

Module02_Day06_Sorting_2

December 16, 2022

Sorting

Optimized Buble Sort

```
[ ]: def optBubbleSort(lst):  
    count = 0  
    n = len(lst)  
    for i in range(n-1):  
        already_sorted = True  
        for j in range(n-1-i):  
            count+=1  
            if lst[j] > lst[j+1]:  
                lst[j],lst[j+1] = lst[j+1],lst[j]  
                already_sorted=False  
        if already_sorted==True:  
            break  
    return count
```

```
[ ]: a = list(range(1000,1,-2))  
    optBubbleSort(a)
```

```
[ ]: 124750
```

```
[ ]: optBubbleSort(a)
```

```
[ ]: 499
```

Selection Sort

```
[ ]: def selSort(arr):  
    n = len(arr)  
    count=0  
    for i in range(n-1):  
        min_idx = i  
        for j in range(i+1,n):  
            count+=1  
            if arr[i] > arr[j]:  
                min_idx = j  
        arr[i],arr[min_idx] = arr[min_idx],arr[i]  
    return arr,count
```

```
[ ]: a = [1,4,2,6,5,7,0]
      b = list(range(1000,1,-2))
      selSort(a)
```

```
[ ]: ([0, 1, 2, 4, 5, 6, 7], 21)
```

```
[ ]: def optSelSort(arr):
      n = len(arr)
      count=0
      for i in range(n-1):
          array_sorted=True
          min_idx = i
          for j in range(i+1,n):
              count+=1
              if arr[i] > arr[j]:
                  min_idx = j
                  array_sorted=False
          arr[i],arr[min_idx] = arr[min_idx],arr[i]
          if array_sorted==True:
              break
      return arr,count
```

```
[ ]: a = [1,4,2,6,5,7,0]
      b = list(range(1000,1,-2))
      optSelSort(a)
```

```
[ ]: ([0, 1, 2, 6, 5, 7, 4], 15)
```

```
[ ]: optSelSort(a)[0][:2]
```

```
[ ]: [0, 1]
```

Insertion Sort

```
[ ]: def insSort(arr):
      n = len(arr)
      for i in range(1,n):
          key = arr[i] # Create our temp variable
          j = i-1 # Checks all element from left
          while j >=0 and key<arr[j]:
              arr[j+1] = arr[j] # Moves element to right untill key<arr[j]
              j -=1
          arr[j+1] = key # update this with temp value
      return arr
```

```
[ ]: a = [1,4,2,6,5,7,0]
      insSort(a)
```

```
[ ]: [0, 1, 2, 4, 5, 6, 7]
```

0.0.1 Homework

```
[ ]: def merge(A,B):  
    n = len(A)  
    m = len(B)  
    C = list()  
    i,j = 0,0  
  
    while(len(C)!=n+m):  
        if A[i] < B[j]:  
            C.append(A[i])  
            i +=1  
            if i == n:  
                C.extend(B[j:])  
  
        else:  
            C.append(B[j])  
            j +=1  
            if j == m:  
                C.extend(A[i:])  
  
    return C
```

```
[ ]: a =[1,3,4,5,7,9,11]  
    b = [2,4,6,8]  
    merge(a,b)
```

```
[ ]: [1, 2, 3, 4, 4, 5, 6, 7, 8, 9, 11]
```
