



APRENDERAPROGRAMAR.COM

## INSTRUCCIÓN GOTO EN C. ETIQUETAS O LABELS. POR QUÉ NO DEBE USARSE (HARMFUL). EJEMPLOS (CU00546F)

Sección: Cursos

Categoría: Curso básico de programación en lenguaje C desde cero

Fecha revisión: 2031

**Resumen:** Entrega nº46 del curso básico "Programación C desde cero".

Autor: Mario Rodríguez Rancel

### INSTRUCCIÓN GOTO (IRA) EN C. ETIQUETAS O LABELS.

El uso de la instrucción `goto` no debe contemplarse en la práctica habitual de la programación, de hecho se recomienda prescindir de ella porque se ha demostrado que empeora la calidad del código y dificulta su depuración. Recomendamos leer la descripción de esta instrucción en el curso "Bases de la programación nivel I" de [aprenderaprogramar.com](http://aprenderaprogramar.com). La sintaxis a emplear en C es:



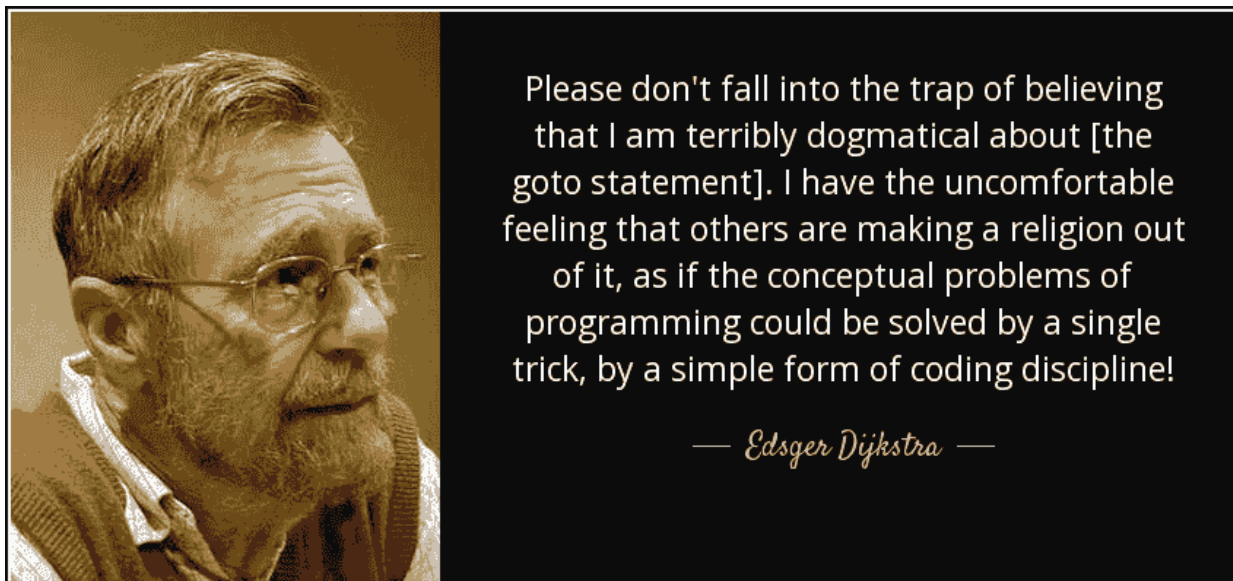
```
goto etiqueta;
```

Donde `etiqueta` es un punto de referencia introducido en el código de programación indicado escribiendo simplemente `etiqueta: .` Para ejecutar un `goto` no es necesario que todas las líneas o secciones de código estén identificadas, pero sí es necesario que la ubicación de destino esté etiquetada con el identificador que se elija.

`goto` es una sentencia en general mal vista, ya que normalmente hace los programas más difíciles de entender y depurar, hasta el punto de que en muchos cursos de programación ni siquiera se explica. Nosotros expondremos un ejemplo de su uso simplemente para tener constancia de su existencia, pero recomendamos que no se use en el código de programas.



...entonces escribí GOTO 500 ¡Y aquí estoy!



### EJERCICIO RESUELTO Nº1: ENUNCIADO

Transformar en código el siguiente pseudocódigo y razonar para comprender su lógica. Se trata de un ejemplo de aplicación de *IrA*, que debemos implementar en C mediante la sentencia *goto*.

Nota: considerar que *Numerodatos* es un valor fijo de 35, y que los datos se encuentran en el fichero *notas.dat*, que habremos creado previamente.

```

1. Inicio [Ejemplo de IrA goto aprenderaprogramar.com]
2. Leer Numerodatos [Establece el nº de datos a extraer]
3. [El dato esperado en fichero debe valer entre cero y diez]
4. Desde i = 1 hasta Numerodatos Hacer
    Leer Dato(i)
    Si Dato(i) < 0 ó Dato(i) > 10 Entonces [Dato no válido]
        IrA 99 [Envío a Gestión de Errores]
    FinSi
    Suma = Suma + Dato(i)
    Siguiente
5. Media = Suma / Numerodatos
6. Mostrar "La media es", Media
7. Finalizar
99. [Gestión de Errores]
    99.1 Mostrar "No se pudo completar la extracción de datos. Motivo:
        dato no válido. Revise archivo y vuelva a intentarlo"
    99.2 Mostrar "Gracias por utilizar este programa"
100. Fin
    
```

## EJERCICIO RESUELTO N°1: SOLUCIÓN

```

#include <stdio.h>
#include <stdlib.h>
#define NumeroAlumnos 35

int main() {
    double nota[NumeroAlumnos]; //35 elementos numerados de 0 a 34
    double sumatorio = 0;
    double media = 0;
    int i=0;
    puts ("Programa Calcular Notas Medias \n");
    //Extracción de datos de archivo
    FILE* fichero;
    fichero = fopen("notas.dat", "rt");
    for (i=0; i<=NumeroAlumnos-1; i++) {
        fscanf (fichero, "%lf", &nota[i] );
        if (nota[i] < 0 || nota[i] > 10) { //[Dato no válido]
            goto gestionDeErrores; //Salto a gestión de errores
        }
        sumatorio = sumatorio + nota[i];
    }
    fclose(fichero);
    media = sumatorio / NumeroAlumnos;
    printf ("La nota media de la clase es %lf\n", media);
    exit(-1); // Ejemplos curso C aprenderaprogramar.com

gestionDeErrores:
    printf("No se pudo completar la extraccion de datos. Motivo:\n");
    printf("dato no valido. Revise archivo y vuelva a intentarlo.\n");
    printf("Gracias por utilizar este programa.\n");
    fclose(fichero);
    return 0;
}

```

Describimos el código propuesto. En primer lugar se utiliza la directiva *#define* para establecer como valor constante asociado a *NumeroAlumnos* el valor 35. Se declara la variable tipo array *double nota*, que constará de 35 elementos (numerados de 0 a 34). También se declaran e inicializan las variables de tipo *double* *sumatorio* y *media*, así como la variable entera *i*. Se muestra el mensaje "Programa Calcular Notas Medias" y se abre el fichero *notas.dat* para lectura de datos (modo *rt*). Se van extrayendo los datos existentes en el fichero y almacenándolos en el array *nota*. En caso de detectarse una nota menor que cero o mayor que 10 se considera que existe un dato no válido y se introduce un salto en el flujo del programa dirigiéndolo a la etiqueta <<gestionDeErrores>>. Si se accede a esta parte del programa, se muestra un mensaje de error, se cierra el fichero y termina el programa.

En caso de que la lectura de datos sea exitosa se cierra el fichero, se calcula la media de las notas de los alumnos y se muestra la nota media por pantalla antes de finalizar el programa.

**EJERCICIO N°1**

Estudia el siguiente código y responde a las preguntas:

```
#include <stdio.h>
#include <stdlib.h>
// Curso de programación C aprenderaprogramar.com
/* C program to demonstrate the working of goto statement. */

int main(){
    float num,average,sum;
    int i,n;
    printf("Maximum no. of inputs: ");
    scanf("%d",&n);
    for(i=1;i<=n;++i){
        printf("Enter n%d: ",i);
        scanf("%f",&num);
        if(num<0.0)
            goto jump;        /* control of the program with goto */
        sum=sum+num;
    }
    jump:
    average=sum/(i-1);
    printf("Average: %.2f",average);
    return 0;
}
```

- a) Sin ejecutar el código (sólo pensando) responde: ¿para qué sirve este programa? ¿cuál será el resultado del programa si introducimos los números 10, 5 y 2? ¿cuál será el resultado del programa si introducimos los números 10, -5 y 2?
- b) Ejecutando el código ¿cuál será el resultado del programa si introducimos los números 10, 5 y 2? ¿cuál será el resultado del programa si introducimos los números 10, -5 y 2?
- c) ¿Qué significado tiene ++i en el contexto de este programa?
- d) ¿Es equivalente escribir `if( num<0.0) goto jump;` y `if( num < 0.0) { goto jump; }` ?
- e) Haz que el código ofrezca los mismos resultados evitando el uso de goto.

Para comprobar si tus respuestas son correctas puedes consultar en los foros [aprenderaprogramar.com](http://aprenderaprogramar.com).

## EJERCICIO N°2

Estudia el siguiente código:

```
#include <stdio.h>
#include <stdlib.h>
// Curso de programación C aprenderaprogramar.com
/* C program to demonstrate the working of goto statement. */

int main() {
    int age;
    Vote:
    printf("you are eligible for voting");

    NoVote:
    printf("you are not eligible to vote");

    printf("Enter you age:");
    scanf("%d", &age);
    if(age>=18)
        goto Vote;
    else
        goto NoVote;

    return 0;
}
```

- a) Sin ejecutar el código (sólo pensando) responde: ¿para qué sirve este programa? ¿cuál será el resultado del programa si introducimos el número 22?
- b) Crea otro programa equivalente (que produzca los mismos resultados) sin hacer uso de la sentencia goto.

Para comprobar si tus respuestas son correctas puedes consultar en los foros [aprenderaprogramar.com](http://www.aprenderaprogramar.com).

**Próxima entrega:** CU00547F

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