1

25th Feb, 2025

Design and Analysis of Algorithms

Step by step problem

Specific problem { Multiple solutions. | Time & Space.

Q. What are algorithms?

Ans:- An algorithm is any well defined computing procedure that takes some value or set of values as input and produces some values or set of values as output

∴ It is a sequence of computational steps that transform the input into the output.

I/P $\longrightarrow$ Algo $\longrightarrow$ O/P

1. Finiteness:- The steps should be finite.
2. Unambiguous:- The steps should have a single meaning.
3. Input: Zero/more inputs.
4. Output:- One/more outputs.

pseudo-code & convention — writeup (Assignment) next class

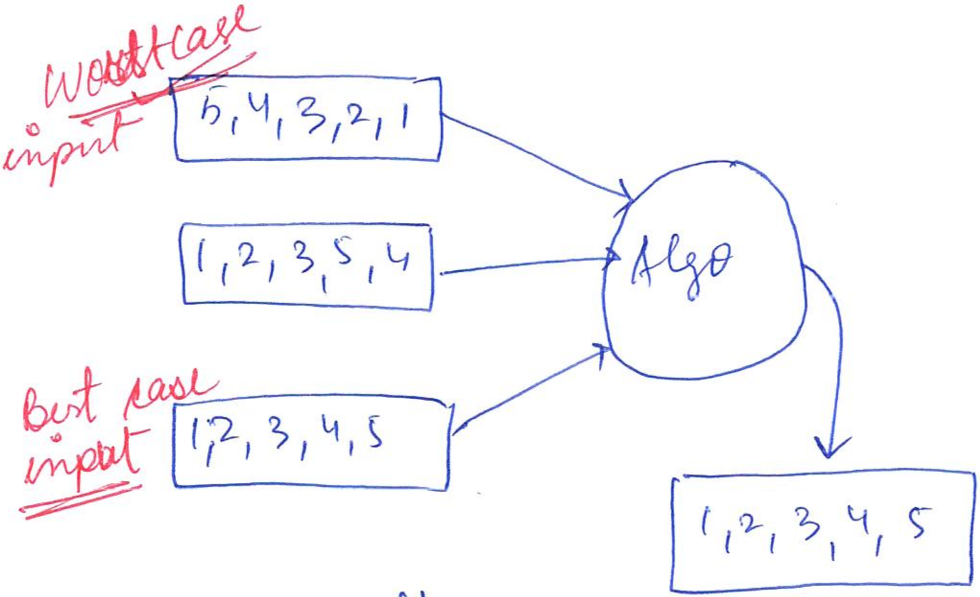
Book 2

② **Application of algorithms** ** | **human genome project.**

Sorting :- Procedure to arrange a set of values either in inc or dec. order..

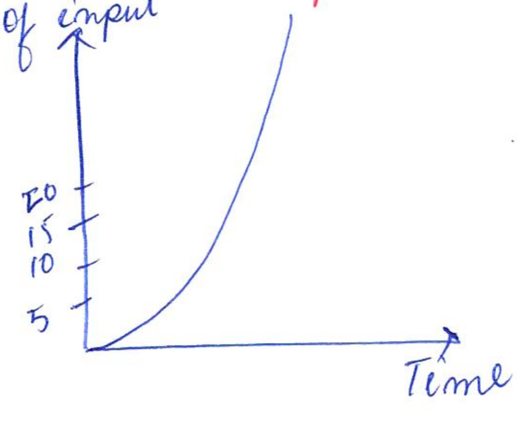
Card Playing
52 cards

Insertion sort :-



How Analysis!!

Machine Dependent | Machine Independent



③

Asymtotic Analysis ③ Impt.

- Big Oh (O)
- Big Otheta (θ)
- Big Omega (Ω)

} measure

Analysing insertion sort

Insertion Sort (A).

5th March, 2025

```
{
    for j = 2 to A.length
        key = A[j]
        // Insert A[j] into the sorted sequence
              A[1...j-1]
        i = j-1.
        while i > 0 and A[i] > key
            A[i+1] = A[i]
            i = i-1.
        A[i+1] = key.
}
```

2 5 4 6 1 3

1	2	3	4	5	6
5	2	4	6	1	3

i j

key = ②

j = 3
key = A[3]
key =

④

Insertion-Sort (A)	cost	times
1. for $j = 2$ to A.length	c_1	n
2. key $= A[j]$	c_2	$n-1$
3. // insert $A[j]$ into the sorted sequence $[A[1 \ldots j-1]]$	0	$n-1$
4. $i = j-1$	c_4	$n-1$
5. while $i > 0$ and $A[i] >$ key	c_5	$\sum_{j=2}^{n} t_j$
6. $A[i+1] = A[i]$	c_6	$\sum_{j=2}^{n} (t_j - 1)$
7. $i = i-1$	c_7	$\sum_{j=2}^{n} (t_j - 1)$
8. $A[i+1] =$ key.	c_8	$n-1$.

The running time of an algorithm is the sum of running times for each statement executed, a statement that takes c_i steps to executes n times will contribute $c_i * n$ to the total running time.

Let $t(n)$ be the total running time.

$$t(n) = c_1 * n + c_2 \times (n-1) + 0 * (n-1) + c_4 * (n-1) + c_5 * \left(\sum_{j=2}^{n} t_j\right) + c_6 * \left(\sum_{j=2}^{n} (t_j - 1)\right) + c_7 * \left(\sum_{j=2}^{n} (t_j - 1)\right) + c_8 \times (n-1) \quad \text{—— (1)}$$

⑤

Best case.

1	2	3	4	5

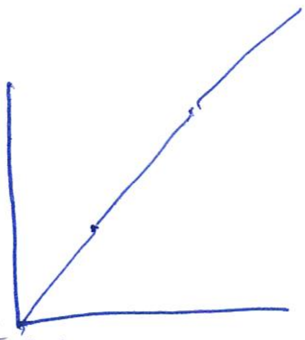
$t_j = 1$ for $j = 2, 3, \ldots, n$.

$$\underbrace{1 + 1 + 1 + 1 + 1}_{n}$$

Accordingly. Eq ① becomes.

$$T(n) = c_1 * n + c_2 * (n-1) + c_4 * (n-1) + c_5(n-1) + c_8(n-1)$$

$$= (c_1 + c_2 + c_4 + c_5 + c_8)\, n - (c_2 + c_4 + c_5 + c_8)$$



= best case the linear function.

Thus, total running time is a linear function of n.

<u>Worst case Scenario</u>

5	4	3	2	1

reverse sorted order.

$$\left.\begin{array}{ll} t_j = 1 & j = 2 \\ t_j = 2 & j = 3 \\ t_j = 3 & j = 4 \end{array}\right\}$$

$$\sum_{j=2}^{n} j = \frac{n(n+1)}{2} - 1.$$

$$\sum_{j=2}^{n} (j-1) = \frac{n(n+1)}{2}$$

so, $t_j = j$ for $j = 2, 3, \ldots, n$.

So, accordingly. ① becomes

$$T(n) = c_1 * n + c_2(n-1) + c_4 * (n-1) + c_5\left(\frac{n(n+1)}{2} - 1\right) + c_6 * \left(\frac{n(n-1)}{2}\right) + c_7 * \left(\frac{n(n+1)}{2}\right) + c_8 * (n-1)$$

⑥

$$= \left(\frac{c_5}{2} + \frac{c_6}{2} + \frac{c_7}{2}\right)n^2 + \left(c_1 + c_2 + c_4 + \frac{c_5}{2} - \frac{c_6}{2} - \frac{c_7}{2} + c_8\right)n$$

$$-\left(c_2 + c_4 + c_5 + c_8\right).$$

Thus, the total running time is a quadratic function of 'n'.