	Carátula para entrega de prácticas	
Facultad de Ingeniería		Laboratorios de docencia

Laboratorio de Computación

Salas A y B

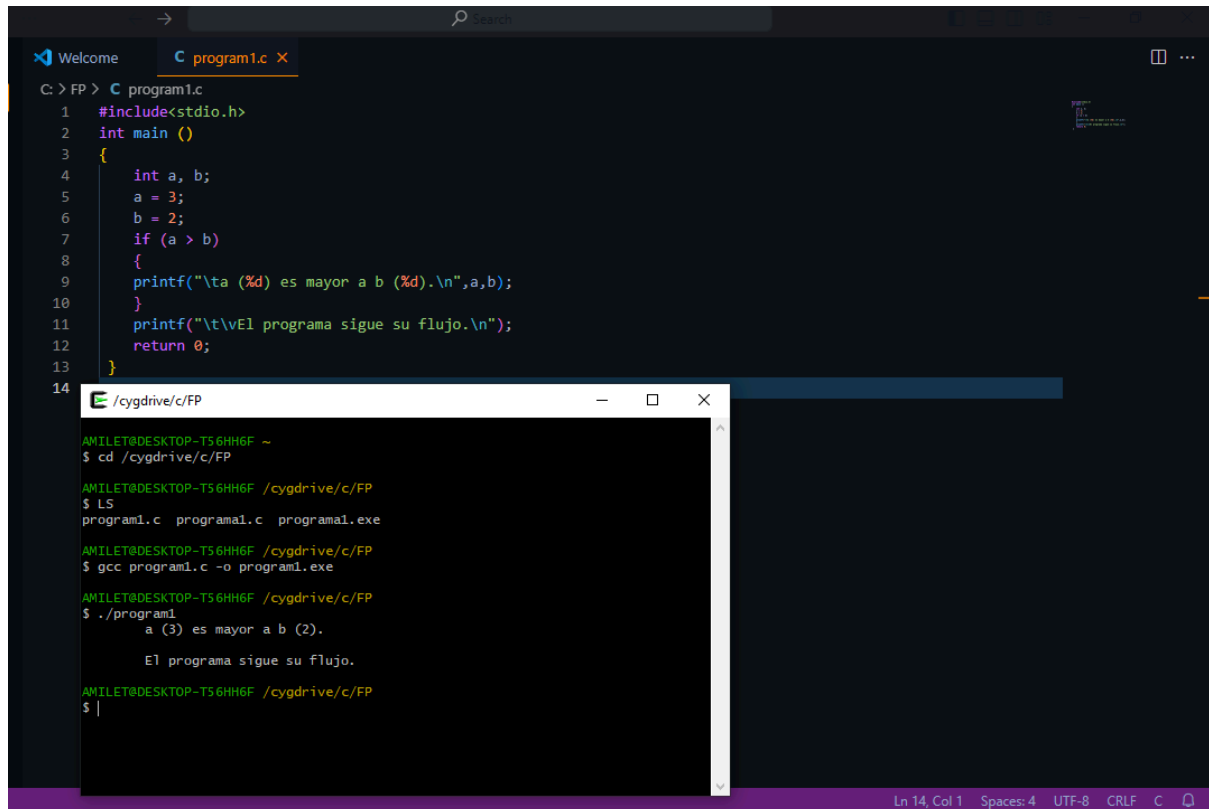
Profesor(a):	César Fabián Domínguez Velasco
Asignatura:	Fundamentos de Programación
Grupo:	15
No de Práctica(s):	7
Integrante(s):	Camacho Duarte Héctor Enrique
	Gutiérrez Esquivel Giovani Emiliano
	Flores Jiménez Diego
	Moreno Chapan Amilet
	Reyes García Raúl de Jesús
No. de lista	07,11,15,16,25,35.
o brigada:	
Semestre:	2024-02
Fecha de entrega:	20/03/24
Observaciones:	
CALIFICACIÓN:	

Objetivo:

El alumno elaborará programas en lenguaje C que incluyan las estructuras de selección

if, if-else, switch y ternaria (o condicional) para la resolución de problemas básicos.

Programa1.c



The image shows a code editor window with a file named 'C program1.c' open. The code is as follows:

```
1 #include<stdio.h>
2 int main ()
3 {
4     int a, b;
5     a = 3;
6     b = 2;
7     if (a > b)
8     {
9         printf("\ta (%d) es mayor a b (%d).\n",a,b);
10    }
11    printf("\t\tEl programa sigue su flujo.\n");
12    return 0;
13 }
```

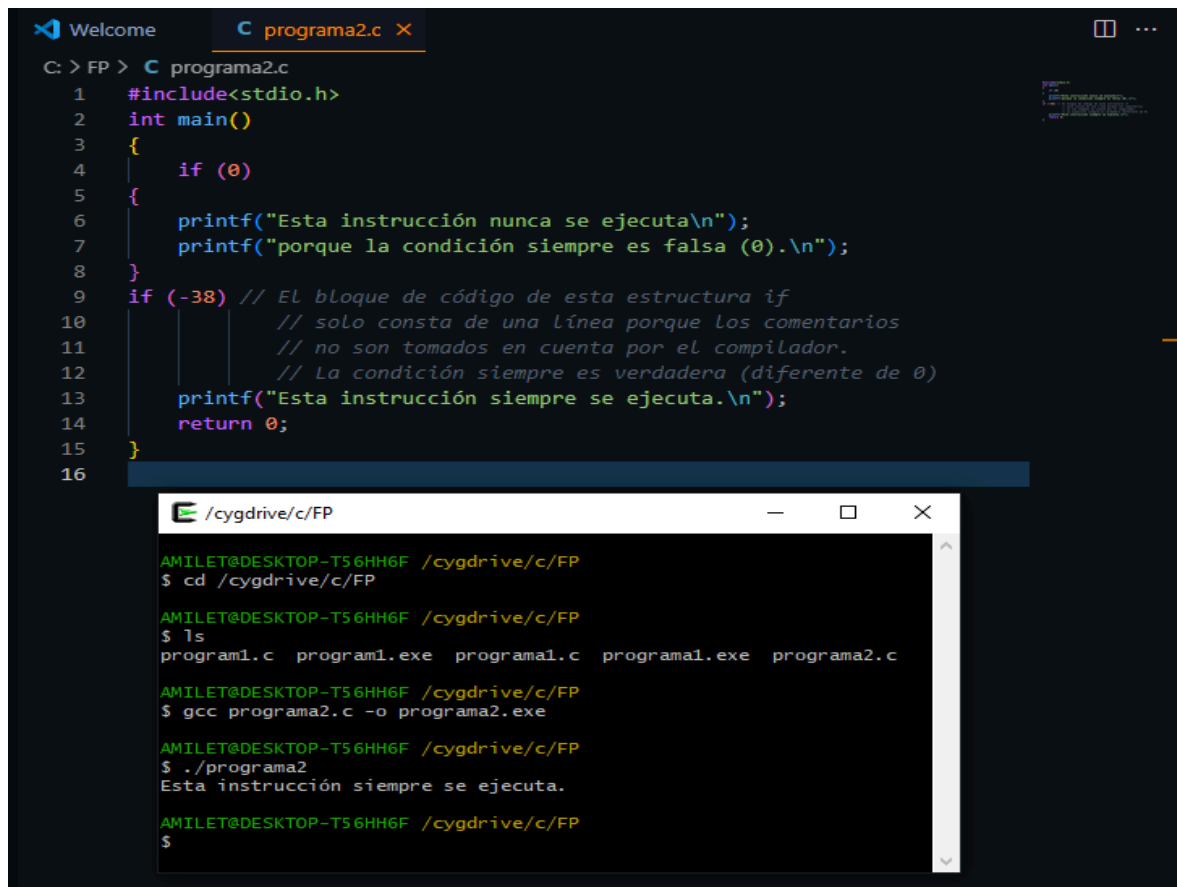
Below the code editor is a terminal window titled '/cygdrive/c/FP'. It shows the following commands and output:

```
AMILET@DESKTOP-T56HH6F ~
$ cd /cygdrive/c/FP
AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ ls
program1.c  program1.c  program1.exe
AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ gcc program1.c -o program1.exe
AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ ./program1
    a (3) es mayor a b (2).

    El programa sigue su flujo.
AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$
```

The status bar at the bottom of the code editor indicates 'Ln 14, Col 1', 'Spaces: 4', 'UTF-8', 'CRLF', and 'C'.

Programa2.c



The image shows a code editor window with a file named `programa2.c` open. The code is a C program that demonstrates a false condition in an `if` statement. The code is as follows:

```
1  #include<stdio.h>
2  int main()
3  {
4      if (0)
5      {
6          printf("Esta instrucción nunca se ejecuta\n");
7          printf("porque la condición siempre es falsa (0).\n");
8      }
9      if (-38) // El bloque de código de esta estructura if
10         { // solo consta de una línea porque los comentarios
11             // no son tomados en cuenta por el compilador.
12             // La condición siempre es verdadera (diferente de 0)
13             printf("Esta instrucción siempre se ejecuta.\n");
14             return 0;
15         }
16
```

Below the code editor, a terminal window is open, showing the execution of the program. The terminal output is as follows:

```
AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ cd /cygdrive/c/FP

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ ls
program1.c  program1.exe  programa1.c  programa1.exe  programa2.c

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ gcc programa2.c -o programa2.exe

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ ./programa2
Esta instrucción siempre se ejecuta.

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$
```

Programa3.c

```

Welcome  C programa3.c X
C: > FP > C programa3.c
1  #include<stdio.h>
2  int main()
3  {
4      int num;
5      printf("Ingrese un número:\n");
6      scanf("%d",&num);
7      if ( num%2 == 0 )
8          printf("El número %d es par.\n",num);
9      else
10         printf("El número %d es impar.\n",num);
11     return 0;
12 }
13

/cygdrive/c/FP
AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ cd /cygdrive/c/FP

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ ls
programa1.c  programa1.c  programa2.c  programa3.c
programa1.exe programa1.exe programa2.exe programa3.exe

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ gcc programa3.c -o programa3.exe

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ ./programa3
Ingrese un número:
2
El número 2 es par.

AMILET@DESKTOP-T56HH6F /cygdrive/c/FP
$ |
```

Programa4.c

```

/cygdrive/c/Fundamentos

Raúl@LAPTOP-5JRCPAOK /cygdrive/c/Fundamentos
$ ls
programa4.c  programa4.exe

Raúl@LAPTOP-5JRCPAOK /cygdrive/c/Fundamentos
$ gcc programa4.c -o programa4

Raúl@LAPTOP-5JRCPAOK /cygdrive/c/Fundamentos
$ ./programa4.exe
Ingrese 3 números separados por espacios:
15 23 50
50 es mayor a 23 que es mayor a 15

Raúl@LAPTOP-5JRCPAOK /cygdrive/c/Fundamentos
$ ./programa4
Ingrese 3 números separados por espacios:
11 34 90
90 es mayor a 34 que es mayor a 11
```

C programa4.c ×

C programa4.c > main()

```
1  #include <stdio.h>
2  int main()
3  {
4      int uno, dos, tres;
5      printf ("Ingrese 3 números separados por espacios:\n");
6      scanf ("%d %d %d", &uno, &dos, &tres);
7      if (uno > dos)
8      {
9          if (dos > tres)
10         {
11             printf("%d es mayor a %d que es mayor a %d\n", uno, dos, tres);
12         }
13         else
14         {
15             if (uno > tres)
16             {
17                 printf("%d es mayor a %d que es mayor a %d\n", uno, tres, dos);
18             }
19             else
20             {
21                 printf("%d es mayor a %d que es mayor a %d\n", tres, uno, dos);
22             }
23         }
24     }
25 }
26 else
27 {
28     if (dos > tres)
29     {
30         if (tres > uno)
31         {
32             printf("%d es mayor a %d que es mayor a %d\n", dos, tres, uno);
33         }
34         else
35         {
36             printf("%d es mayor a %d que es mayor a %d\n", dos, uno, tres);
37         }
38     }
39     else
40     {
41         printf("%d es mayor a %d que es mayor a %d\n", tres, dos, uno);
42     }
43 }
44 return 0;
45 }
```

Programa5.c

```
C programa5.c
C programa5.c > main()
1  #include <stdio.h>
2  int main()
3  {
4      char op = '\0';
5      printf("\tMenú\n");
6      printf("Elegir la opción deseada\n");
7      printf("a) Ingresar\n");
8      printf("b) Registrarse\n");
9      printf("c) Salir\n");
10     scanf("%c",&op);
11     switch(op)
12     {
13         default:
14             printf("Opción no válida.\n");
15             break;
16         case 'a':
17             printf("Se seleccionó 'Ingresar'.\n");
18             break;
19         case 'b':
20             printf("Se seleccionó 'Registrarse'.\n");
21             break;
22         case 'c':
23             printf("Se seleccionó 'Salir'.\n");
24             break;
25     }
26     return 0;
27 }
28 }
```

```
/cygdrive/c/Fundamentos
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ ls
programa4.c programa4.exe programa5.c
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ gcc programa5.c -o programa5
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ ls
programa4.c programa4.exe programa5.c programa5.exe
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ ./programa5.exe
Menú
Elegir la opción deseada
a) Ingresar
b) Registrarse
c) Salir
a
Se seleccionó 'Ingresar'.
```

Programa6.c

```
C programa6.c X
C programa6.c > main()
1  #include <stdio.h>
2  int main()
3  {
4      int op = 0;
5      printf("\tMenú\n");
6      printf("Elegir la opción deseada\n");
7      printf("1) Ingresar\n");
8      printf("2) Registrarse\n");
9      printf("3) Salir\n");
10     scanf("%d",&op);
11     switch(op)
12     {
13         case 1:
14             printf("Se seleccionó 'Ingresar'.\n");
15             break;
16         case 2:
17             printf("Se seleccionó 'Registrarse'.\n");
18             break;
19         case 3:
20             printf("Se seleccionó 'Salir'.\n");
21             break;
22         default:
23             printf("Opción no válida.\n");
24     }
25     return 0;
26 }
27 }
```

```
/cygdrive/c/Fundamentos
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ ls
programa4.c programa4.exe programa5.c programa5.exe programa6.c
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ gcc programa6.c -o programa6
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ ls
programa4.c programa5.c programa6.c
programa4.exe programa5.exe programa6.exe
Raúl@LAPTOP-5JRCPA0K /cygdrive/c/Fundamentos
$ ./programa6.exe
Menú
Elegir la opción deseada
1) Ingresar
2) Registrarse
3) Salir
1
Se seleccionó 'Ingresar'
```

Programa7.c

```
C programa7.c > main()
1  #include <stdio.h>
2  int main()
3  {
4      // declaración de la enumeración
5      enum boolean {NO, YES};
6
7      // declaración de una variable tipo enumeración
8      enum boolean valorBooleano;
9      valorBooleano = YES;
10
11     // Se comprueba que el valor de una enumeración es entero
12     printf("%d\n", valorBooleano);
13
14     // Se comprueba que el valor de una enumeración se puede reasignar
15     enum diasSemana {LUNES, MARTES, MIERCOLES=5, JUEVES, VIERNES};
16     printf("\n%d", LUNES);
17     printf("\n%i", MARTES);
18     printf("\n%d", MIERCOLES);
19     printf("\n%i", JUEVES);
20     printf("\n%d\n", VIERNES);
21     return 0;
22 }
```

```
bpsom@Hope /cygdrive/c/FP
```

```
$ ls
```

```
Cadena.c      Programa5.exe  Programachar.ex  ola.exe
Cadena.exe    Programa6.c   Programachar.exe programa7.c
Programa5.c   Programa6.exe  ola.c            programa7.exe
```

```
bpsom@Hope /cygdrive/c/FP
```

```
$ gcc programa7.c -o programa7.exe
```

```
bpsom@Hope /cygdrive/c/FP
```

```
$ ./programa7.exe
```

```
1
```

```
0
```

```
1
```

```
5
```

```
6
```

```
7
```

programa 8.c

```
#include <stdio.h>
int main()
{
    // Los valores de una enumeración son enteros y c
    enum diasSemana {LUNES, MARTES, MIERCOLES, JUEVES
    int op;
    printf("Ingrese el día de la semana.\n");
    printf("1) Lunes\n");
    printf("2) Martes\n");
    printf("3) Miércoles\n");
    printf("4) Jueves\n");
    printf("5) Viernes\n");
    printf("6) Sábado\n");
    printf("7) Domingo\n");
    scanf("%d", &op);
    switch(op-1)
    {
        case LUNES:
        case MARTES:
            printf("Inicio de semana.\n");
            break;
        case MIERCOLES:
            printf("Mitad de semana.\n");
            break;
        case JUEVES:
            printf("¡Casi inicia el fin de semana!\n");
            break;
        case VIERNES:
        case SABADO:
            printf("¡Fin de semana!\n");
            break;
        case DOMINGO:
            printf("Día de descanso.\n");
            break;
        // No se necesita default
    }
    return 0; // Valor entero en hexadecimal
```



```

bpsom@Hope /cygdrive/c/FP
$ gcc programa8.c -o programa8.exe

bpsom@Hope /cygdrive/c/FP
$ ./programa8.exe
Ingrese el día de la semana.
1) Lunes
2) Martes
3) Miércoles
4) Jueves
5) Viernes
6) Sábado
7) Domingo
5
¡Fin de semana!

```

programa9.c

```

C programa9.c > main()
1  #include <stdio.h>
2  int main()
3  {
4      double a, b, res;
5      printf("Calcular el error matemático E = |a - b|\n\n");
6      printf("Ingrese el valor de a:\n");
7      scanf("%lf",&a);
8      printf("Ingrese el valor de b:\n");
9      scanf("%lf",&b);
10     res = a < b ? b-a : a-b;
11     printf("El error matemático de\n");
12     printf("| %lf - %lf | es %lf\n", a, b, res);
13     return 0;
14 }

```

```

bpsom@Hope /cygdrive/c/FP
$ ./programa9.exe
Calcular el error matemático E = |a - b|

Ingrese el valor de a:
96
Ingrese el valor de b:
94
El error matemático de
| 96.000000 - 94.000000 | es 2.000000

```

Problema propuesto 1.c

Escribe un programa en C, el cual verifique si un número es divisible por 7 y 11 en la misma sentencia condicional. Pide al usuario que ingrese un número, guarda esta entrada en una variable entera, luego realiza el paso de comparación. Deberás imprimir "El número es divisible entre 7 y 11" en caso de que la condición sea verdadera e imprimir "El número no es divisible entre 7 y 11" en el caso de que la condición sea falsa. Nota: El operador módulo (%) y ampersand (&) en la condición pueden ayudarte a resolver el problema.

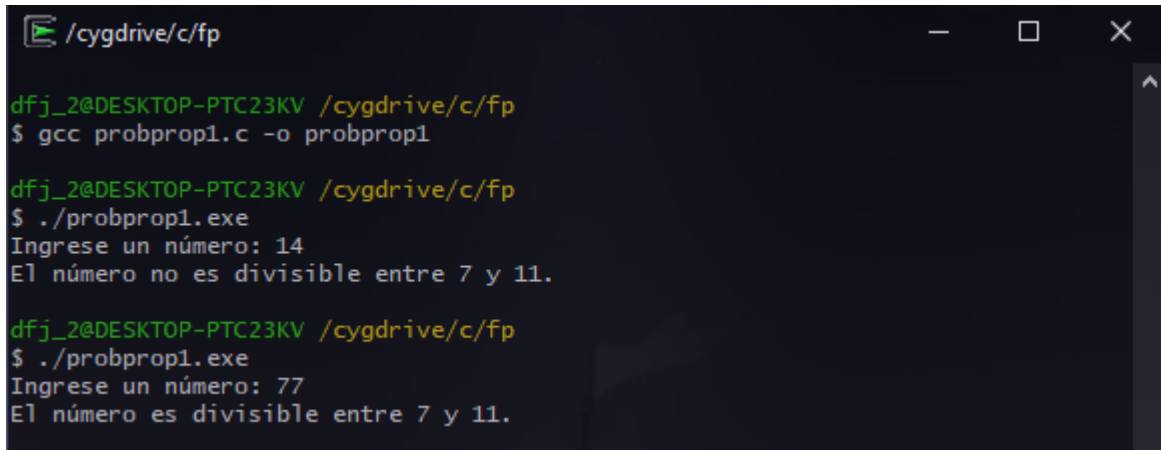
```
#include <stdio.h>

int main() {
    int numero;

    // Pedir al usuario que ingrese un número
    printf("Ingrese un número: ");
    scanf("%d", &numero);

    // Verificar si el número es divisible por 7 y 11
    if (numero % 7 == 0 && numero % 11 == 0) {
        printf("El número es divisible entre 7 y 11.\n");
    } else {
        printf("El número no es divisible entre 7 y 11.\n");
    }

    return 0;
}
```



```
/cygdrive/c/fp
dfj_2@DESKTOP-PTC23KV /cygdrive/c/fp
$ gcc probprop1.c -o probprop1

dfj_2@DESKTOP-PTC23KV /cygdrive/c/fp
$ ./probprop1.exe
Ingrese un número: 14
El número no es divisible entre 7 y 11.

dfj_2@DESKTOP-PTC23KV /cygdrive/c/fp
$ ./probprop1.exe
Ingrese un número: 77
El número es divisible entre 7 y 11.
```

Problema propuesto 2.c

Escribe un programa en C, el cual imprima en la pantalla si tras leer un carácter desde la consola, este es una vocal o consonante. Utiliza la estructura de selección "switch" para resolver este problema. Como bien se describe, debes leer de la consola un carácter y guardarlo en una variable tipo char y posteriormente evaluar esta variable en una estructura switch. En el caso por defecto (default) debes imprimir en la consola "Consonante." Si en caso es a, e, i, o, u; debes imprimir en la consola "Vocal".

```

#include <stdio.h>
#include <string.h>

char* cadena = "s";
int contador = 0;
int i;

int main(int argc, char const *argv[])
{
    for (i = 0; i < strlen(cadena); i++)
    {
        switch (cadena[i])
        {
            case 'a':
            case 'e':
            case 'i':
            case 'o':
            case 'u':
            case 'A':
            case 'E':
            case 'I':
            case 'O':
            case 'U':
                contador++;
                printf("Vocal\n");
                break;
            default:
                printf("Consonante\n");
                break;
        }
    }
    return 0;
}

```

```

opsom@Hope /cygdrive/c/FP
$ gcc propuesto2.c -o propuesto2.exe

opsom@Hope /cygdrive/c/FP
$ ./propuesto2.exe
Consonante

opsom@Hope /cygdrive/c/FP
$

```

```

#include <stdio.h>
#include <string.h>

char* cadena = "a";

```

```
int contador = 0;
int i;

int main(int argc, char const *argv[])
{
    for (i = 0; i < strlen(cadena); i++)
    {
        switch (cadena[i])
        {
            case 'a':
            case 'e':
            case 'i':
            case 'o':
            case 'u':
            case 'A':
            case 'E':
            case 'I':
            case 'O':
            case 'U':
                contador++;
                printf("Vocal\n");
                break;
            default:
                printf("Consonante\n");
                break;
        }
    }
    return 0;
}
```

```
bpsom@Hope /cygdrive/c/FP
$ gcc propuesto2.c -o propuesto2.exe

bpsom@Hope /cygdrive/c/FP
$ ./propuesto2.exe
Vocal

bpsom@Hope /cygdrive/c/FP
$
```

Problema propuesto 3.

Escribe un programa en C, el cual imprima "Año bisiesto" o "Año común" según la entrada a través de la consola. Debes pedir al usuario el año a evaluar y guardar este valor en un entero. Antes de evaluar si un año dado es bisiesto debes validar que el año ingresado sea mayor que 0, si esto es falso el programa termina. Nota: Si la variable "year" es exactamente divisible por 4 y al mismo tiempo no es divisible por 100 o si año es divisible por 400, entonces "año" es considerado bisiesto.

```
C AñoBisiestoAñoComún.c > ...
1  #include <stdio.h>
2  int main(int argc, char const *argv[])
3  {
4      int year;
5
6      printf("Ingresa un año para determinar si es Año bisiesto o Año común: ");
7      scanf("%d", &year);
8
9      if(year <= 0)
10     {
11         printf("El año ingresado no es válido.\n");
12         return 0;
13     }
14     if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0 )
15     {
16         printf("Año bisiesto.\n");
17     }
18     else
19     {
20         printf("Año común.\n");
21     }
22     return 0;
23 }
24
```

Prueba del problema propuesto 3

```
ineff@Inefable /cygdrive/c/Fundamentos
$ gcc AñoBisiestoAñoComún.c -o AñoBisiestoAñoComún.exe

ineff@Inefable /cygdrive/c/Fundamentos
$ ./AñoBisiestoAñoComún
Ingresa un año para determinar si es Año bisiesto o Año común: 0
El año ingresado no es válido.

ineff@Inefable /cygdrive/c/Fundamentos
$ ./AñoBisiestoAñoComún
Ingresa un año para determinar si es Año bisiesto o Año común: 2024
Año bisiesto.

ineff@Inefable /cygdrive/c/Fundamentos
$ ./AñoBisiestoAñoComún
Ingresa un año para determinar si es Año bisiesto o Año común: 2023
Año común.

ineff@Inefable /cygdrive/c/Fundamentos
$ ./AñoBisiestoAñoComún
Ingresa un año para determinar si es Año bisiesto o Año común: 2020
Año bisiesto.

ineff@Inefable /cygdrive/c/Fundamentos
$ ./AñoBisiestoAñoComún
Ingresa un año para determinar si es Año bisiesto o Año común: 2028
Año bisiesto.

ineff@Inefable /cygdrive/c/Fundamentos
$ |
```

Conclusión

Hemos desarrollado una comprensión más sólida de cómo usar las estructuras de selección en C para resolver una variedad de problemas. A través de la práctica con las instrucciones if, if-else, switch y el operador ternario, hemos aprendido a tomar decisiones dentro de nuestros programas de una manera más eficiente y efectiva.

Se concluye que los objetivos de la práctica se cumplieron ya que se pudo ejecutar programas en donde se utilizaron distintas sentencias tanto los de la práctica como los propuestos.

Bibliografía

Laboratorio Salas A y B. (n.d.). <http://lcp02.fi-b.unam.mx/>

El lenguaje de programación C. Brian W. Kernighan, Dennis M. Ritchie, segunda edición, USA, Pearson Educación 1991.