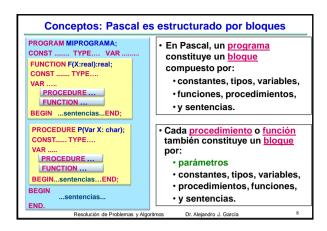
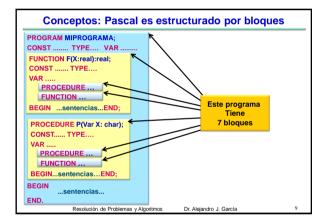
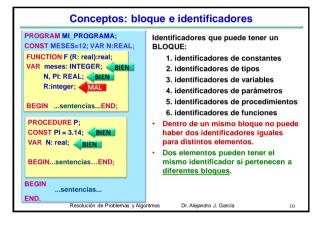


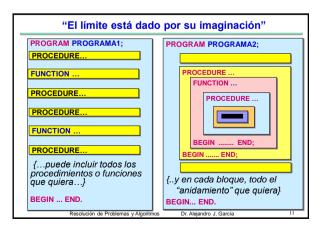
El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:

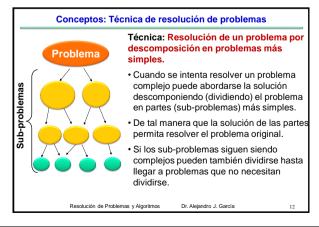
## Preguntas para reflexionar La siguientes preguntas son sobre el programa "reflexion4", (antes de responderlas tiene que hacer la traza) • (fácil): ¿cuáles son parámetros por referencia? • La variable v2 no tiene valor al ser usada en el parámetro efectivo de la llamada a P1, ¿es un error de programación? • La variable v1 si tiene valor ¿es un error? ¿es mejor? • ¿Qué ocurriría si en P2 no se hiciera C:=0?



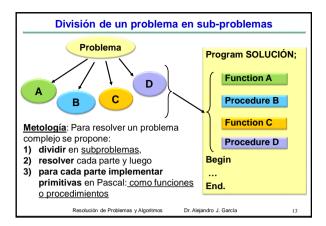


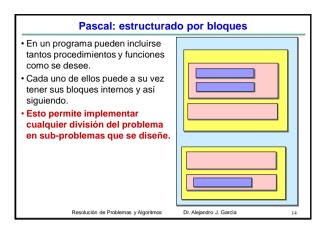


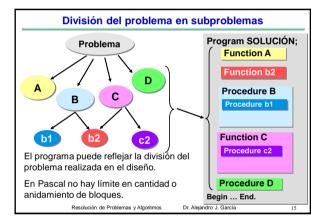




El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:







### Conceptos: bloques e identificadores Cada procedimiento y función determina un nuevo bloque. En cada bloque se puede tanto declarar nuevos identificadores como usar identificadores. En esta clase se introducen las reglas que definen cuales identificadores son visibles para un bloque (i.e., pueden usarse) aunque estén declarados en otros bloques del programa. A continuación se mostrará un programa en Pascal (llamado simple) con el objetivo de ejemplificar los nuevos conceptos que surgen de utilizar procedimientos y funciones. El programa no resuelve ningún problema en particular, está construido desde un punto de vista didáctico para mostrar la mayor cantidad de declaración y uso de identificadores.

Dr. Alejandro J. García

Resolución de Problemas y Algoritmos

```
Ejemplo: bloques (demarcados con un recuadro)
program simple; {para entender los conceptos}
const Pi = 3.14; type Tdig=0..9; var A, B, C:CHAR;
  PROCEDURE P1 (A:REAL);
  var B: REAL; F2: Tdig;
  begin B:= A; WRITE(B) end; {P1}
   PROCEDURE P2 (A:REAL);
  var B, MIA: real;
      FUNCTION F2 (A:REAL):REAL;
     var B, DE_F2: REAL;
     begin B:= A; F2:= B + Pi; end; {F2}
  begin B:= A; WRITE(F2(A)); P1(B) end; {P2}
begin
   P2(5); P1(10);
           Resolución de Problemas y Algoritmos
                                    Dr. Alejandro J. García
```

```
Repaso: diferentes elementos de un programa
program simple; {para entender los conceptos}
                                                     (1) Palabras
const Pi = 3.14; type Tdig =0..9; var A, B, C: CHAR;
                                                     reservadas
   PROCEDURE P1 (A: REAL);
   var B: REAL; F2: Tdig;
                                                     (2) Símbolos
                                                     y valores.
  begin B:= A; WRITE (B) end; {P1}
   PROCEDURE P2 (A: REAL);
                                                     (3) Comenta-
                                                     rios.
   var B. MIA: real:
      FUNCTION F2 (A: REAL): REAL;
                                                     (4) Identifica-
      var B, DE_F2 : REAL;
                                                     dores.
      begin B := A; F2 := B + Pi ; end; {F2}
   begin B:=A; WRITE (F2(A)); P1 (B) end; {P2}
begin
   P2 (5); P1 (10);
            Resolución de Problemas y Algoritmos
                                     Dr. Alejandro J. García
```

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:

```
Repaso: diferentes elementos de un programa
program simple; {para entender los conceptos}
                                                    (1) Palabras
const Pi = 3.14; type Tdig =0..9; var A, B, C: CHAR;
                                                    reservadas:
  PROCEDURE P1 (A: REAL);
                                                    tienen un
  var B: REAL; F2: Tdig;
                                                    significado
  begin B:= A; WRITE (B) end; {P1}
                                                    propio y el
                                                    programador
   PROCEDURE P2 (A : REAL);
                                                    no puedo
  var B. MIA: real:
                                                    cambiarlo.
     FUNCTION F2 (A: REAL):REAL;
     var B, DE F2 : REAL;
     begin B := A; F2 := B + Pi ; end; {F2}
  begin B:=A; WRITE (F2(A)); P1 (B) end; {P2}
begin
   P2 (5); P1 (10);
end.
           Resolución de Problemas y Algoritmos
                                   Dr. Alejandro J. García
```

```
Repaso: diferentes elementos de un programa
program simple; {para entender los conceptos}
                                                    (2) Símbolos
const Pi = 3.14; type Tdig =0..9; var A, B, C: CHAR;
                                                    y valores:
   PROCEDURE P1 (A: REAL);
                                                    tienen un
                                                    significado
   var B: REAL; F2: Tdig;
                                                    propio y el
  begin B:= A; WRITE (B) end; {P1}
                                                    programador
   PROCEDURE P2 (A : REAL);
                                                    no puede
  var B. MIA: real:
                                                    cambiarlo.
      FUNCTION F2 (A: REAL): REAL;
      var B, DE F2 : REAL;
      begin B := A; F2 := B + Pi ; end; {F2}
   begin B:=A; WRITE (F2(A)); P1 (B) end; {P2}
begin
   P2 (5); P1 (10);
end.
           Resolución de Problemas y Algoritmos
                                    Dr. Alejandro J. García
```

```
Repaso: diferentes elementos de un programa
program simple; {para entender los conceptos}
                                                    (4) Identifica-
const Pi = 3.14; type Tdig =0..9; var A, B, C: CHAR;
                                                    dores.
   PROCEDURE P1 (A: REAL);
                                                    tienen el
                                                    significado
   var B: REAL; F2: Tdig;
                                                    que quiera el
  begin B:= A; WRITE (B) end; {P1}
                                                    programador.
   PROCEDURE P2 (A : REAL);
                                                    Algunos son
  var B. MIA: real:
                                                    predefinidos.
      FUNCTION F2 (A: REAL):REAL;
      var B, DE_F2 : REAL;
                                                    Observación:
                                                    a los
      begin B := A; F2 := B + Pi ; end; {F2}
                                                    predefinidos
  begin B:=A; WRITE (F2(A)); P1 (B) end; {P2}
                                                    es posible
begin
                                                    cambiarle su
   P2 (5); P1 (10);
                                                    significado.
end.
            Resolución de Problemas y Algoritmos
                                    Dr. Alejandro J. García
```

```
Ejemplo: bloques (demarcados con un recuadro)
program simple; {para entender los conceptos}
const Pi = 3.14; type Tdig=0..9; var A, B, C:CHAR;
   PROCEDURE P1 (A:REAL);
                                              Escriba en sus notas
   var B: REAL; F2: Tdig;
                                                 este programa
                                             (mientras lo copiamos
  begin B:= A; WRITE(B) end; {P1}
                                                 en el pizarrón)
   PROCEDURE P2 (A:REAL);
   var B, MIA: real;
      FUNCTION F2 (A:REAL):REAL;
      var B, DE F2: REAL;
      begin B:= A; F2:= B + Pi; end; {F2}
   begin B:= A; WRITE(F2(A)); P1(B) end; {P2}
begin
   P2(5); P1(10);
end
           Resolución de Problemas y Algoritmos
                                    Dr. Alejandro J. García
```

```
Preguntas sobre el programa "simple"

· ¿puedo llamar a P1 desde las sentencias de P2?

· ¿puedo llamar a F2 desde las sentencias de P2?

· ¿puedo llamar a F2 desde las sentencias de P1?

· ¿puedo llamar a P1 desde las sentencias de P1?

· ¿puedo llamar a P1 desde las sentencias de F2?

· ¡ HAGA AHORA SUS PREGUNTAS!
(y copie las de sus compañeros)

· Pregunta más general: ¿desde qué lugar del programa puedo llamar a una función o procedimiento?

· ¿en qué bloques puedo usar la variable "MIA"?

· ¿y la variable DE_F2?

· ¿en qué bloques puedo usar una variable?

· Todas las respuestas en la teoría que sigue a continuación...
```

```
Conceptos: declaración vs. referencia

Es importante distinguir entre:

1. La declaración de un identificador de constante, tipo, variable, parámetro, función, o procedimiento. Ejemplos: CONST pi = 3.14; TYPE Tdig = 0..9; VAR precio: real; PROCEDURE recargo (precio,rec: real; var monto: real);

2. La referencia o el uso de un identificador. Ejemplos: recargo (24, incremento, precio); a_pagar := precio + intereses (round ( precio ));

• En cada bloque, se declaran identificadores; y además, se hace referencia (usan) identificadores.

• A continuación se muestra para el programa "simple" (1) en primer lugar donde se declaran identificadores y (2) en segundo lugar donde se usan identificadores.
```

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) 18/05/2016

Dr. Alejandro J. García

Resolución de Problemas y Algoritmos

```
Eiemplos de declaración de identificadores
PROGRAM simple; {para entender los conceptos}
Const Pi = 3.14; type Tdig = 0..9; var A, B, C :CHAR;
   PROCEDURE P1 (A: REAL);
   var B: REAL; F2: Tdig;
   begin B:= A; WRITE(B) end; {P1}
   PROCEDURE P2 (A: REAL);
   var B, MIA: real;
      FUNCTION F2 (A: REAL): REAL;
      var B, DE F2: REAL;
      begin B:= A; F2:= B + Pi; end; {F2}
   begin B:= A; WRITE(F2(A)); P1(B) end; {P2}
BEGIN
   P2(5); P1(10);
END.
           Resolución de Problemas y Algoritmos
                                    Dr. Alejandro J. García
```

```
Eiemplos de uso de identificadores
PROGRAM simple; {para entender los conceptos}
Const Pi = 3.14; type Tdig = 0..9; var A, B, C : CHAR;
   PROCEDURE P1 (A: REAL);
   var B: REAL; F2: Tdig;
  begin B:= A; WRITE (B) end; {P1}
   PROCEDURE P2 (A: REAL);
   var B. MIA: real:
      FUNCTION F2 (A: REAL): REAL;
     var B, DE F2: REAL;
     begin B:= A; F2:= B + Pi; end; {F2}
   begin B:= A; WRITE(F2(A)); P1(B) end; {P2}
BEGIN
   P2(5); P1(10);
END.
           Resolución de Problemas y Algoritmos
                                    Dr. Alejandro J. García
```

### Concepto: Entorno de referencia para un bloque B

El entorno de referencia de un bloque B está formado por los siguientes cuatro entornos:

- El entorno local: conjunto de identificadores (parámetros formales, constantes, tipos, variables, el nombre de los procedimientos y funciones) declarados dentro del bloque B.
- 2. El <u>entorno global</u>: conjunto de identificadores <u>declarados</u> en el bloque del programa principal.
- El entorno no-local: conjunto de identificadores declarados en los bloques que contienen al bloque B, exceptuando al global.
- El entorno predefinido: conjunto de identificadores ya declarados por el compilador de Pascal y disponible para todo programa (Ejemplos de identificadores predefinidos: maxint, char, write, eof).

Ejemplo: considere el programa simple mostrado antes, indique cuales son sus bloques y el entorno de referencia de cada bloque.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García

### Ejemplos de entornos de referencia

- El programa "simple" tiene 4 bloques: P1,
  - P2,

F2

y el bloque del programa "simple".

- A continuación se muestran los entornos de referencia para cada uno de estos bloques.
- Observe que el entorno predefinido y el entorno global es siempre el mismo para todos.

Resolución de Problemas y Algoritmos

Dr. Alejandro J. García

### Los cuatro bloques en distintos colores PROGRAM simple; {para entender los conceptos} Const Pi = 3.14; type $\underline{\text{Tdig}} = 0..9$ ; $\underline{\text{var A}}$ , B, C :CHAR; PROCEDURE P1 (A: REAL); var B: REAL; F2: Tdig; begin B:= A; WRITE (B) end; {P1} PROCEDURE P2 (A: REAL); var B, MIA: real; FUNCTION F2 (A: REAL):REAL; var B, DE\_F2: REAL; begin B:= A; F2:= B + Pi; end; {F2} begin B:= A; WRITE(F2(A)); P1(B) end; {P2} BEGIN P2(5); P1(10); END. Resolución de Problemas y Algoritmos Dr. Alejandro J. García

### Entorno de referencia para el Bloque "F2" - Entorno local: A, B, DE\_F2 - Entorno no-local: A, B, MIA, F2 (declarados en P2) - Entorno global: Pi, Tdig, A, B, C, P1, P2 - Entorno predefinido: maxint, char, write, etc, ( todos los elementos predefinidos provistos por Pascal). Entorno de referencia para el Bloque "P2" - Entorno local: A, B, MIA, F2 - Entorno no-local: (vacío, no tiene) - Entorno global: Pi, Tdig, A, B, C, P1, P2

Entorno predefinido: todos los elementos predefinidos

Dr. Alejandro J. García

Ejemplos: bloques del programa "simple"

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) 18/05/2016

Resolución de Problemas y Algoritmos

provistos por Pascal.

# Ejemplos: bloques del programa "simple" Entorno de referencia para el Bloque "P1" - Entorno local: A, B, F2 - Entorno no-local: (vacío, no tiene) - Entorno global: Pi, Tdig, A, B, C, P1, P2 - Entorno predefinido: (el mismo siempre para todos) Entorno de referencia para el Bloque "simple" - Entorno no-local: (vacío, no tiene) - Entorno global (y también local): Pi, Tdig, A, B, C, P1, P2 - Entorno predefinido: (el mismo siempre para todos)

### Conceptos: identificadores ocultos

Cuando se hace referencia a un identificador:

- 1. primero se busca en su entorno de referencia local,
- 2. luego en su entorno de referencia no local,
- 3. luego en su entorno de referencia global,
- 4. y finalmente en el entorno de referencia predefinido

Por lo anterior, <u>si hay identificadores iguales en diferentes entornos uno oculta al otro</u>.

- Un identificador de nombre N en un entorno local <u>oculta</u> a todo identificador del mismo nombre N en otro entorno (nolocal, global, predefinido)
- 2. Uno no-local N oculta a otro N global o predefinido,
- 3. Un identificador global N oculta a uno predefinido N

Resolución de Problemas y Algoritmos

Dr. Alejandro J. García

### Conceptos: identificador visible y alcance de un identificador

- Un identificador es referenciable en un bloque, si es parte de su entorno de referencia y no está oculto.
- · Un identificador es visible, si es referenciable.
- El <u>alcance</u> de un identificador D, son aquellas sentencias (o bloques) del programa donde el identificador D es visible.

### Ejercicios propuestos:

- Para cada uno de los cuatro bloques del programa simple, encuentre los identificadores visibles (referenciables).
- Indique el alcance del identificador P1 y el alcance de la variable MIA.

Resolución de Problemas y Algoritmos

Dr. Alejandro J. García

### **Ejemplos**

- La constante Pi es visible (referenciable) en todos los bloques (ya que está en todos los entornos por ser parte del entorno global). Lo mismo ocurre con el procedimiento P1 y el tipo Tdig.
- · La función F2 es visible en P2 y en F2.
- La variable "de\_f2" solamente es visible en F2.

Resolución de Problemas y Algoritmos

Dr. Alejandro J. García

### Ejemplo de identificador oculto PROGRAM PruebaDMS; El nombre del parámetro puede ser TYPE digito = 0..9; igual a uno de una variable global. VAR N, D:Integer; FUNCTION digito\_mas\_significativo(N:integer): digito; BEGIN Aunque tengan el mismo if N < 0 then N:=-1\*N: nombre, los cambios del while (N >= 10) do N:=N div 10; parámetro N no afectarán a digito\_mas\_significativo:= N; la variable global N, ya que END: el parámetro oculta a la variable global. write('Ingrese un número:'); Sugerencia: copie el readIn(N); programa y ejecute en la D:=digito\_mas\_significativo(N); máquina para ver la writeln('el D.M.S. de', N, 'es', D); traza real en pantalla.

### Reflexión final

Como tarea que ayudará a comprender mejor los conceptos que se han compartido en lo anterior, se sugiere que en cada uno de los ejercicios y problemas de los prácticos reflexione sobre:

- Los identificadores declarados y usados en cada bloque.
- · El entorno de referencia de cada bloque.
- ¿En quÉ caso es interesante usar identificadores del entorno predefinido, cuando del entorno global y cuando del entorno local? ¿La misma respuesta vale para tipos, constantes, variables o primitivas?

Resolución de Problemas y Algoritmos

. Alejandro J. García

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente: