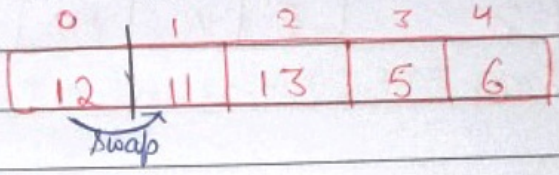
Date: / / Page no:

• Insertion Sort :-

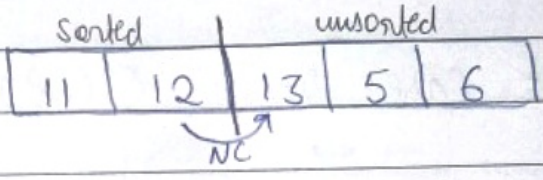
It is simple sorting algorithm that works similar to way you sort playing cards in your hands. The array is virtually split into a sorted and an unsorted part.

Values from the unsorted part are picked and placed at the correct position in the sorted part.

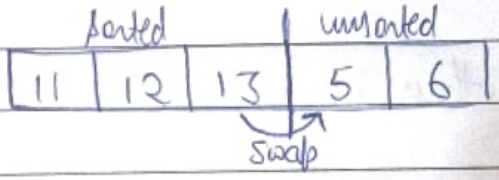
- Working :-



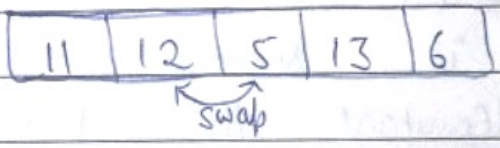
(i=0) first pass :-

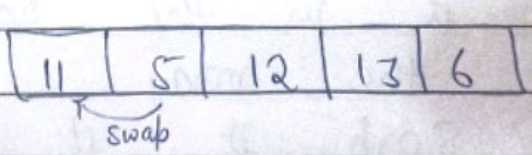


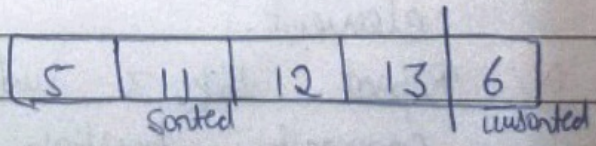
(i=1) second pass :-



(i=2) Third pass :-

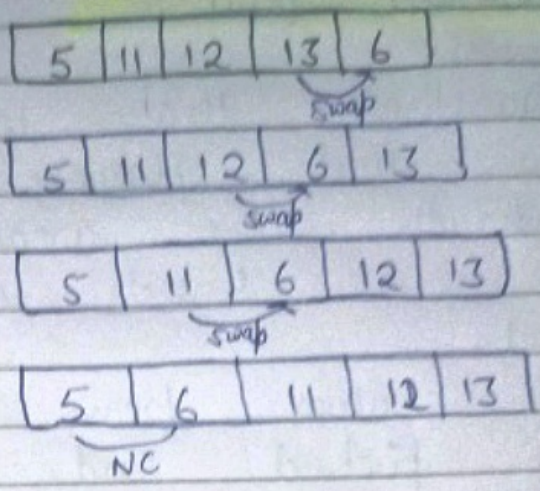






Date: / / Page no:

(i=3) fourth pass :-



Sorted array :- | 5 | 6 | 11 | 12 | 13 |

Worst case :- $O(n^2)$

- Algorithm :-

i) Start with 2nd element (Key), assume 1st is sorted

ii) Compare the Key with element before it in the sorted portion of the array.

iii) Swap it, if it is smaller than element.

iv) Repeat step 3 until you find the correct position.

v) Move to next element and repeat (2-4) step.

Date: / / Page no:

- **Code Implementation :-**

```
import java.util.*;
public class main {
    public static void main(String[] args)
{
    int[] arr = {11, 13, 12, 5, 6};
    insertion(arr);
    Sout(Arrays.toString(arr));
}

Static void insertionSort(int[] arr)
{
for(int i=0; i<arr.length-1; i++)
{
  for(int j=i+1; j>0; j--)
{
  if(arr[j] < arr[j-1])
{
    int temp = arr[j];
    arr[j] = arr[j-1];
    arr[j-1] = temp;
}
  else {
      break;
}
}
}
}
```