



Pseudocódigo

Definición:

En ciencias de la computación, y análisis numérico, el **pseudocódigo** (o falso lenguaje) es una descripción de alto nivel compacta e informal del principio operativo de un programa informático u otro algoritmo.

Utiliza las convenciones estructurales de un lenguaje de programación real, pero está diseñado para la lectura humana en lugar de la lectura mediante máquina, y con independencia de cualquier otro lenguaje de programación. Normalmente, el pseudocódigo omite detalles que no son esenciales para la comprensión humana del algoritmo, tales como declaraciones de variables, código específico del sistema y algunas subrutinas.

El lenguaje de programación se complementa, donde sea conveniente, con descripciones detalladas en lenguaje natural, o con notación matemática compacta. Se utiliza pseudocódigo pues este es más fácil de entender para las personas que el código del lenguaje de programación convencional, ya que es una descripción eficiente y con un entorno independiente de los principios fundamentales de un algoritmo. Se utiliza comúnmente en los libros de texto y publicaciones científicas que se documentan varios algoritmos, y también en la planificación del desarrollo de programas informáticos, para esbozar la estructura del programa antes de realizar la efectiva codificación.

No existe una sintaxis estándar para el pseudocódigo, aunque los IDE's que manejan pseudocódigo tengan su sintaxis propia. Aunque sea parecido, el pseudocódigo no debe confundirse con los programas esqueleto que incluyen código ficticio, que pueden ser compilados sin errores. Los diagramas de flujo y UML pueden ser considerados como una alternativa gráfica al pseudocódigo, aunque sean más amplios en papel.

Aplicaciones:

Generalmente se utiliza pseudocódigo en los libros de texto y publicaciones científicas relacionadas con la informática y la computación numérica, para la descripción de algoritmos, de manera que todos los programadores puedan entenderlo, aunque no todos conozcan el mismo lenguaje de programación. Generalmente, en los libros de texto se adjunta una explicación que acompaña



a la introducción y que explica las convenciones particulares en uso. El nivel de detalle del pseudocódigo puede, en algunos casos, acercarse a la de formalizar los idiomas de propósito general.

Un programador que tiene que aplicar un algoritmo específico, sobre todo uno des familiarizado, generalmente comienza con una descripción en pseudocódigo, y luego "traduce" esa descripción en el lenguaje de programación meta y lo modifica para que interactúe correctamente con el resto del programa. Los programadores también pueden iniciar un proyecto describiendo la forma del código en pseudocódigo en el papel antes de escribirlo en su lenguaje de programación, como ocurre en la estructuración de un enfoque de Top-down y Bottom-up arriba hacia abajo.

Sintaxis:

En la actualidad y por lo general, el pseudocódigo, como su nombre lo indica, no obedece a las reglas de sintaxis de ningún idioma en particular ni es de forma estándar sistemática, a pesar de que cualquier escritor en particular vaya a pedir prestado las estructuras de control general, la sintaxis y el estilo, por ejemplo, de algún lenguaje de programación convencional. Pero en caso de que se quiera ejecutar, se debe llevar a forma tipo, para que no genere mensajes de error. Las fuentes populares incluyen la sintaxis de Pascal, BASIC, C, C++, Java, Lisp, y ALGOL. Por lo general, se omiten las declaraciones de variables. A veces, las llamadas a funciones, los bloques de código y el código contenido dentro de un loop se rempazan por una sentencia de una línea en lenguaje natural.

Dependiendo del escritor, el pseudocódigo puede variar mucho en su estilo, yendo desde en un extremo, una imitación casi exacta de un lenguaje de programación real, hasta al acercarse a una descripción en prosa de formato de pseudocódigo en el otro extremo.

Este es un ejemplo de pseudocódigo (para el juego matemático bizz buzz):



Pseudocódigo estilo Fortran:

```
programa bizzbuzz
hacer i = 1 hasta 100
    establecer print_number a verdadero
    si i es divisible por 3
        escribir "Bizz"
        establecer print_number a falso
    si i es divisible por 5
        escribir "Buzz"
        establecer print_number a falso
    si print_number, escribir i
    escribir una nueva línea
fin del hacer
```

Pseudocódigo estilo Pascal:

```
procedimiento bizzbuzz
para i := 1 hasta 100 hacer
    establecer print_number a verdadero;
    Si i es divisible por 3 entonces
        escribir "Bizz";
        establecer print_number a falso;
    Si i es divisible por 5 entonces
        escribir "Buzz";
        establecer print_number a falso;
    Si print_number, escribir i;
    escribir una nueva línea;
fin
```

Definición de datos en pseudocódigo

La definición de datos se da por supuesta, sobre todo en las variables sencillas, si se emplea formaciones: pilas, colas, vectores o registros, se pueden definir en la cabecera del algoritmo, y naturalmente cuando empleemos el pseudocódigo para definir estructuras de datos, esta parte la desarrollaremos adecuadamente.



Ejemplos de Pseudocódigo

Realizar el pseudocódigo de un programa que permita calcular el área de un rectángulo. Se debe introducir la base y la altura para poder realizar el cálculo.

```
Programa; área  
  
Entorno: BASE, ALTURA, AREA son número enteros  
  
Algoritmo:  
  
    escribir "Introduzca la base y la altura"  
  
    leer BASE, ALTURA  
  
    calcular AREA = BASE * ALTURA  
  
    escribir "El área del rectángulo es "AREA  
  
Finprograma
```

Realizar el pseudocódigo que permita al usuario introducir por teclado dos notas, calculando la suma y el producto de las notas.

```
Programa: SumaProducto  
  
Entorno: NOTA1,NOTA2,SUMA,PRODUCTO son números enteros  
  
Algoritmo:  
  
    escribir "Introduzca las notas"  
  
    leer NOTA1,NOTA2  
  
    calcular SUMA = NOTA1 + NOTA2  
  
    calcular PRODUCTO = NOTA1 * NOTA2  
  
    escribir "La suma de las dos notas es:" SUMA  
  
    escribir "El producto de las dos notas es : "PRODUCTO  
  
Finprograma
```

Realizar el pseudocódigo de un programa que permita saber si un número es mayor, menor o igual a cero.

```
Programa: ComparaNúmeros
    Entorno: NUMERO es un número entero
Algoritmo:
Escribir "Introduzca un número "
leer NUMERO
SI NUMERO>0 ENTONCES
    escribir "El número introducido es positivo"
SI NO
    SI NUMERO<0 ENTONCES
        escribir "El número introducido es negativo"
    SI NO
        escribir "El número es cero"
    FINSI
FINSI
Finprograma
```