



In worst case ie either the element we are searching is not present or at extremes. The algorithm divides the array into 2 halfs after every comparison until the array contains only one element. Since the algorithm divides the array by 2 till we compare it with single element, the number of comparisons = log, n

Hence Thorst Case Time Complexity = O(log, n)

c. In large Scale library due to huge number of books we need to locate quickly by its unique ISBN number.

Implementing linear search would be inefficient because in linear search we have to check every book until we find the target for the worst case. If there are n books, we need n comparisons thence it is inefficient Binary search technique on the other hand would require less number of comparisons since after every comparison the search space gets halved. The halving process continues until we find the book

Hence worst case time complexity Binary Search = O(1091)

D. Space Complexity of Binary Search for iterative approach.

The iterative approach does not use any extra space.

It uses only some space for integer variables like.

1eft right mid. Since these use constant space.

the space complexity = O(1)

Page	No.		
Date			

Mathematical Expression:

- We start with an array of nelements. The target is first compared with middle element of the array.

If the target is not found after first comparison, the array is divided into half and either of one is selected for further comparison based on whether the target is small or greater than middle element.

Now we have n elements to search

Again the target is compared with the middle element of array that have n elements.

If the target is not found even after second comparison, we again divide the array by 2. Hence now the total elements = n

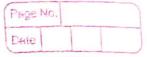
If we perform this division until we get a single element, let consider the total number of comparsions to compare it with array of 1 element = KHence, $\frac{n}{n} = 1$

 \Rightarrow $n = 2^k$ take log, on both sides

 $log_2 n = log_2 2^k$

=) log_n = 1 log_2

=) K = log_n



Application:
Application:
and application where pinary search is
The diction of the source
Jak A-7 INPOUNT MEG COSCO
first page and compare every word. After every
thrst page and compare they have the search space.
comparison we narrow down the search space.
we continue this process until the target word is
found. It requires less comparisons compared to
linear cearch technique.
Binary search technique help make searches faster
and efficients a grant tout your head was
and efficience
The state of the s
Language to the same to the same and the
, H. C. i vnym alt about ainon sut month
n= diminal lata + Add mod
this release litera a distribution of the market san the