Name :- Abhay Rawat

Course :- B. Tech (CSE) Sec :- B

Sem :- V Roll No :- 01

University Roll No :- 1961002

# Design & Analysis of Algorithm

## Tutorial Sheet 3

Ques 1 Write linear search pseudocode to search an element in a sorted array with minimum comparisions.

Ans
```
Linear_Search (Arr, key)
    for i = 0 to Arr.length
        if (Arr[i] == key)
            return i

    return -1
```

Ques 2 Write pseudo code for iterative & recursive insertion sort. Insertion sort is called online sorting. Why? What would other sorting algorithm that has been discussed in lectures?

Ans Insertion sort (iterative method)

```
INSERTION_SORT (A)
  for j = 2 to A.length
      key = A[j]
      for i = j-1
      i = j - 1
      while i>0 and A[i]>key
              A[i+1] = A[i]
              i = i - 1
      A[i+1] = key
```

Insertion sort (Recursive Pseddocode)

```
INSERTION_SORT(A, n)
    if n>1
        INSERTION_SORT (A, n-1)
        key = A[n]
        i = n-1
        while i>0 and A[i]>key
              A[i+1] = A[i]
              i = i-1
        A[i+1] = key
```

Insertion sort produces a partial solution each iteration & hence it is called online ~~solution~~ sorting

The sorting algorithm that can be discuss in lectures are:-

I Bubble Sort
II Selection Sort
III Insertion Sort
IV Merge Sort
V Quick Sort
VI Count Sort
VII Heap Sort

Ques3 Complexity of all sorting algorithms that has been discussed in lecture.

Ans

| Sorting Methods | Best Case | Average Case | Worst Case |
|---|---|---|---|
| Bubble Sorting | $n$ | $n^2$ | $n^2$ |
| Selection Sorting | $n^2$ | $n^2$ | $n^2$ |
| Insertion | $n$ | $n^2$ | $n^2$ |
| Merge Sort | $n \log n$ | $n \log n$ | $n \log n$ |
| Quick Sort | $n \log n$ | $n \log n$ | $n^2$ |
| counting Sort | $n+k$ | $n+k$ | $n+k$ |
| Heap Sort | $n \log n$ | $n \log n$ | $n \log n$ |

Ques4 Divide all the sorting algorithm into inplace / stable / online sorting.

Ans

| Sorting method | Inplace | Stable | Online |
|---|---|---|---|
| Bubble sort | Yes | Yes | No |
| Insertion Sort | Yes | Yes | Yes |
| Selection sort | Yes | No | No |
| Merge Sort | Yes | Yes | No |
| Quick Sort | Yes | No | No |
| Heap Sort | Yes | No | No |
| Count Sort | No | Yes | No |

Ques5 Write recursive / iterative pseudo code for binary search. What is the Time & Space complexity of Linear & Binary Search (Recursive & Iterative)

Ans Binary Search (Iterative)

```
BINARY_SEARCH(A, beg, end, key)
  while beg ≤ end
      mid = beg + (end - beg)/2
      if a[mid] == key
          return key
      if A[mid] < key
          beg = mid + 1:
      if A[mid] > key
          end = mid - 1
  return -1
```

Time complexity — $O(\log_2 n)$

Space complexity — $O(1)$

Binary Search (Recursion)

```
BINARY_SEARCH (A, beg, end, key)
   if( end ≥ beg)
        mid = ( beg + end) /2
        if (A[mid ] = = item)
                return  mid + 1
   else if (A[mid ]< item)
            return BINARY_SEARCH (A, mid+1, end,
                                        key)
   else
            return BINARY_SEARCH (A, beg, mid-1,
                                        end)
   return - 1
```

Time complexity — $\Theta(\log_2 n)$

Space complexity — $O(\log_2 n)$

Ques 6 Write recurrance relation for Binary recursive search.

Ans $T(n) = T(n/2) + c$

Ques 7 Find two index such that $A[i] + A[j] = k$ is minimum time complexity

Ans
```
create map a(int, int)
for element in array
        if (K - element) exist in a
                print key, index
        else
                add( elem, key) in map
```

Ques 8 which sorting is best for practical uses ? Explain.

Ans For practical uses, less time complexity is required, merge sort is has minimum time complexity cost. then other algorithm. Therefore merge sort is best algorithm in practical used.

Ques 9 what do you mean by number of inversion in an array ? count the number of inversion in Array arr[] = {7, 21, 31, 8, 10, 1, 20, 6, 4, 5} using merge sort.

Ans The inversions of an array indicate ; how many changes are required to convert the array into its sorted form

when an array is already sorted, it need 0 inversions & in another case the number of inversions will be maximum, if the array is reverse

For the given array, the no. of inversion would be 31.

Ques 10 In which cases Quick sort will given the best & the worst case time complexity?

Ans when the array is divided into two equal parts, the case will be consider as best case whereas when the divided array

are in the ratio of 99:1 or similar, the can is considered as worst case.

Ques 11 Write Recurrence relation of merge & Quick sort in best & worst case? What are the similarities & differences b/w complexity of two algorithms & why?

Ans Merge Sort

Best case $T(n) = 2T(n/2) + 1$

Worst case $T(n) = 2T(n/2) + 1$

Quick sort

Best case $T(n) = 2T(N/2) + N - 1$

Worst case $T(n) = 2T(N-1) + N - 1$

Similarity

| | Quick sort | Merge Sort |
|---|---|---|
| Best Case | $O(n \log n)$ | $O(n \log n)$ |
| Average Case | $O(n \log n)$ | $O(n \log n)$ |

Difference

| | Quick Sort | Merge sort |
|---|---|---|
| Worst Case | $O(n \log n)$ | $O(n^2)$ |

Ques 12 Selection sort is not stable by default but you write a version of stable selection sort.

Ans A stable algorithm can be made by changing the method from swapping to shifting in Selection sort, instead of swapping we can shift elements so that the order

remain same.

SELECTION_SORT(array, n)

for i = 0 to arr

SELECTION_SORT(A, n)

for j = 2 to n
key = A[j]
i = j - 1
while i > 0 and A[i] > key
A[i+1] = A[i]
i = i - 1
A[i+1] = key

Ques 13 Bubble sort scans whole array even when array is sorted. Can you modify the bubble sort so that it doesn't scan the whole array once it is stored.

Ans
```
void sort( vector<int> a , int n).
{
   int swapping = 0;
   for(int i=0; i<n; i++)
   { int swap = 0;
   for(int j=0; j<n-i-1; j++)
   {
      if(a[j] > a[j+1])
      {
         int temp = a[j];
                a[j] = a[j+1];
```

```
            a[j+1] = t;
            swap = 1;
        }
    }
    if (swap == 0)
        break;
}
}
```

Ques 14 You computer has a RAM of 2 GB & you are given an array of 4 GB for sorting. Which algorithm you are going to use for this purpose & why? Also explain the concept of External & Internal sorting.

Ans When RAM memory is unsufficient, we have to divide the data. Therefore we will use the merge sorting we will move data one by one & sort it. in RAM. This type of sorting. ~~On the~~ is also known as external sorting. On the otherhand internal sorting happens when no extra space is required