

Resolución de Problemas y Algoritmos

Buenas prácticas de programación.



Dr. Diego R. García



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

Conceptos: Pautas para un buen estilo de programación

- En la vida profesional del desarrollo de software, es sumamente importante que un programa (o algoritmo) pueda ser **entendido rápidamente** por una **persona** que lo tiene que leer.
- Para lograr esto, al escribir programas (por más modestos que sean), hay que seguir ciertas **pautas** de **“buen estilo de programación”**.
- En las evaluaciones de RPA tendremos en cuenta las siguientes pautas de buena programación:
 1. **Usar tipos de datos adecuados.**
 2. **Uso de nombres representativos en identificadores.**
 3. **Indentación (del inglés “indent”).**
 4. **Comentarios en el código fuente.**

```
{...entre llaves...} // o al finalizar una línea
```

Buen estilo de programación: tipos de datos apropiados

Tener tipos de datos para las variables permite claridad y abstracción: dos conceptos fundamentales en el desarrollo, mantenimiento y futuras actualizaciones del software. Usar nombres representativos ayuda a comprender el uso que se le da a un dato en un programa.

Ejemplos :

Dato a representar	Nombre de la variable	Tipo de dato predefinido
Sueldo de un empleado	Sueldo_empleado	REAL
Letra inicial del apellido	Inicial_apellido	CHAR
Si es o no es año bisiesto	Es_bisiesto	BOOLEAN
Año de nacimiento	Anio_nacimiento	INTEGER

Uso de nombres representativos



```

IF (x1 > x2)
THEN
IF (x1 > w)
THEN z8 := x1
ELSE z8 := w
ELSE IF (x2 > w)
THEN z8 := x2
ELSE z8 := w
    
```



```

IF (num1 > num2)
THEN
IF (num1 > num3)
THEN maximo := num1
ELSE maximo := num3
ELSE IF (num2 > num3)
THEN maximo := num2
ELSE maximo := num3
    
```

Indentación



```

IF (num1 > num2)
THEN
IF (num1 > num3)
THEN maximo := num1
ELSE maximo := num3
ELSE IF (num2 > num3)
THEN maximo := num2
ELSE maximo := num3
    
```



```

IF (num1 > num2)
THEN
    IF (num1 > num3)
        THEN maximo := num1
        ELSE maximo := num3
    ELSE
        IF (num2 > num3)
            THEN maximo := num2
            ELSE maximo := num3
    
```

Comentarios en el código



```

IF (num1 > num2)
THEN
    IF (num1 > num3)
        THEN maximo := num1
        ELSE maximo := num3
    ELSE
        IF (num2 > num3)
            THEN maximo := num2
            ELSE maximo := num3
    
```



```

{ calcula el máximo entre 3
  números: num1, num2 y num3}
IF (num1 > num2)
THEN
    IF (num1 > num3)
        THEN maximo := num1
        ELSE maximo := num3
    ELSE // num1 <= num2
        IF (num2 > num3)
            THEN maximo := num2
            ELSE maximo := num3
{... la variable máximo ahora tiene
 el mayor valor de los 3....}
    
```

Compare...



```

IF (x1 > x2)
THEN
IF (x1 > w)
THEN z8:= x1
ELSE z8 := w
ELSE IF (x2 > w)
THEN z8 := x2
ELSE z8 := w
    
```



```

{ calcula el máximo entre 3
  números: num1, num2 y num3}
IF (num1 > num2)
THEN
  IF (num1 > num3)
    THEN maximo := num1
    ELSE maximo := num3
  ELSE // num1 <= num2
    IF (num2 > num3)
      THEN maximo := num2
      ELSE maximo := num3
{... la variable máximo ahora tiene
  el mayor valor de los 3....}
    
```

En papel...



```

{ calcula el máximo entre 3  números: num1, num2 y num3}
IF (num1 > num2)
THEN
  IF (num1 > num3)
    THEN máximo := num1
    ELSE máximo := num3
  ELSE // num1 <= num2
    IF (num2 > num3)
      THEN máximo := num2
      ELSE máximo := num3
{... la variable máximo ahora tiene  el mayor valor de los 3....}
    
```



Si está trabajando sobre un papel o pizarrón (práctico, examen, etc), también puede usar líneas demarcadoras