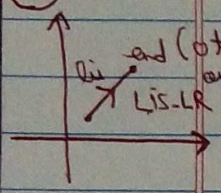
→ LIS/LDS variation

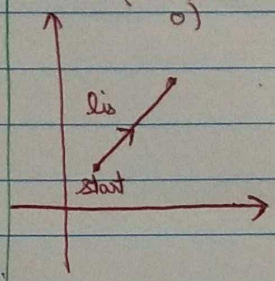
(Note:- where the end/start pt. is present dir^{n} of inner loop ~~start~~ originates from that point) (curr. i)

1.



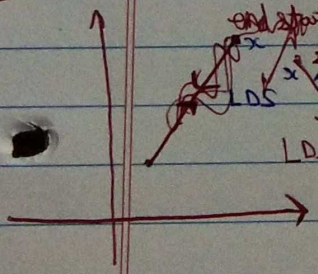

```
int lis_ending(int arr[], int n){
  int dp[] = new int[n]; int maxLen=0
for(int i=0; i<n; i++){
    dp[i]=1;
    for(int j=i-1; j>=0; j--){
      if(arr[i]> arr[j])
      dp[i]= max(dp[i],dp[j]+1);
    }
  maxLen = max(maxLen, dp[i]);
  }
return maxLen;
  }
```

2.




```
int lis_starting(int arr[], int n)
  int dp[] = new int[n];
  int maxLen = 0;
  for(int i = n-1; i>=0; i--)
    {
    ...
  for(int j=i+1; j<n; j++)
  { if(arr[i] < arr[j])
```

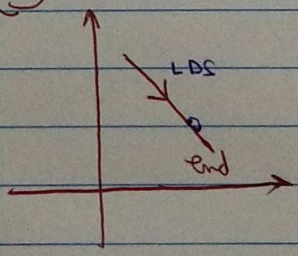
3.



LDS starting at x = LIS ending at x

```
for(int i=0; i<n; i++){
  dp[i]=1;
  for(int j=i-1; j>=0; j--)
    {
  if(arr[j] > arr[i])
```

4.



LDS end at x = LIS start at x from diff dir^{n}

```
for(int i=n-1; i>=0; i--)
  { dp[i]=1;
for(int j=i+1; j<n; j++)
  { if(arr[j] < arr[i])
  }
```

Sangam A Quality Product GOLD

Signature