LEC. 15 09/29/99

RECURSIVE ALGORITHMS

BY THE EUCLIDEAN ALGORITHM

$gcd(a,b) = gcd(b, r_2)$ CIRCULAR PROCEDURE? No!

THIS GUARANTEES THE CONVERGENCY ⇐ A SMALLER INPUT $\min(a,b) \geq \min(b, r_2)$

def AN ALGORITHM IS CALLED RECURSIVE IF IT WORKS BY REDUCING TO ITS OWN VALUE ON SMALLER INPUTS (USUALLY PERFORMED BACKWARD)

EXAMPLE $gcd(a,b) = gcd(b, r_2)$ WHERE $a \geq b > 0$

$a = b \cdot q_1 + r_2 \quad 0 \leq r_2 < b$

$$\begin{cases} gcd(a,b) = gcd(b, a \bmod b) \\ gcd(d,0) = d \text{ when } d > 0 \end{cases}$$

SPECIFIES A SEQUENCE OF REDUCTIONS THAT ENDS by $gcd(d,0) = d$

RECURSIVE COMPUTING OF a^n

$$\begin{cases} a^0 = 1 \\ a^n = a\, a^{n-1} \end{cases}$$ ← THE SAME PROCEDURE, SMALLER INPUT

```
PROCEDURE POWER(a: NONZERO REAL, n ≥ 0 INTEGER)
IF n=0 THEN POWER(a,n)=1
ELSE POWER(a,n) := a * POWER(a, n-1)
```

RECURSIVE SEQUENTIAL SEARCH ALGORITHM: FIND x IN $a_1, \dots, a_n$

```
PROCEDURE SEARCH(i,n,x)
IF a_i = x THEN
    location := i
ELSE IF i=n THEN
    location := 0
ELSE
    SEARCH(i+1, n, x)
```

SEARCHES FOR x IN $a_i, a_{i+1}, \dots a_n$

DECREASING PARAMETER $(n-i)$ IS ONE LESS AFTER EACH ITERATION

⇒ CONVERGES

Th. IF $f(n)$ IS DEFINED RECURSIVELY

$$\begin{cases} f(0) = a \\ f(n) = \text{SOME FUNCTION } G \text{ of } f(0), \dots, f(n-1) \end{cases}$$

AND G IS COMPUTABLE, THEN f IS COMPUTABLE.

PROOF. COMPUTE $f(0), f(1), f(2), \dots, f(n)$

def ITERATIVE PROCEDURE P:

$$\begin{cases} \text{INPUT } a \\ P_n(a) = P(P_{n-1}(a)) \end{cases}$$ USUALLY PERFORMED STRAIGHT

$a, P(a), P(P(a)), \dots \underbrace{P(P(P\dots(Pa)\dots)}_{n \text{ TIMES}}$

RECURSIVE

```
PROCEDURE factorial(n)
IF n=1 THEN
  factorial(n) := 1
ELSE
  factorial(n) :=
     n * factorial(n-1)
```

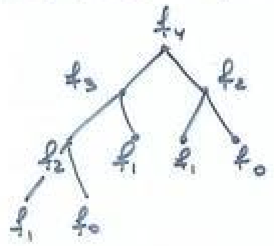
ITERATIVE

```
PROCEDURE iterfactorial(n)
x := 1
FOR i:=1 TO n
     x := i * x
{x is n!}
```

```
PROCEDURE fibonacci(n)
IF n=0 THEN fibonacci(n):=0
ELSE IF n=1 THEN
          fibonacci(n):=1
ELSE fibonacci(n) :=
fibonacci(n-1)+fibonacci(n-2)
```

```
PROCEDURE itfibonacci(n)
IF n=0 THEN y:=0
ELSE
BEGIN x:=0 y:=1
    FOR i=1 to n-1
    BEGIN z:=x+y, x:=y, y:=z
    END
END {y is nth Fibonacci num}
```


RECURSIVE FIBONACCI COMPUTATION



$f_{n+1} - 1$ ADDITIONS TO FIND f_n

(EXERCISE!)

EXPONENTIAL

ITERATIVE FIBONACCI COMPUTATION

$f_0, f_1, f_2 = f_1 + f_0, f_3 = f_2 + f_1, \dots, f_n = f_{n-1} + f_{n-2}$

$n-1$ ADDITIONS / LINEAR

HW 3.4: 4, 14