Algorithms with order $\log_2 n$

Analysis of Nonrecursive Algos

Example 1: The **number of binary digits** in the binary representation of a positive integer.

Algorithm Binary(n)

// Input: A positive decimal integer n

//Output: The number of binary digits in the bin. repr.

$\text{count} \leftarrow 1$

while $n > 1$ **do**

$\quad \text{count} \leftarrow \text{count} + 1$

$\quad n \leftarrow \lfloor n/2 \rfloor$

return count

Analysis of Nonrecursive Algos

- **The basic operation:** the number of times of the **comparisons** executed (which is larger than the number of repetitions of the loop's body by exactly one)

- **The input size:** the loop variable n takes only a few values between its lower and upper limits:

$$n, \left\lfloor \frac{n}{2} \right\rfloor, \left\lfloor \frac{n}{4} \right\rfloor, \left\lfloor \frac{n}{8} \right\rfloor, \ldots, 1$$

- **The total number:** In each loop repetition, n is about halved, thus,

the total $\# \approx \log_2 n$

Analysis of Recursive Algos

Example 2: Find the **number of binary digits** in the binary representation of a positive integer.

Algorithm BinRec(n)

// Input: A positive decimal integer n

//Output: The number of binary digits in the bin. repr.

if $n = 1$ **return** 1

else return BinRec$(\lfloor n/2 \rfloor) + 1$

Analysis of Recursive Algos

Example 2: Find the **number of binary digits** in the binary representation of a positive integer.

Algorithm BinRec(n)

// Input: A positive decimal integer n

//Output: The number of binary digits in the bin. repr.

if $n = 1$ **return** 1

else return BinRec$(\lfloor n/2 \rfloor) + 1$

Let $A(n)$ be the **total number of additions.**

$$A(n) = A(\lfloor n/2 \rfloor) + 1, n > 1, \qquad A(1) = 0$$

Analysis of Recursive Algos

For the simplicity we choose $n = 2^k$ (the smoothness rule)

$$A(2^k) = A\left(\frac{2^k}{2}\right) + 1 = A(2^{k-1}) + 1$$

$$= \left(A(2^{k-2}) + 1\right) + 1 = A(2^{k-2}) + 2$$

$$= \left(A(2^{k-3}) + 1\right) + 2 = A(2^{k-3}) + 3$$

$$= \cdots$$

$$= A(2^{k-i}) + i$$

$$= \cdots$$

$$= A(2^{k-k}) + k = A(1) + k = k = \log_2 n$$