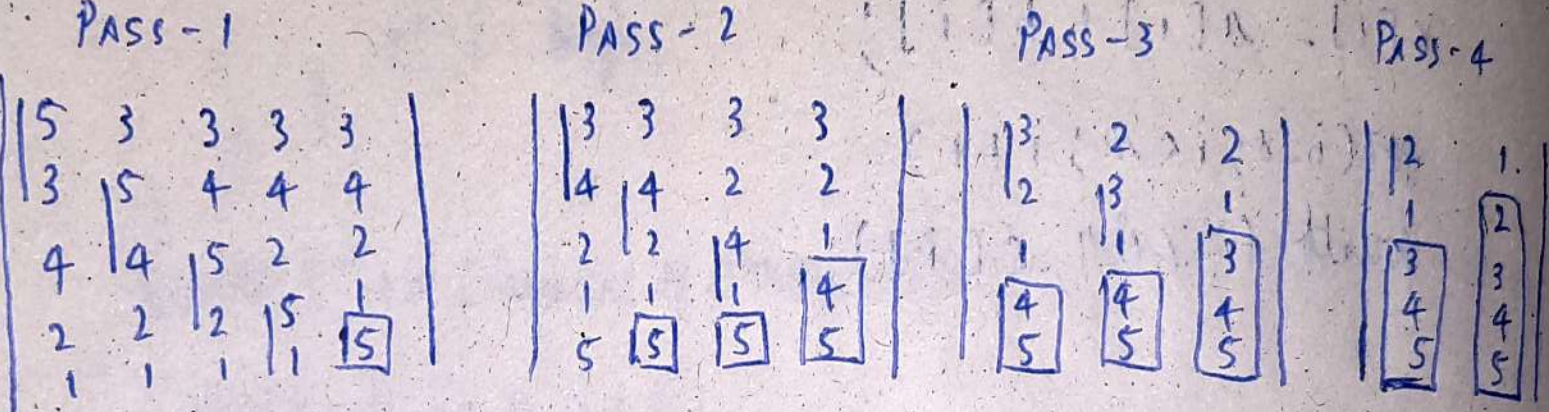
BUBBLE SORT / EXCHANGE SORT =>

Ex:- 5 3 4 2 1

n - 1

n - pass




```
main()
{
    int a[20];
    int i, n, t, p, c, ex;
    printf("enter n");
    scanf("%d", &n);
    printf("enter a elements");
    for(i=0; i<n; i++)
    scanf("%d", &a[i]);
```

ex -> EXCHANGE

t -> TEMPORARY

P -> PASS

```
for (p=0; p<n-1; p++)
{  for (ex=0, c=0; c<n-p-1; c++)
   {
     if (a[c] > a[c+1])
     {
       t = a[c];
       a[c] = a[c+1];
       a[c+1] = t;
       ex = 1;
     }
   }
   if (ex==0)
   break;
}
for (i=0; i<n; i++)
printf ("%d ", a[i]);
}
```

nums = [0 0 1 1 2 2 3 3 4]
[0 1 2 3 4 - -]
i++

=

Leetcode : 26

```
#include <stdio.h>
int remove (int num, int numsize)
{
  if (numsize==0)
  return 0;
  int i, j;
  i=0;
  for (j=0; j< numsize; ++j)
  { if (i<1 || num[j] > num[i-1])
    { num[i++] = num[j];
  } }
```

```
    return i;
}

int main()
{
    int nums[] = {1, 1, 2};
    int numsSize = sizeof(nums) / sizeof(nums[0]);
    int newLength = remove(nums, numsSize);
    printf("new length: %d\n", newLength);
    printf("After removing duplicates in array: ");
    for(i = 0; i < newLength; ++i)
    {
        printf(" %d", nums[i]);
    }
    printf("\n");
    return 0;
}
```