

**Resolución de Problemas y Algoritmos**

**Clase 16**  
**Planteos recursivos.**  
**Funciones y Procedimientos Recursivos**



**Dr. Alejandro J. García**

http://cs.uns.edu.ar/~ajg



Departamento de Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur  
Bahía Blanca - Argentina

**(Repaso) Algoritmos recursivos**

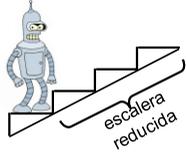
- Un algoritmo recursivo **será válido**, si:
  - (a) **existe un caso base** que **no** se define en términos de sí mismo, y
  - (b) la **referencia a sí mismo es relativamente más sencilla o reducida** que el caso considerado.

**Ejemplo: subir una escalera**

(a) Caso base: no hay escalones

(b) Caso general:

- sube un escalón
- y luego queda por subir una escalera que tiene un escalón menos, y por lo tanto es más reducida.



Resolución de Problemas y Algoritmos Dr. Alejandro J. García 2

**Metodología propuesta**

- Identificar **ejemplos significativos** que ayuden a entender el problema y su solución.
- Realizar un **planteo recursivo** en el cual se distinga el "caso base", y el "caso general" (donde se define en términos de sí mismo pero para una instancia más simple/reducida/menor).
- Verificar que el planteo es correcto (con alguno de los ejemplos significativos).
- Determinar si se realizará una **función** o un **procedimiento recursivo**, e implementarlo en Pascal.
- Realizar la **traza** de la primitiva en Pascal.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 3

**Recursión**

- Las función factorial (!) tiene una definición recursiva:
 
$$N! = \begin{cases} 0! = 1 \\ N! = N * (N-1)! \end{cases}$$
- Observe que tiene un **caso base** y otro caso que se define con una **instancia más simple** de sí mismo.

**Factorial de N**

- Caso base: si N es 0 el factorial es 1
- Caso general : si N > 0 factorial de N es N multiplicado por el **factorial de N-1**

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 4

```

program Factorial-Recursivo;
var num, fact: integer;

Function Factorial (N:integer):integer;
var aux: integer;
begin
  IF N=0 THEN Factorial:=1
  ELSE begin
    aux:= Factorial(N-1);
    Factorial:= N * aux;
  end;
end;

begin
  writeln(' Ingrese un número'); Readln(Num);
  fact:=factorial(Num);
  writeln(' Factorial de ', Num, ' es ', Fact );
end.
    
```

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 5

**Otras dos formas de implementar la función**

```

Function Factorial (N:integer):integer;
var aux: integer;
begin
  IF N=0
  THEN aux:=1
  ELSE aux:= N* Factorial(N-1);
  Factorial:= aux;
end;

Function Factorial (N:integer):integer;
begin
  IF N=0
  THEN Factorial :=1
  ELSE Factorial := N* Factorial(N-1);
end;
    
```

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 6

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:  
**"Resolución de Problemas y Algoritmos. Notas de Clase". Alejandro J. García. Universidad Nacional del Sur. (c)1998-2012.**

### Planteos Recursivos

**Tarea propuesta:**

- 1) Realice la traza (en papel) del programa anterior.,
- 2) Vea su ejecución en la computadora.
- 3) Realice las trazas de los programas Repaso2 y Repaso3 vistos la clase pasada y compare con la traza realizada en (1).
- 4) Vea la ejecución de estos programas en la computadora.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    7

### Planteo Recursivo

**Enviar una lista de e-mails (del buzón de salida)**

- **Caso base:** si la lista tiene un único e-mail, entonces enviarlo.
- **Caso general :** si la lista tiene más de un e-mail, entonces, enviar el primero y luego enviar la lista de e-mails restante (del buzón de salida).
- **Observe** que con muy pocas modificaciones el planteo podría resolver otros problemas análogos, como enviar el mismo SMS a una lista de destinatarios, o varios SMS pendientes, etc. Esto es, la solución de este problema nos servirá en el futuro para resolver otros problemas análogos en forma recursiva.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    8

### Planteo recursivo: buscar elemento en una secuencia

**(opción 1) con caso base: 1 elemento**

**Buscar el elemento X en la secuencia S:**

- **Caso base:** Si la secuencia tiene un único elemento entonces; si ese elemento es X retornar verdadero, de lo contrario retornar falso.
- **Caso general :** Si la secuencia tiene más de un elemento entonces: si el primer elemento de S es X retornar verdadero, de lo contrario buscar el elemento X en la secuencia S' que es S sin su primer elemento.
- **Observe** que el caso base podría ser también la secuencia vacía.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    9

### Planteo recursivo: buscar elemento en una secuencia

**(opción 2) caso base: secuencia vacía**

**Buscar el elemento X en la secuencia S:**

- **Caso base:** Si la secuencia es vacía retornar falso.
- **Caso general :** Si la secuencia tiene más de un elemento entonces: si el primer elemento de S es X retornar verdadero, de lo contrario buscar el elemento X en la secuencia S' que es S sin su primer elemento.
- Este planteo permite tener como una entrada válida una secuencia vacía, y probablemente se adapte mejor a buscar un elemento en un archivo secuencial de Pascal.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    10

### Problema propuesto: dígito más significativo

- El dígito más significativo de un número es el dígito de más a la izquierda.  
**Ejemplos:** dms(1835)=1, dms(5381)=5, dms(3)=3 y dms(0)=0.
- Escriba una primitiva recursiva para obtener el dígito más significativo de un número natural N.

**Solución recursiva para dms(N)**

- **Caso base:** N tiene un dígito, dms(N) = N
- **Caso general:** N tiene más de un dígito, dms(N) = dms(N div 10)

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    11

### Función recursiva: dígito más significativo

```

PROGRAM dig_mas_sig;
var numero:integer;
FUNCTION dms (N:integer):integer;
begin
  if (N >= 0) and (N <= 9) {tiene un dígito}
  then dms:=N
  else dms:= dms(N div 10);
end;
begin
  writeln('ingrese un número: ');readln(numero);
  writeln('dms es ', dms(numero));
  readln;
End. {tarea: escriba un procedimiento "dms"}
    
```

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    12

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:  
 "Resolución de Problemas y Algoritmos. Notas de Clase". Alejandro J. García. Universidad Nacional del Sur. (c)1998-2012.