مدونة المناهج السعودية

https://eduschool40.blog

الموقع التعليمي لجميع المراحل الدراسية

في المملكة العربية السعودية

Question 3: (5 Marks) **Do the following for Quick sort algorithm [5]**

A. Write the algorithm for Quick sort technique. (2)

```
QuickSort(A,P,r){
if(P<r){
q = Partition(A,P,r)
QuickSort(A,P,q-1)
QuickSort(A,q+1,r)}}
```

```
Partition(A,P,r){
x = A[r]
i -> P-1
for j  P -> r-1
if(A[j] <= x){ i = i+1
exchange A[i] <-> A[j]}
exchange A[i+1] <-> A[r]
return i+1}
```

B. Given the following array A, use the partition algorithm to find the pivot value and to divide A into three subsets. List all intermediate steps and give the upper bound time complexity of this algorithm. (3)

ex A=[7, 10, 3, 14, 5, 6, 15, 4, 12, 1, 9, 8] x = 8

7, 10, 3, 14, 5, 6, 15, 4, 12, 1, 9, 8

7, 3, 10, 14, 5, 6, 15, 4, 12, 1, 9, 8

7, 3, 5, 14, 10, 6, 15, 4, 12, 1, 9, 8

7, 3, 5, 6, 10, 14, 15, 4, 12, 1, 9, 8

7, 3, 5, 6, 4, 14, 15, 10, 12, 1, 9, 8

7, 3, 5, 6, 4, 1, 15, 10, 12, 14, 9, 8

7, 3, 5, 6, 4, 1, 8, 10, 12, 14, 9, 15

Best wishes

Question 2: Answer the following question (8 Marks)

1. Order the following growth rate in ascending order. [2]

$500n^2$, $30000n\log n$, 2^n, n^3, $10000n^2\log n$, $10^5 n$, 500^{500}

2. Analyze the following codes to find the time complexity? [2]

A.

```
for(i=0; i < N; i++)
  {
for(j=0; j < N;j++)
{
statement;
}
  }
```

B.

```
while(low <= high)
{
 mid = (low + high) / 2;
 if (target < list[mid])
     high = mid - 1;
else if (target > list[mid])
    low = mid + 1;
else break;
}
```

3. Find the upper bound for this function? [2]

A. $3N^4 + 20N^2 - 2N + 1$

B. $10 + 2 + 50000 + 90!$

4. Complete the following to solve the problem of finding the total elements in an array by recursive algorithm [2]

- Base case

  if the array is empty return 0

- Recursive step

  counter → 0

  1- Add the first element to the counter

  2- move the counter to next element

1. What do we analyze about them

a. Correctness
b. complexity
c. Amount of space used
d. All of them

2. Which of the following sorting algorithm is of brute force type?

a. Merge sort
b. Selection sort
c. Quick sort
d. Insertion sort

3. The complexity of Polynomial algorithm by brute force is

a. $O(n^2)$
b. $O(n)$
c. $O(n \log n)$
d. $O(\log n)$

4. The complexity of the traveling salesman problem by exhaustive search is

a. $O(n!)$
b. $O(n)$
c. $O(n \log n)$
d. $O(n*m)$

5. The complexity of the knapsack problem by exhaustive search is

a. $O(n \log n)$
b. $O(n)$
c. $O(2^n)$
d. $O(3)$

. An algorithm design paradigm that works by recursively breaking down a problem into two or more sub-roblems of the same type, until these become simple enough to be solved directly.

a. Divide and conquer
b. Brute force
c. exhaustive search
d. Decrease and conquer

	2	3	4	5	6
d	b	a	a	c	a