

Tipos de datos

Las estructuras de datos en el lenguaje C son más simples que las que ofrece Java porque no existe el concepto de "clase" ni de "objeto". C ofrece tipos de datos básicos y dos construcciones para crear datos más complejos.

Tipos de datos básicos

1. Números enteros definidos con la palabra clave "int".
2. Letras o caracteres definidos con palabra clave "char".
3. Números reales definidos con las palabras "float" o "double".

Arreglos

Los arreglos (arrays) permiten almacenar vectores y matrices. Los arreglos unidimensionales sirven para manejar vectores y los arreglos bidimensionales para matrices. Sin embargo, los matrices también se pueden almacenar mediante arreglos unidimensionales y por medio de apuntadores a cantidades se identifican.

El siguiente ejemplo muestra la definición de tres arreglos, uno de 80 elementos doble precisión, otro de 30 elementos enteros y uno de 20 elementos tipo carácter.

double x[80]; int factores[30]; char codSexo[20];

Los nombres deben cumplir con las normas para los identificadores. La primera línea indica que se han reservado 80 posiciones para números doble precisión. Estas posiciones son contiguas. Es importante recordar que en C, a diferencia de otros lenguajes, el primer elemento es x[0], el segundo x[1], el tercero es x[2], y así sucesivamente hasta llegar al último elemento x[79].

En X hay espacio reservado para 80 elementos, pero esto no obliga a trabajar con los 80; el programa puede utilizar menos de 80 elementos.

C no controla si los subíndices están fuera del rango previsto; esto es responsabilidad del programador.

Cuando un parámetro de una función es un arreglo, se considera implícitamente que es un parámetro por referencia. O sea, si en la función se modifica algún elemento del arreglo, entonces se modificará realmente el valor original y no una copia. Pasar un arreglo como parámetro de una función y llamar esta función es muy sencillo.

Tamaño de memoria (arrays)

Un array (arreglo, unidimensional, también denominado vector) es una variable estructurada de un número "n" de variables simples del mismo tipo que son denominadas los componentes o elementos del array. El número de componentes "n" es, entonces, la dimensión o el tamaño del array. De igual manera que en matemáticas, decimos de "A" es un vector de dimensión "n".



n componentes

Formato para declarar un array unidimensional

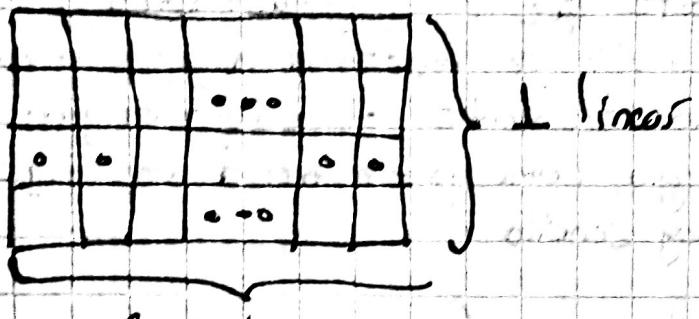
tipo de nombre [n];

Ejemplo int A[4]; define un array de tipo entero de dimensiones 4.

Arrays de dos dimensiones

Un array en C puede tener una, dos o más dimensiones. Por ejemplo, un array de dos dimensiones también denominada matriz, es interpretado como un array (unidimensional) de dimensiones "f" (número de filas), donde cada componente es un array (unidimensional) de dimensiones "C" (número de columnas). Un array de dos dimensiones, contiene, pues, "f" "C" componentes.

A:



Formato para declarar un array multidimensional

`int nombre [f][c];` donde $f, c \dots \geq 1$

Ejemplo

```
int A[3][4] = { (0,1,2,3),
                 (1,2,3,4),
                 (2,4,5,6) };
```

Especificadores
de conversión de printf()

%A → float	%d → int
%A → float	%i → signed int
%C → char	%o → unsigned octal int
%d → int	%p → pointer
%e → float	%s → cadena de chars
%F → float	%x → unsigned hex int
%f → float decimal	%% → %

%g → exponente menor a -4
%i → signed int
%o → unsigned octal int
%p → pointer
%s → cadena de chars
%x → unsigned hex int
%% → %

31-oct-2012

Especificaciones de formato en scanf

- c → entrada de un carácter
- d → entrada de número entero
- f → entrada de número real
- s → entrada de una cadena

Referencias

- * professor.fisica.unam.mx/cintia/cm25.pdf
- * Carboque.com "Curso de Lenguaje C"
- * wiki.cabal.mx "Curso de programación en C"