Recursive Algorithms

CS2002D Program Design

Recursive Linear Search

LINEAR_SEARCH(A, Key, n)

```
1. if(n=0)
2.    return -1
4. else if (A[n]=Key)
5.    return n
6. else
7.    return LINEAR_SEARCH(A, Key, n-1)
```

Recursive Linear Search - Analysis

$T(n) = T(n-1) + c \qquad T(1) = 1$

$T(n) = T(n-2) + c + c$

$T(n) = T(n-2) + 2c$

$T(n) = T(n-k) + kc$

When $k=n-1$, $T(n) = T(n-(n-1)) + (n-1)c$

$T(n) = (n-1)c + 1$

Binary Search

Binary-Search(A, key, low, high)

1. if (low>high)
    a. Return -1
2. else
    a. Find mid = (low + high) / 2
    b. if(A[mid]=key)
        i. Print the search is successful and position is mid
    c. else if(A[mid]<key)
        i. Binary-Search(A, key, mid+1, high)
    d. else
        i. Binary-Search(A, key, low, mid-1)

Recursive Binary Search - Analysis

$T(n) = T(n/2) + c \qquad T(1) = c$

$T(n/2) = T(n/4) + c$

$T(n) = T(n/4) + 2c$

$T(n) = T(n/2^k) + kc$

When $n=2^k$, $T(n) = T(n/(2^k)) + kc$

$T(n) = kc + 1 = \log_2(n)c + 1$ $\qquad k=\log_2(n)$

Thank You !!!