q
let L and R be new arrays
    for i = 1 to n1
        L[i] = A[p+i-1]
        for j = 1 to n2
            R[j] = A[q+j]
    L[n1+1] = ∞
    R[n2+1] = ∞
     i = 1
```

```
j = 1
for k = p to r
  if L[i] ≤ R[j]
    A[k] = L[i]
    i = i + 1
  else  A[k] = R[j]
        j = j + 1
```

Diagram Explanation