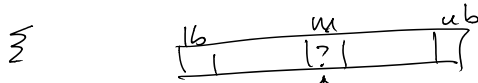


Exam 1 - Monday 29th in class
 word search (not Key[], int n, int v)
 Allow 1 page of notes (2 sides of
 1 8.5" x 11")



① int lb = 0;
 ① int ub = n - 1;
 ① int m;
 bool found = false;

while (lb ≤ ub) ub - lb + 1 = n
 m = (lb + ub) / 2;
 if (Key[m] == v) ✓
 found = true;
 break;
 else if (Key[m] < v) ✓
 lb = m + 1;
 else ✓
 ub = m - 1;

n/2
 n/4

n/2^k

n/2^k = 1

return found;

n = 2^k

log₂ n = k

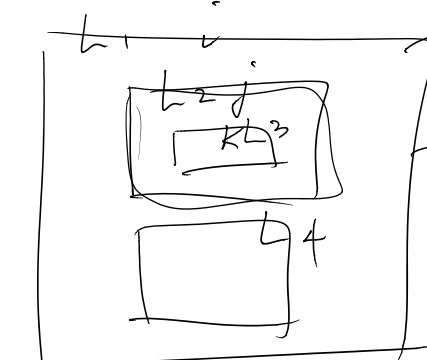
ET(n) = O(g(n)) = O(log(n))

~~T(n) = 5 log₂(n) + 3~~

T(n) = O(log₂(n))

T(n) = O(log(n))

$$T(n) = O(\log(n))$$



$$T(n) = L_1 \cdot L_2 \cdot L_3 + L_1 \cdot L_4$$

$$= O(n^3)$$

```

for ( i = 1; i <= n; i++)
{
    for ( j = 1; j <= i; j++)
    {
        stmt;
    }
}

```

$$T(n) = \sum_{i=1}^n \left(1 + \sum_{j=1}^i (1 + 1) \right)$$

$$= \sum_{i=1}^n (1 + 2 \cdot (i - 1))$$

$$= \sum_{i=1}^n (2i + 1)$$

$$= \sum_{i=1}^n (2i) + \sum_{i=1}^n (1)$$

$$\begin{aligned}
 &= \sum_{i=1}^n (2i) + \sum_{i=1}^n (1) \\
 &= 2 \sum_{i=1}^n (i) + \sum_{i=1}^n (1) \\
 &= 2 \left(\frac{n(n+1)}{2} + n \right)
 \end{aligned}$$

$$\begin{aligned}
 T(n) &= n^2 + 2n \\
 &= O(n^2)
 \end{aligned}$$

```

for ( i = 1; i <= n; i++ )
{
    for ( j = 1; j <= i; j++ )
    {
        stmt;
    }
}

```

```
bool bsearch(int key[], int lb, int ub, int v)
```

$T(n)$

```
if (lb <= ub)
```

```
int m = lb + ub / 2;
```

```
if (key[m] == v)
```

```
return true;
```

```
else if (key[m] < v)
```

```
return bsearch(key, m+1, ub, v);
```

```
else return bsearch(key, lb, m-1, v);
```

$T(n/2)$

$T(n/2)$

3

$$T(n) = K + T(n/2)$$

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

$$T(n/4) = K + T(n/8)$$

$$T(n/2^{k-1}) = K + T(n/2^k)$$

$\log_2 n$

$$T(n) = K + K + T(n/4)$$

$$= K + K + K + T(n/8)$$

$$= K + K + K \dots + T(0)$$

$$= K + K + K \dots + K$$

$\log_2(n)$

$$\begin{aligned}
 &= \cancel{K} \log_2(n) + \cancel{c} \\
 &= O(\log n)
 \end{aligned}$$

```

if ( base case )
    stmt;
else
    recursive
  
```

```

void bubble (int key[], int n)
{
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n - i - 1; j++)
        {
            if (key[j] > key[j+1])
                swap(key[j], key[j+1]);
        }
    }
}
  
```

Diagram illustrating the bubble sort process: A horizontal array of 6 cells. The first two cells contain circles with 'x' marks, and the last cell contains a circle with an 'x' mark. A dashed line connects the first two cells. Above the array, the indices 0 and n-1 are marked at the ends.

3 3

$$T(n) = O(n^2)$$

$$T(n) = 3n^2 + 2n + 1$$

$$\begin{matrix} n^2 \\ \log n \\ 0 \end{matrix} \left[\begin{array}{l} \text{bubbleSort}(key, n) \\ \text{cout} \ll \text{bsearch}(key, n, 50); \\ \text{cout} \ll \text{bsearch}(key, n, 20); \end{array} \right];$$

$$\begin{aligned} T(n) &= O(n^2 + \cancel{2 \log n}) \\ &= O(n^2) \end{aligned}$$