

Universidad Autónoma de Baja California

Facultad de Ingeniería.



Práctica 2: Tipos
operadores y expresiones

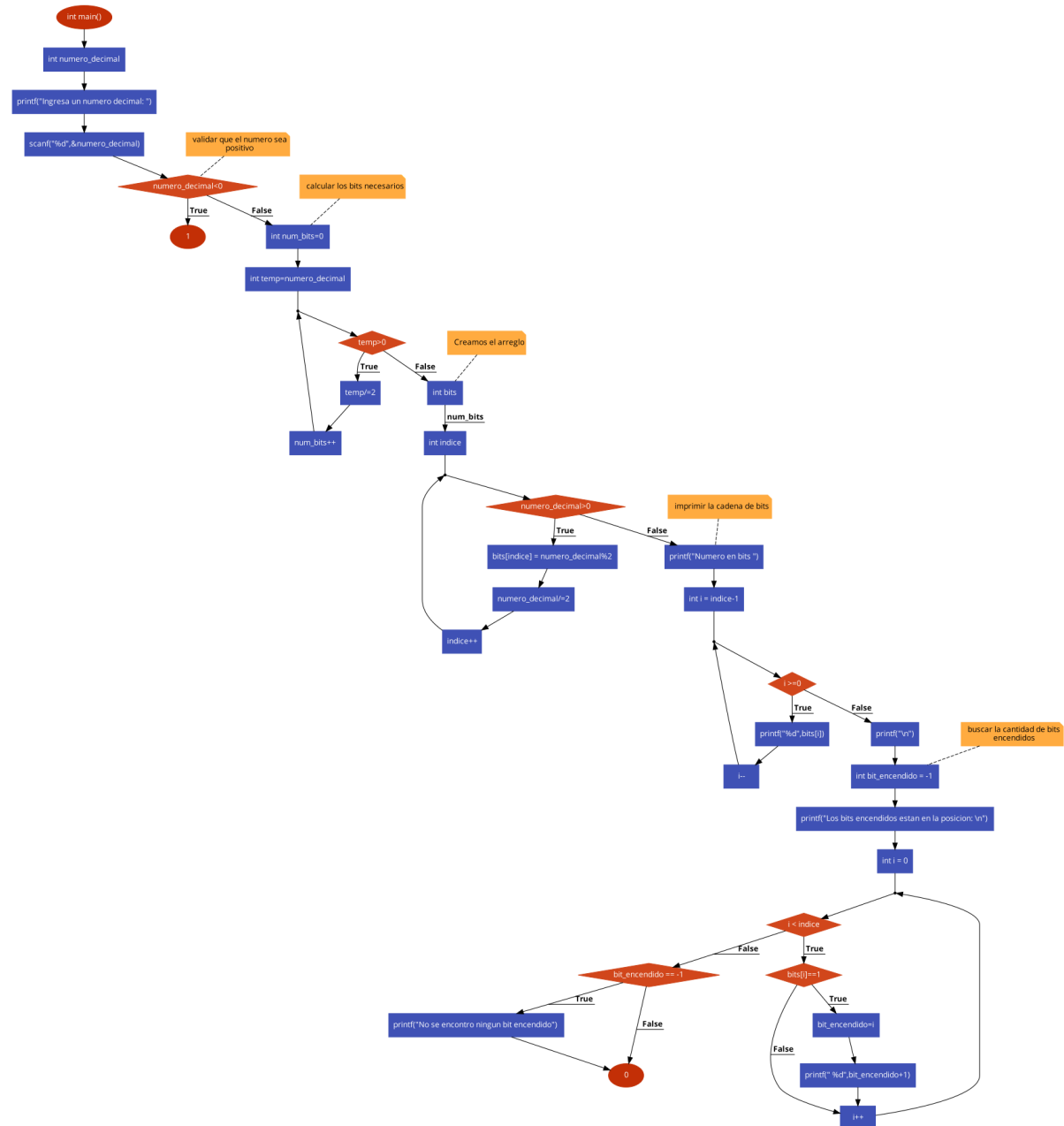
Mtra.: Yurith Altamirano

Alumno: Diego Quiros

Matricula: 372688

Grupo: 932

Ejercicio 1. Crear un programa que permita a los usuarios ingresar un número entero, especificar el número de bits que se deben considerar y luego analizar el número en términos de bits encendidos, posiciones y representación binaria.



```
1  #include <stdio.h>
2  #include <limits.h>
3
4  //Practica2_BINARIO.cpp
5  //Quiros_Vicencio_Diego_Demian_372688
6  //09/07/2023
7  //Convierte numeros decimales a numeros binarios
8  //Practica2_TiposOperadoresExpresiones_Quiros_Diego.cpp
9
10 int main(){
11
12     int numero_decimal;
13     printf("Ingresa un numero decimal: ");scanf("%d",&numero_decimal);
14
15     //validar que el numero sea positivo
16
17     if (numero_decimal<0)
18     {
19         return 1;
20     }
21
22     //calcular los bits necesarios
23
24     int num_bits=0;
25     int temp=numero_decimal;
26
27     while (temp>0)
28     {
29         temp/=2;
30         num_bits++;
31     }
32
33     //Creamos el arreglo
34
35     int bits[num_bits];
36     int indice;
37
```

```

38     while (numero_decimal>0)
39     {
40         bits[indice] = numero_decimal%2;
41         numero_decimal/=2;
42         indice++;
43     }
44
45     //imprimir la cadena de bits
46
47     printf("Numero en bits ");
48
49     for (int i = indice-1 ; i >=0 ; i--)
50     {
51         printf("%d",bits[i]);
52     }
53     printf("\n");
54
55     //buscar la cantidad de bits encendidos
56
57     int bit_encendido = -1;
58
59     printf("Los bits encendidos estan en la posicion: \n");
60
61     for (int i = 0; i < indice; i++)
62     {
63         if (bits[i]==1)
64         {
65             bit_encendido=i;
66             printf(" %d",bit_encendido+1);
67         }
68     }
69
70     if (bit_encendido == -1)
71     {
72         printf("No se encontro ningun bit encendido");
73     }
74

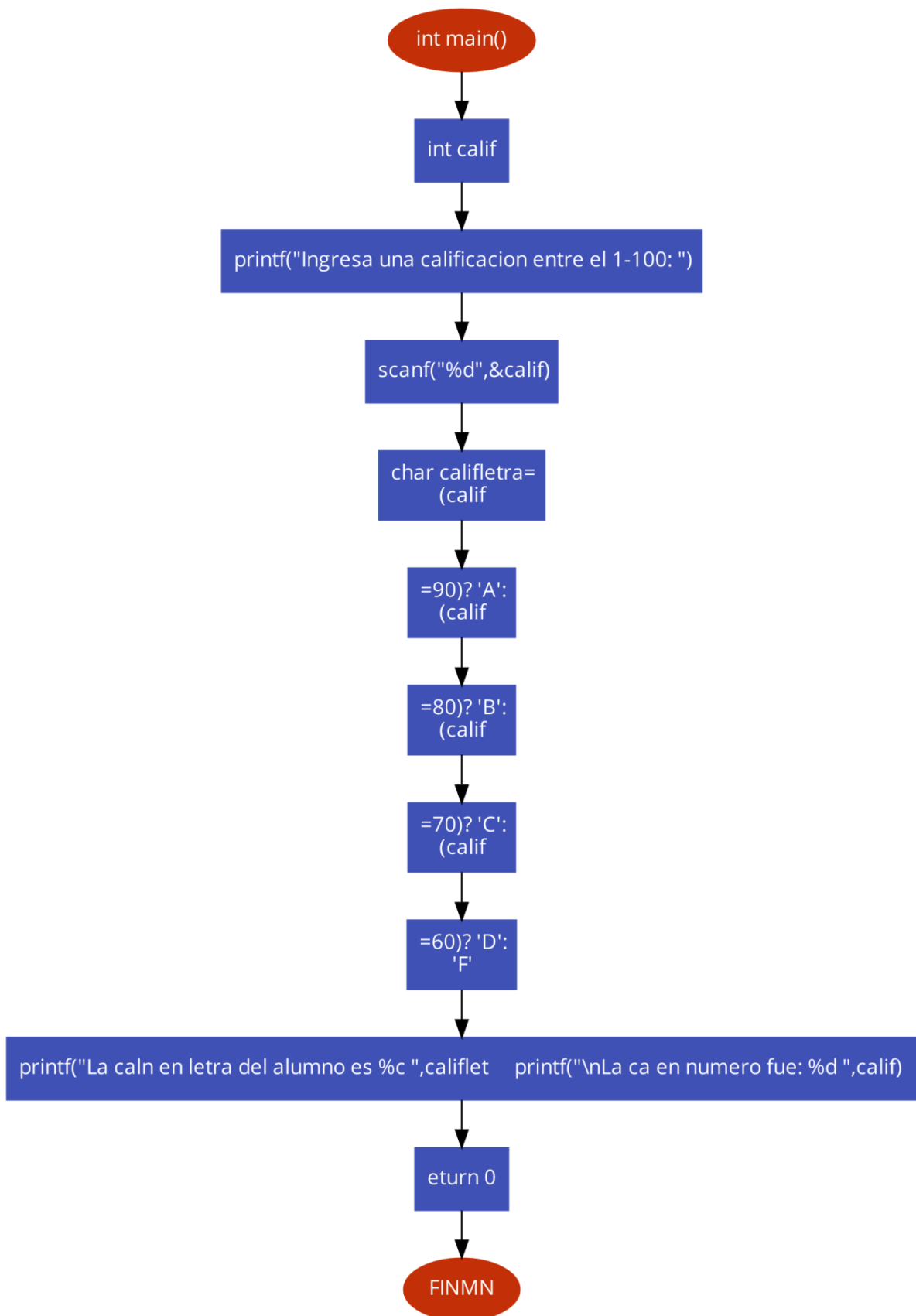
```

```

74
75
76     return 0;
77 }

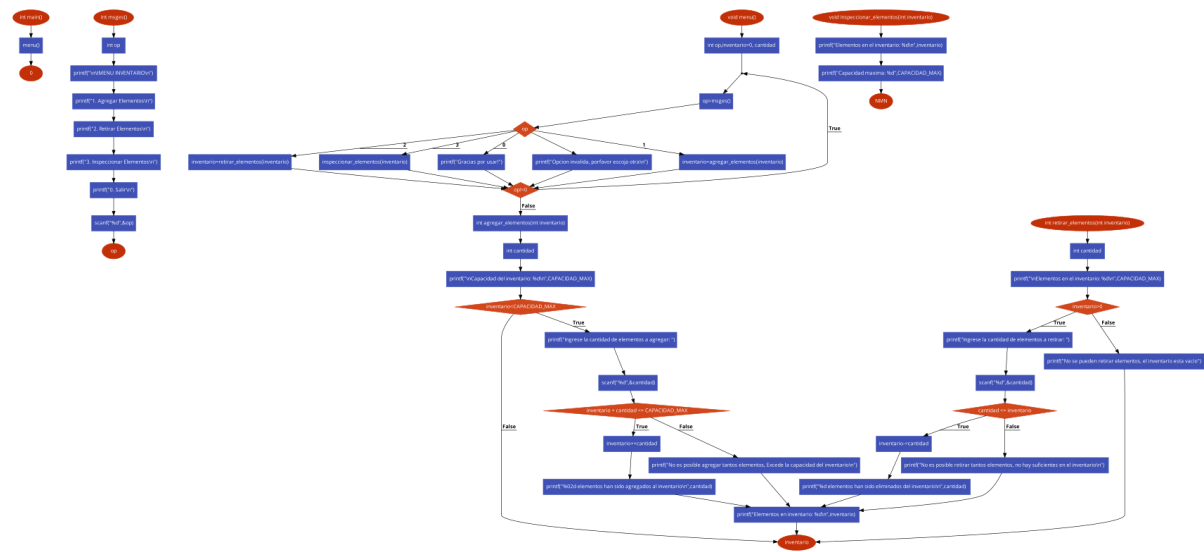
```

Ejercicio 2: Crear un programa que tome una calificación numérica y la convierta en una calificación en letra utilizando operadores ternarios.



```
1  #include <stdio.h>
2
3  //Practica2_CALIF.cpp
4  //Quiros_Vicencio_Diego_Demian_372688
5  //09/07/2023
6  /*Convierte una calificacion ingresada en decimal a una calificacion en
7  caracter utilizando operadores ternarios*/
8  //Practica2_TiposOperadoresExpresiones_Quiros_Diego.cpp
9
10 int main(){
11
12     int calif;
13
14
15     printf("Ingresa una calificacion entre el 1-100: ");scanf("%d",&calif);
16     char califletra=
17         (calif>=90)? 'A':
18         (calif>=80)? 'B':
19         (calif>=70)? 'C':
20         (calif>=60)? 'D':
21         'F'
22     ;
23
24     printf("La calificacion en letra del alumno es %c ",califletra);
25     printf("\nLa calificacion en numero fue: %d ",calif);
26
27     return 0;
28 }
```

Ejercicio 3: El objetivo de este ejercicio es simular un sistema básico de gestión de inventario en C, donde los usuarios pueden agregar y retirar elementos del inventario.



```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  #define CAPACIDAD_MAX 100
5
6  //Practica2_INVENTARIO.cpp
7  //Quiros_Vicencio_Diego_Demian_372688
8  //09/08/2023
9  //Inventario donde se puedan ingresar, retirar e ispeccionar elementos
10 //Practica2_TiposOperadoresExpresiones_Quiros_Diego.cpp
11
12
13 int msges(void);
14 void menu(void);
15 int agregar_elementos(int inventario);
16 int retirar_elementos(int inventario);
17 void inspeccionar_elementos(int inventario);
18
19 int main()
20 {
21     menu();
22
23     return 0;
24 }
25
26 int msges()
27 {
28     int op;
29     printf("\n\tMENU INVENTARIO\n");
30     printf("1. Agregar Elementos\n");
31     printf("2. Retirar Elementos\n");
32     printf("3. Inspeccionar Elementos\n");
33     printf("0. Salir\n");
34     scanf("%d",&op);
35     return op;
36 }
37
```



```

38 void menu()
39 {
40     int op,inventario=0, cantidad;
41
42     do
43     {
44         op=msges();
45
46         switch (op)
47         {
48             case 1:
49                 inventario=agregar_elementos(inventario);
50                 break;
51             case 2:
52                 inventario=retirar_elementos(inventario);
53                 break;
54             case 3:
55                 inspeccionar_elementos(inventario);
56                 break;
57             case 0:
58                 printf("Gracias por usar!");
59                 break;
60             default:
61                 printf("Opcion invalida, porfavor escoja otra\n");
62                 break;
63         }
64     } while (op!=0);
65 }
66
67 int agregar_elementos(int inventario)
68 {
69     int cantidad;
70     printf("\nCapacidad del inventario: %d\n",CAPACIDAD_MAX);
71     if (inventario<CAPACIDAD_MAX)
72     {
73         printf("Ingrese la cantidad de elementos a agregar: ");scanf("%d",&cantidad);
74         if (inventario + cantidad <= CAPACIDAD_MAX)

```

```

75     {
76         inventario+=cantidad;
77         printf("%02d elementos han sido agregados al inventario\n",cantidad);
78         printf("Elementos en inventario: %d\n",inventario);
79     }
80     else
81     {
82         printf("No es posible agregar tantos elementos, Excede la capacidad del inventario\n");
83         printf("Elementos en inventario: %d\n",inventario);
84     }
85 }
86 }
87 return inventario;
88 }
89
90 int retirar_elementos(int inventario)
91 {
92     int cantidad;
93     printf("\nElementos en el inventario: %d\n",CAPACIDAD_MAX);
94     if (inventario>0)
95     {
96         printf("Ingrese la cantidad de elementos a retirar: ");scanf("%d",&cantidad);
97         if (cantidad <= inventario)
98         {
99             inventario-=cantidad;
100             printf("%d elementos han sido eliminados del inventario\n",cantidad);
101             printf("Elementos en inventario: %d\n",inventario);
102         }
103         else
104         {
105             printf("No es posible retirar tantos elementos, no hay suficientes en el inventario\n");
106             printf("Elementos en inventario: %d\n",inventario);
107         }
108     }
109     else
110     {
111         printf("No se pueden retirar elementos. el inventario esta vacio"):

```

```

112     }
113     return inventario;
114 }
115
116 void inspeccionar_elementos(int inventario)
117 {
118     printf("Elementos en el inventario: %d\n",inventario);
119     printf("Capacidad maxima: %d",CAPACIDAD_MAX);
120 }

```