

Práctica No. 2

Estructuras de control.

Objetivo: El alumno se familiarizará con el uso de las estructuras de control selectivas e iterativas para el control del flujo de ejecución de un conjunto de instrucciones, implementando:

- Operadores
- Estructuras de selección
- Estructuras de Iteración

Material:

- Computadora Personal (PC)
- Programa Editor de texto (ASCII), compilador GCC

Equipo:

- Computadora Personal

Introducción

Los enunciados ejecutables en C, llevan a cabo ya sea acciones (como son cálculos, entradas o salidas de datos), o toman decisiones. Las condiciones en las estructuras **if** se forman utilizando los operadores de igualdad y los operadores relacionales, los operadores relacionales tienen un mismo nivel de precedencia y se asocian de izquierda a derecha. Los operadores de igualdad tienen un nivel de precedencia menor que los operadores relacionales y también se asocian de izquierda a derecha.

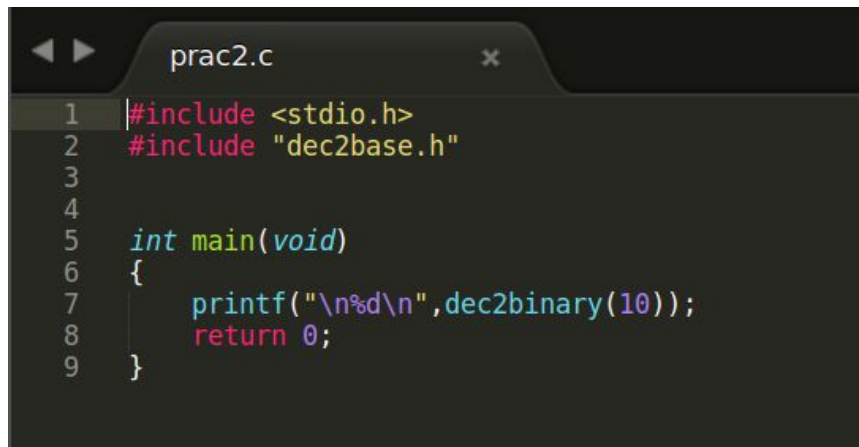
Teoría:

- Tablas de operadores:
 - Aritméticos
 - Relacionales
 - Lógicos
 - Incremento y decremento
 - Bit a bit
 - Jerarquía de operadores

Desarrollo:

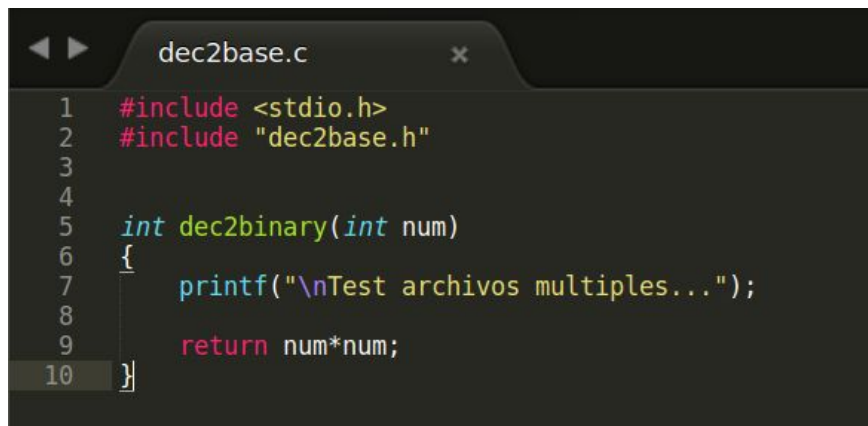
- 1) Para comenzar escriba los siguientes programas usando un editor de texto (Notepad++, Sublime Text, Notepad, etc), guardarlos con el nombre **prac2.c**,

dec2base.c, dec2base.h de preferencia en sus Documentos.



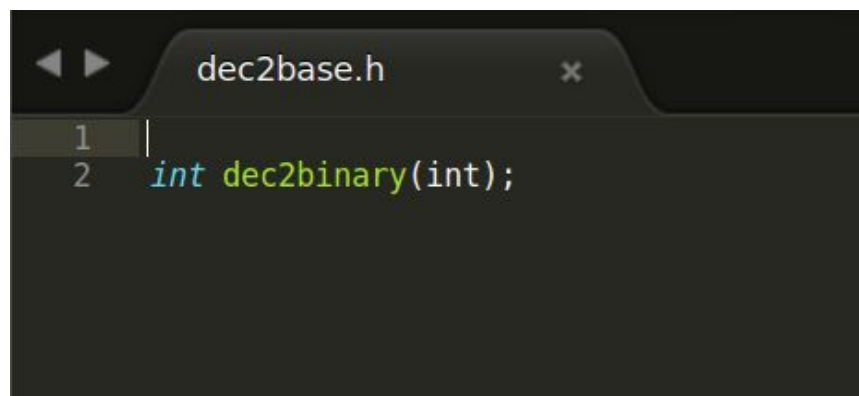
```
1 #include <stdio.h>
2 #include "dec2base.h"
3
4
5 int main(void)
6 {
7     printf("\n%d\n", dec2binary(10));
8     return 0;
9 }
```

Figura 1. Programa principal en C



```
1 #include <stdio.h>
2 #include "dec2base.h"
3
4
5 int dec2binary(int num)
6 {
7     printf("\nTest archivos multiples...");
8
9     return num*num;
10 }
```

Figura 2. Programa en C



```
1
2 int dec2binary(int);
```

Figura 3. Archivo de cabeceras en C

- 2) Para compilar el programa con el compilador GCC abra la línea de comandos de windows mediante la combinación de teclas Windows+R y escriba cmd, presione enter, ya en consola deberá teclear la siguiente instrucción, considerando que es una ruta válida:

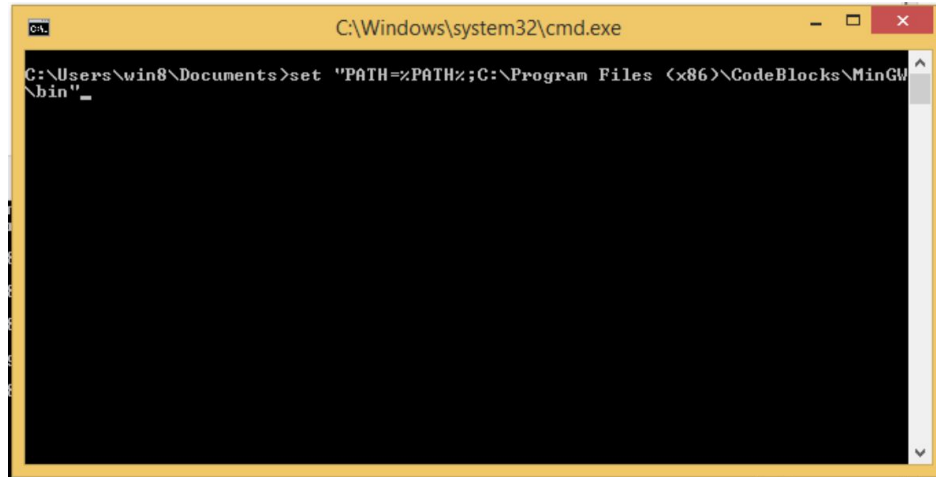


Figura 2. Consola de Windows

set "PATH=%PATH%;C:\Program Files (x86)\CodeBlocks\MinGW\bin"

- 3) Compile el programa `prac2.c` y `dec2base.c` mediante el comando:

```
C:\Users\win8\Documents>gcc -Wall -c dec2base.c prac2.c
```

Esto generará los archivos **prac2.o** y **dec2base.o**

- 4) Para crear un ejecutable desde los archivos **o** debe hacerlo mediante el comando:

```
C:\Users\win8\Documents>gcc prac2.o dec2base.o -o prac2.exe
```

Esto generará el archivo **prac2.exe**

Desarrollo de la práctica

Desarrollar las 4 funciones contenidas en el archivo `dec2base.c`:

1. `dec2binchar`
2. `dec2bin`
3. `dec2hex`
4. `dec2oct`

Las 4 funciones reciben como parámetro un número entero decimal, y regresan valores de la representación de éste valor en otra base según la función:

1. `dec2binchar(int numero, char* binchar, int size)` **regresa una**

cadena de caracteres que representa el número en base binaria. Por ejemplo, si recibe el número 6 deberá regresar una cadena de caracteres con los siguientes valores ['1', '1', '0']. Hacer uso de paso de parámetro por referencia a través del apuntador binchar.

2. `dec2bin(int numero)` regresa un valor entero que representa el número en base binaria. Por ejemplo, si recibe el número 6 deberá regresar el valor 110.
3. `dec2hex(int numero, char* hexchar, int size)` regresa una cadena de caracteres que representa el número en base 16 (hexadecimal). Por ejemplo, si recibe el número 500 deberá regresar una cadena de caracteres con los siguientes valores ['1', 'F', '4']. Hacer uso de paso de parámetro por referencia a través del apuntador hexchar.
4. `dec2oct(int numero)` regresa un valor entero que representa el número en base 8 (octal). Por ejemplo, si recibe el número 10 deberá regresar el valor 12.

Conclusiones y Comentarios.