APRENDERAPROGRAMAR.COM

TYPEDEF C. DECLARACIÓN DE TIPOS DEFINIDOS POR EL USUARIO. EJEMPLO CON ARRAYS O ARREGLOS (CU00513F)

Sección: Cursos
Categoría: Curso básico de programación en lenguaje C desde cero

Fecha revisión: 2031

Resumen: Entrega nº13 del curso básico "Programación C desde cero".

Autor: Mario Rodríguez Rancel

DECLARACIÓN DE TIPOS CON TYPEDEF EN C

C permite que el programador defina sus propios tipos de datos mediante la palabra clave typedef. Esta palabra puede tener distintos usos. Vamos a ver ahora uno de ellos.

La sintaxis a emplear requiere cumplir con dos pasos.

El primero, declarar un tipo de datos que es el tipo del array y que nos permitirá crear cuantas variables queramos de ese tipo, es decir, cuantos arrays queramos conteniendo un mismo tipo de dato y un mismo número de elementos en el array. La sintaxis será:

```
typedef tipoDeElementosDelArray nombreDelTipoArray [numeroElementos];
```

Ejemplo: *typedef int TipoVectorEnteros [4];* declara que se crea un nuevo tipo de datos, definido por nosotros, denominado TipoVectorEnteros, que contendrá 4 valores de tipo *int*.

El segundo paso para hacer uso del tipo definido implica que hemos de crear una variable del tipo definido por nosotros en la expresión anterior. Al igual que para declarar un entero escribimos *int nombreVariable;*, para declarar un array del tipo creado por nosotros escribiremos *nombreDelTipoDefinido nombreVariable;*

Para el ejemplo que hemos puesto, escribiríamos lo siguiente: *TipoVectorEnteros vector1;* donde vector1 es el nombre que le damos al vector que hemos creado, que contendrá cuatro valores enteros (obviamente estos valores podemos modificarlos).

Ejemplos de declaración de tipos arrays serían:

- typedef int TipoVectorVez [9];
- typedef char VectorAmigo [1000];
- typedef double DecimalNum [24];
- typedef int VectorInt [23];
- typedef int TipoVectorLong[8];
- typedef char TipoPalabra[255];

Tras declarar el tipo tendríamos que declarar variables de ese tipo, por ejemplo: *TipoVectorVez vez;*, *VectorAmigo amigo;*, *DecimalNum numDecimal;*, *VectorInt numeroDeCoches;* ó *TipoVectorLong jugador;*. Podemos definir múltiples arrays del mismo tipo, por ejemplo:

VectorInt numeroDeCoches;

VectorInt numeroDePersonas;

VectorInt numeroDeRuedas;

VectorInt pruebasRealizadas;

¿Cómo elegir los nombres de los tipos y los nombres de las variables? No existen reglas precisas al respecto, pero te recomendamos para los nombres de tipos usar siempre un prefijo que comience con mayúsculas que podría ser Tipo y para las variables un nombre que comience por minúsculas. Es un convenio que siguen muchos programadores, aunque no es obligatorio. Los nombres de tipos y variables deben ser lo más descriptivos posibles para hacer el programa fácil de leer y de entender. Piensa que es válido tanto declarar *typedef int TipoVectorInt;* como *typedef int TVI;* Sin embargo es más correcto usar TipoVectorInt que TVI porque resulta más descriptivo de la función y cometido del tipo VectorInt que tres letras cuyo significado es poco entendible.

Crea un proyecto y escribe el siguiente código:

```
#include <stdio.h>
#include <stdlib.h>
// Ejemplo aprenderaprogramar.com
int main() {
typedef int TipoVectorEnteros [4];
TipoVectorEnteros numeroDeCoches;

numeroDeCoches[0] = 32;
numeroDeCoches[1]=0; numeroDeCoches[2]=0; numeroDeCoches[3]=0;
printf ("El numero de coches en la hora cero fue %d \n", numeroDeCoches[0]);
printf ("El numero de coches en la hora uno fue %d \n", numeroDeCoches[1]);
printf ("El numero de coches en la hora dos fue %d \n", numeroDeCoches[2]);
printf ("El numero de coches en la hora tres fue %d \n", numeroDeCoches[3]);
return 0;
}
```

El resultado de ejecución será el mismo que vimos anteriormente.

La definición de tipos con *typedef* es una posibilidad que brinda el lenguaje C pero que no está disponible en todos los lenguajes.

En estos momentos es normal que puedas confundir nombres de tipos con nombres de variables. Ten en cuenta que se trata de cosas distintas: un tipo es “un molde” con el que podemos crear tantas variables de ese tipo como deseemos. A medida que practiques con estos conceptos te resultará más fácil trabajar con ellos.

EJERCICIO

Crea el código de un programa que cumpla las siguientes premisas. Declara un tipo definido por el usuario que se llame TipoVectorDe10 y que contenga 10 elementos de tipo entero. Declara una variable de nombre edadesConcursantes que sea de tipo TipoVectorDe10. Establece las siguientes edades para cada uno de los elementos del array (arreglo): 55, 24, 34, 26, 46, 62, 45, 24, 31, 61. Muestra por pantalla, haciendo uso de los elementos del array, mensajes informando de la edad de las personas. Por ejemplo: La edad de la primera persona es 55 annos. La edad de la segunda persona es 24 annos. La edad de la tercera persona es... etc. (No usamos la eñe).

Para comprobar si tus respuestas son correctas puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU00514F

Acceso al curso completo en aprenderaprogramar.com -- > Cursos, o en la dirección siguiente:
http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=82&Itemid=210