```
// Size of the problem is n = hi - lo.
//You can assume n = 2^p for some integer p.
int f4(int a[], int lo, int hi, int x){
   if(lo >= hi){
      return -1;
   }
   int m = lo + (hi - lo)/2;
   if(a[m] < x){
      return f4(a, lo, m, x);
   }else{
      return f4(a, m, hi, x);
   }
}
```

suppose $lo = 10$ and $hi = 74$, $n = 74 - 10 = 64$.

$f_4(a, 10, 74, x)$ $\quad n = 74 - 10 = 64$

$m = 10 + 64/2 = 42$

$n = 42 - 10 = 32$ $\quad f_4(a, 10, 42, x)$ $\qquad$ $n = 74 - 42 = 32$ $\quad f_4(a, 42, 74, x)$

$m = 10 + 16 = 26$ $\qquad$ $m = 42 + 32/2 = 58$

$n=16$ $\quad f_4(a, 10, 26, x)$ $\qquad$ $n=16$ $\quad f_4(a, 26, 42, x)$ $\qquad$ $n=16$ $\quad f_4(a, 42, 58, x)$ $\qquad$ $n=16$ $\quad f_4(a, 58, 74, x)$

- The recursive execution will follow one path depending on the value of x and on the values stored in a.
  - ↳ for example the green path to the left.

- At each level of the recursion, a constant amount of work is performed: a pair of tests, a fixed number of arithmetic integer operations and assignments, and a call/return.
- There are $\log_2(n)$ recursive calls.

The tree has $2n-1$ nodes.

$$\underbrace{\underset{2^0}{1} + \underset{2^1}{2} + 4 + \cdots + \underset{2^{\log_2 n}}{n}}_{\log_2(n)+1} = 2^{\log_2 n + 1} - 1 = 2n - 1$$

↳ f_4 is therefore in $\Theta(\log_2 n)$ regardless of the values stored in a and of the value of x.