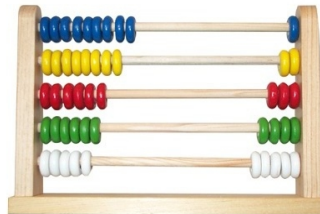




República Bolivariana de Venezuela
Aldea Universitaria Liceo Fray Pedro de Agreda

Prof. Elías Cisneros.




Lenguaje C++
Contadores y Acumuladores



Usted es libre de:

- * Copiar, distribuir y comunicar públicamente la obra .
- * Hacer obras derivadas .

Bajo las condiciones siguientes:

	Reconocimiento. Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).
	No comercial. No puede utilizar esta obra para fines comerciales.
	Compartir bajo la misma licencia. Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.



Contador:

Un contador es una variable cuyo valor se incrementa o decrementa en una cantidad constante cada vez que se produce un determinado suceso o acción. Los contadores se utilizan con la finalidad de contar sucesos o acciones internas de un bucle; deben realizar una operación de inicialización y posteriormente las sucesivas de incremento o decremento del mismo. La inicialización consiste en asignarle al contador un valor inicial. Se situará antes y fuera del bucle.

Ejemplo de contadores

- Cada vez que alguien es atendido en la taquilla de un banco se puede almacenar en un contador de personas atendidas.
- Cada vez que se vende una entrada a un juego de béisbol se puede almacenar en un contador de entradas vendidas.
- Cada vez que se registra un nuevo usuario en una página web se puede almacenar en un contador de usuarios registrados.

A continuación se presentan en pseudocódigo la estructura general de un contador:

```
int cont = 0
inicio del bucle
i = i + 1
fin del bucle
```

Los contadores pueden ser de incremento o de decremento, por lo tanto la siguiente expresión es válida

- $i = i + 1$; o $i++$; (Expresión equivalente).
- $i = i - 1$; o $i--$; (Expresión equivalente)

Un ejemplo sencillo de uso de contadores es el siguiente:

```
1. using namespace std;
2. #include <iostream>
3. int main()
4. {
5.     int cnt=0;
6.     int var=1;
7.     while(var!=0)
8.     {
9.         cout<<"Ingrese valor var: "<<endl;
10.        cin>>var;
11.        cout<<"Valor de contador: "<<cnt<<endl;
12.        cnt=cnt+1;
13.    }
14.    cout<<"** Fin **"<<endl;
15. return 0;
16. }
```



Acumulador o totalizador :

Es una variable que suma sobre sí misma un conjunto de valores para de esta manera tener la suma de todos ellos en una sola variable. La diferencia entre un contador y un acumulador es que mientras el primero va aumentando de uno en uno, el acumulador va aumentando en una cantidad variable.

Ejemplos de acumuladores

- La cantidad de dinero depositado en una cuenta de ahorros en el lapso de un mes se pudiera almacenar en un acumulador de depósitos.
- La cantidad de estudiantes de todos los salones de clases pudiera almacenarse en un acumulador de cantidad de estudiantes de cada sección.
- La suma de los habitantes cada uno de los estados del país se pudiera guardar en el acumulador llamado total habitantes del país.

A continuación se presentan en pseudocódigo la estructura general de un acumulador:

```
int acum = 0
inicio del bucle
acum = acum + valor
fin del bucle
```

Donde valor puede ser un dato introducido por el usuario, el producto de un cálculo, etc. A continuación un ejemplo de funcionamiento de un acumulador:

```
1. using namespace std;
2. #include <iostream>
3. int main()
4. {
5.     int ac=0;
6.     int var;
7.     for(int i=0;i<3;i++)
8.     {
9.         cout<<"Ingrese el valor de la variable :";
10.        cin>>var;
11.        ac=ac +var;
12.        cout<<"El valor del acumulador es: "<<ac<<endl;
13.    }
14. }
```

Ejercicios resueltos

1. Programa que imprime los números impares entre 0 y 100 e imprime cuantos hay. La variable contador está inicializada a uno por conveniencia. Sugerencia: Hacer este programa con la sentencia do while.



```
1. #include <iostream>
2. using namespace std;
3. int main()
4. {
5.     int contador,contador_impar;
6.     contador_impar=0;
7.     cout << "\t Estos son los números impares entre 0 y 100 \n";
8.     cout << "\t -----\n";
9.     for(contador=1; contador<100; contador= contador+ 2)//Este contador permite moverme de impar a
impar
10. {
11.     contador_impar++;        // Este otro contador cuenta el número de impares
12.     cout << contador << " , ";
13. }
14. cout << "\nEl número de impares entre cero y cien es: " << contador_impar << endl;
15. cout << "Pulse una tecla para acabar...";
16. cin.get();
17. return 0;
18. }
```

2. Confeccionar un programa que pida al usuario que ingrese 50 temperaturas. Obtener: La cantidad de temperaturas positivas y la cantidad de temperaturas negativas. Edemas obtener la cantidad de ceros (0) ingresados el problema se resuelve, ACUMULANDO en una variable "CantPos" la cantidad de temperaturas positivas y en "CantNeg" la cantidad de variables Negativas, y en "Ceros" la cantidad de ceros.

```
1. using namespace std;
2. #include <iostream>
3. int main()
4. {
5.     int Temp;
6.     int CantPos=0,CantNeg=0,Ceros=0;
7.     //los acumuladores siempre deben ser inicializados---> "SIEMPRE"
8.     // El motivo es que uno acumula "BASURA" + "EL valor" en la PRIMERA VEZ, lo cual nos dara
como resultado cualquier cosa
9.     for(int i;i<=50;i++)
10.    {
11.        cout<<"Ingrese una temperatura";
12.        cin>>Temp;
13.        if( Temp > 0 ){
14.            CantPos++;
15.        }
16.        else{
```



```
17.     if( Temp == 0 ){
18.         Ceros++;
19.     }
20.     else{
21.         CantNeg++;
22.     }
23. }
24. }
25. cout<<"La cantidad de temperaturas positivas es "<<CantPos<<endl;
26. cout<<"La cantidad de temperaturas positivas es "<<CantNeg<<endl;
27. cout<<"La cantidad de temperaturas positivas es "<<Ceros<<endl;
28. return 0;
29. }
```

3. Programa que cuenta y suma los números pares e impares entre 1 y 100. Lo importante de este programa es ver como alterna el valor la variable booleana.

```
1. using namespace std;
2. int main()
3. {
4.     bool bandera;
5.     int impares,sumaimpares,pares,sumapares,contador;
6.     impares=sumaimpares=pares=sumapares=0;
7.     bandera = true;
8.     for(contador=1; contador<=100; contador++)
9.     {
10.         if(bandera == true)
11.         {
12.             impares=impares+1;
13.             sumaimpares = sumaimpares + contador;
14.             bandera = false;
15.         }
16.         else
17.         {
18.             pares=pares+1;
19.             sumapares = sumapares + contador;
20.             bandera = true;
21.         }
22.     }
23.     cout << "La suma de los impares es: " << sumaimpares << endl;
24.     cout << "El número de impares es: " << impares << endl;
25.     cout << "La suma de pares es: " << sumapares << endl;
26.     cout << "El número de pares es: " << pares << endl;
27.     return 0;
```



28. }

4. El programa ejemplo se trata de una calculadora de operaciones básicas (+, -, /, *, %). Se ha utilizado la sentencia 'switch' para definir qué operación se realizará teniendo en cuenta el operador insertado.

```
1. // Calculadora de operaciones simples.
2. #include <iostream>
3. using namespace std;
4. int main()
5. {
6.     int a, b;
7.     char oper;
8.
9.     cout << "Introduzca la operacion a realizar usando espacios:\n";
10.    cout << "Operadores + , - , * , / , %" << endl;
11.    cout << "Ejemplo 2 + 3 o 3 * 5 5 / 5\n\n";
12.    cin >> a >> oper >> b;
13.    switch(oper)
14.    {
15.        case '+': cout << a << oper << b << " = " << a + b << endl;
16.            break;
17.        case '-': cout << a << oper << b << " = " << a - b << endl;
18.            break;
19.        case '*': cout << a << oper << b << " = " << a * b << endl;
20.            break;
21.        case '/': cout << a << oper << b << " = " << a / b << endl;
22.            break;
23.        case '%': cout << a << oper << b << " = " << a % b << endl;
24.            break;
25.
26.        default: break;
27.    }
28.
29.    return 0;
30. }
```

6. Escribir en pantalla los números de cinco en cinco hasta el cien con la sentencia do...while.

```
1. #include <iostream> //Biblioteca para cin y cout
2. using namespace std;
3. int main() //funcion main lleva el control del programa
4. {
5.     int x=0; //declaracion e inicializacion de variable
```



```
6. do
7. {
8.     x=x+5; //hacer aumento de cinco en cinco
9.     cout <<x<<" , "<<endl; //escribir los numeros separados por coma y salto de pagina
10. }
11. while (x<=99); //condicion siempre que sea menor a cien
12. return 0;
13. }
```

7. El ejemplo muestra el uso de la sentencia condicional IF, y el uso del operador lógico &&. A través de estos se calcula el menor número entre tres números enteros. En el caso del lógico &&, observar como su uso "involucra", ó "exige" que sus dos operandos cumplan determinada condición:

Para el caso del primer if

operando 1 es: num_1 < num_2

operando 2 es: num_1 < num_3

&& es: El operador lógico Y ó AND.

```
1. using namespace std;
2. #include <iostream>
3. int main(void)
4. {
5.     /* Declaramos las variables */
6.     int num_1, num_2, num_3;
7.     int num_menor;
8.     /* Inicializamos las variables num_... */
9.     num_1 = 2;
10.    num_2 = 134;
11.    num_3 = 5;
12.    /* Determinamos el menor de los números */
13.    if (num_1 < num_2 && num_1 < num_3)
14.    {
15.        num_menor = num_1;
16.        cout<<"\n De los valores "<< num_1<<" , "<<num_2<<" , "<<num_3<< " menor es: ";
17.        cout<<num_menor<<endl;
18.    }
19.    else if (num_2 < num_1 && num_2 < num_3)
20.    {
21.        num_menor = num_2;
22.        cout<<"\n De los valores "<< num_1<<" , "<<num_2<<" , "<<num_3<< " menor es: ";
23.        cout<<num_menor<<endl;
24.    }
25.    else if (num_3 < num_1 && num_3 < num_2)
```



```
26.  {
27.    num_menor = num_3;
28.    cout<<"\n De los valores "<< num_1<<","<<num_2<<","<<num_3<< " menor es: ";
29.    cout<<num_menor<<endl;
30.  }
31.  getchar();
32.  return 0;
33. }
```

Referencias

<http://www.psicofxp.com/forums/programacion.313/936520-acumulador-contador-baderas-c.html>
<http://gencervel.wordpress.com/2008/01/22/contadores-y-acumuladores/>
<http://www.conclase.net/c/ejemplos.php?tema=5#item11>
<http://www.conclase.net/c/ejemplos.php?tema=5>