



Universidad
Continental

FUNDAMENTOS DE PROGRAMACIÓN





¿Qué aprendimos la sesión anterior?





Programas utilizando estructura secuencial y selectiva

1.- SECUENCIAL

- Realice un programa que calcule la siguiente ecuación:

$$4x^2 + \sqrt{9x}$$

- Dados los catetos de un triángulo rectángulo, calcule su hipotenusa.

2.- SELECTIVA

- Validar el ingreso de datos de los ejercicios anteriores.



ESTRUCTURAS DE CONTROL PARA LA PROGRAMACIÓN

Mientras, Hacer – Mientras, Para

Fundamentos de Programación



Semana 11

ucontinental.edu.pe



Propósito

Identifica la sintaxis y empleo de estructuras repetitivas en el lenguaje de programación C++ (while, do while, for).





Agenda del día

1. Estructuras Repetitivas

- While
- Do – While
- For

2. Ejercicios



ESTRUCTURAS REPETITIVAS





Estructuras repetitivas

- Los programas utilizados hasta este momento han examinado conceptos de programación, tales como entradas, asignaciones, expresiones, operaciones, sentencias secuenciales, de selección y salida. Sin embargo, muchos problemas requieren de características de repetición, en las que algunos cálculos o secuencia de instrucciones se repiten una y otra vez, utilizando diferentes conjuntos de datos.
- Ejemplos:
 - ✓ Verificaciones de datos
 - ✓ Conteos
 - ✓ Acumulación de totales



Estructuras repetitivas

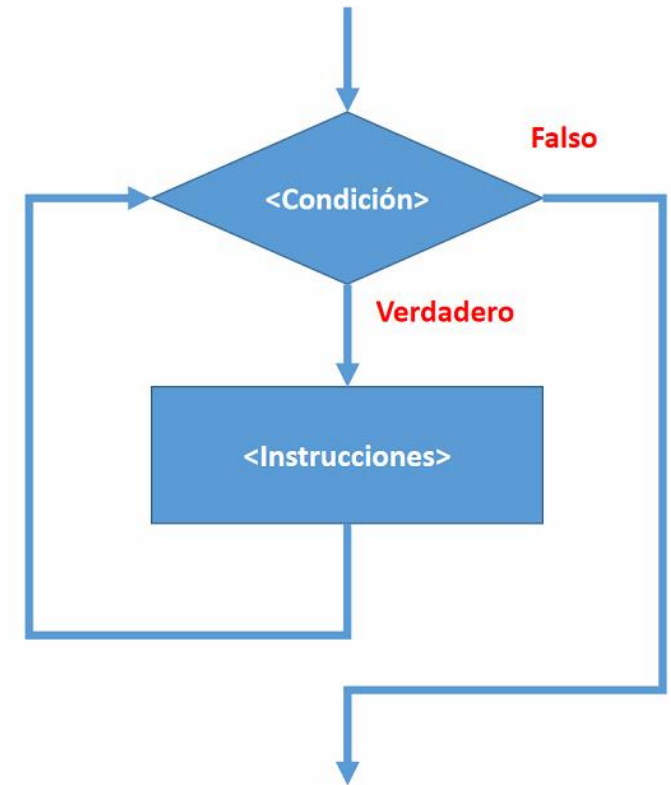
- Examinaremos las diferentes estructuras que utilizan los programadores para construir secciones de código repetitivas.
- Cada repetición se conoce como ***iteración*** o pasada a través del ***ciclo***.
- Estudiaremos los bucles:
 - While
 - Do - While
 - For





Estructura while

- La estructura “**while**”, es una estructura repetitiva que realiza la cantidad de iteraciones que uno determine.
- Necesita **siempre** una **variable** de tipo **int**, esta variable nos permitirá indicar desde donde inicia el bucle.





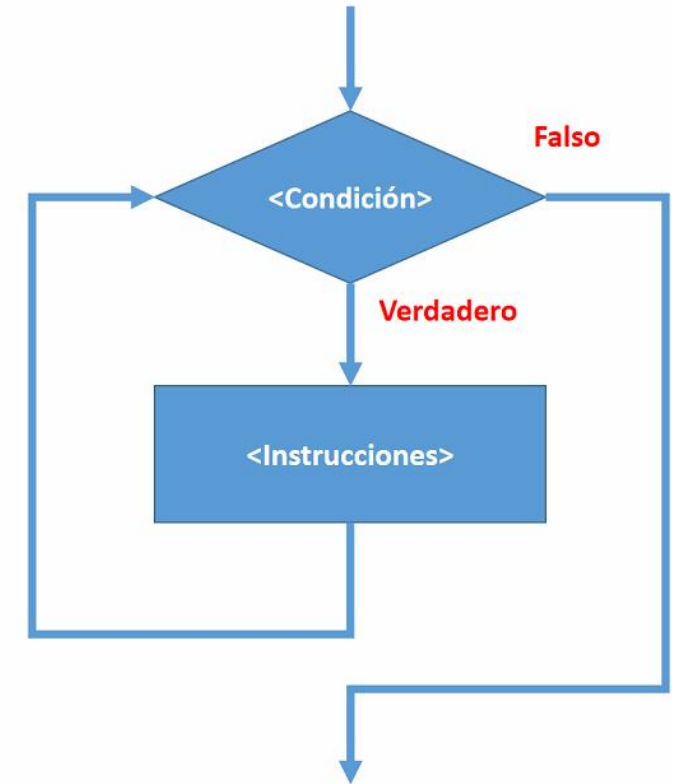
Estructura while

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      int a=1;
6
7      while (a<=10)
8      {
9
10
11         a++;
12     }
13
14 }
```

Variable **a** de tipo **int**, se **inicializa** por lo general en **1**

Condición (**a<=10**), indica que ejecutará 10 veces el bucle

Variable **a** antes de cerrar la llave del While debe de **incrementarse**





Ejemplo: while

```
1  /*
2  Ejemplo: Ingrese un numero "n",
3  y muestre la serie de 1 a "n".
4  */
5  #include<iostream>
6  using namespace std;
7  int main(){
8      // 1. Declarar variables
9      int n, cont = 1;
10     // 2.- Entrada de datos
11     cout << "Ingrese numero: ";
12     cin >> n;
13     // Proceso y Salida
14     while( cont <= n){
15         cout << cont << endl;
16         cont++;
17     }
18     return 0;
19 }
```

EjemploWhile.CPP

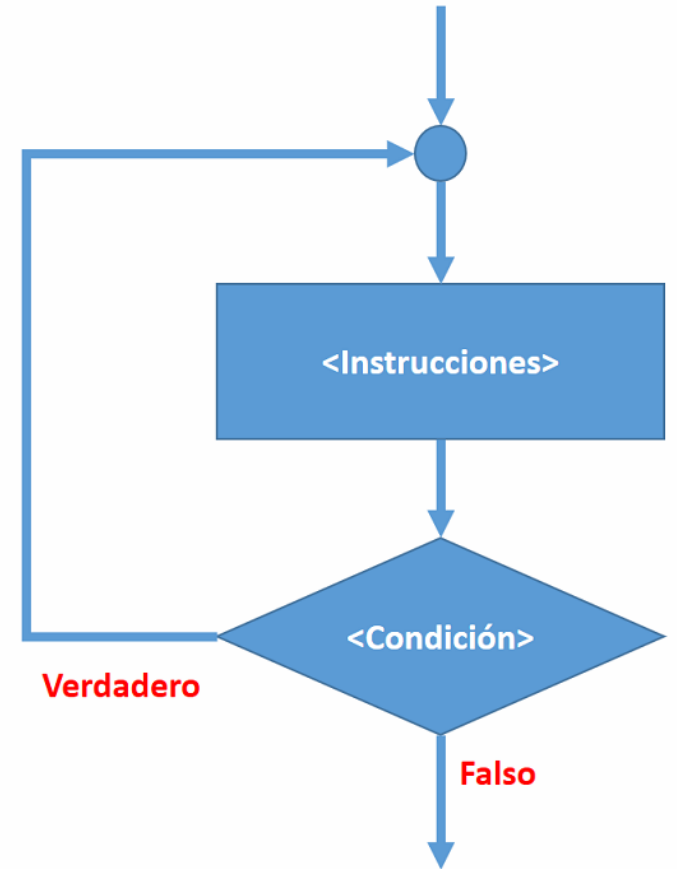
Ingrese numero: 6

1
2
3
4
5
6



Estructura do - while

- La estructura “**do-while**” es el equivalente a la estructura “Repetir - Hasta Que”.
- Do-While repite el bucle hasta que la condición es falsa, se ejecuta por lo menos una vez el bucle.

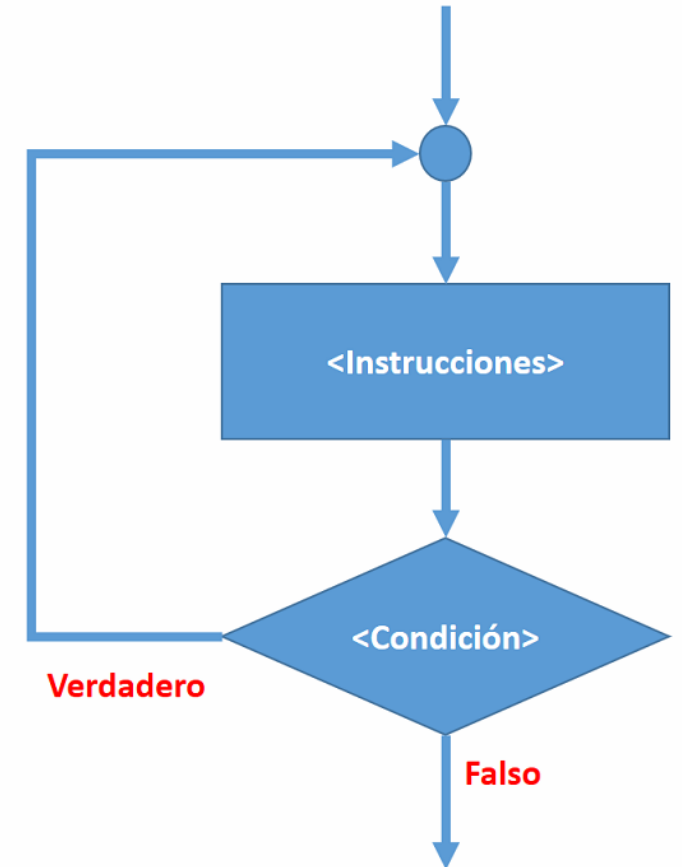




Estructura do - while

```
1  #include<iostream>
2  #include<math.h>
3  using namespace std;
4  int main()
5  {
6
7      do
8      {
9
10
11
12     }
13     while (condicion);
14
15 }
```

Aquí se coloca una condición falsa:
Ejemplo: $(n \leq 10)$
Entonces el bucle se ejecutará hasta que se ingrese un valor superior a 10





Ejemplo: do - while

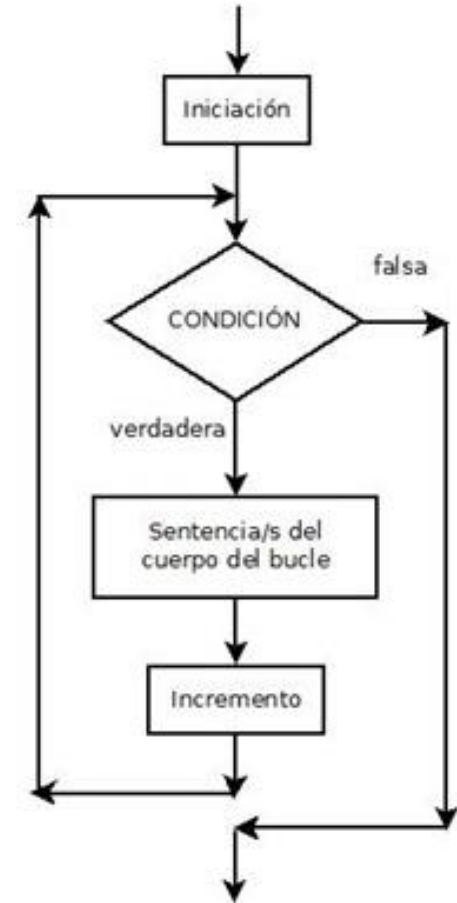
```
1  /*
2  Ejemplo: Ingrese un numero "n",
3  y muestre la serie de 1 a "n".
4  */
5  #include<iostream>
6  using namespace std;
7  int main(){
8      // Variables
9      int numero, suma = 0;
10     // Entrada y Proceso
11     cout << "Ingreso de numeros." << endl;
12     cout << "Ingreso 0 para finalizar." << endl;
13     do{
14         cout << "Numero: ";
15         cin >> numero;
16         suma += numero;
17     } while(numero != 0);
18     // Reporte
19     cout << "Reporte: " << suma << endl;
20     return 0;
21 }
```

```
EjemploDoWhile.CPP
Ingreso de numeros.
Ingreso 0 para finalizar.
Numero: 12
Numero: 56
Numero: 23
Numero: 80
Numero: 0
Reporte: 171
```



Estructura repetitiva For

La estructura “**For**”, permite ejecutar un conjunto de instrucciones de forma iterativa, conociendo un valor específico inicial y otro valor final, además nos permiten determinar el paso entre cada iteración del ciclo.





Sintaxis de estructura for

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13
```

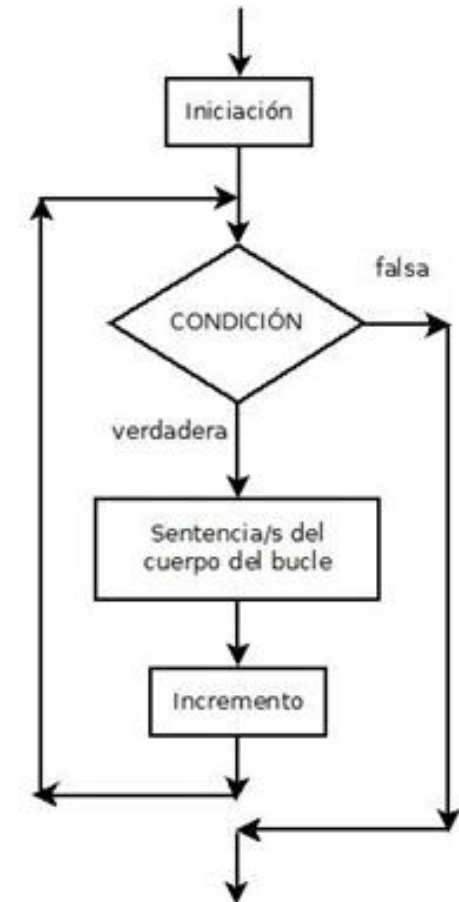
```
#include<iostream>  
using namespace std;  
int main()  
{  
    int i;  
    for(i=1; i<=10; i++)  
    {  
    }  
}
```

Se declara la variable **i** y siempre será de tipo **int**

Valor Inicial

Valor Final

Incremento





Ejemplo: for

```
1 //Muestra números de 10 al 100
2 #include<iostream>
3 using namespace std;
4 int main()
5 {
6     int i;
7     cout<<"Serie de 10 a 100 \n";
8     for (i=10; i<=100; i++)
9     {
10         cout<<i<<" , ";
11     }
12     cout<<endl;
13     return 0;
14 }
```

1. ¿Cuántos elementos generó, cuál es la suma?
2. ¿Puedo generar otras series?
3. ¿Puedo mostrar la serie en orden inverso?



Ejemplo: for

```
1 //Ejemplo: Calcular el factorial de un número
2 #include<iostream>
3 using namespace std;
4 int main()
5 {
6     int F=1,N,i;
7     cout<<"Ingrese un numero entero: ";
8     cin>>N;
9
10    for (i=1; i<=N; i++)
11    {
12        F=F*i;
13    }
14    cout<<"El factorial de "<<N<<" es "<<F;
15    cout<<endl<<endl;
16    system("pause");
17    return 0;
18 }
```



Mostrar los valores de i en los siguientes ciclos

a)

```
int i=12;
while (i<18)
{
    cout<<i<<" ";
    i=i+2;
}
```

- 12, 14, 16,
- 12, 14, 16, 18,
- 12, 14,

b)

```
int i=1;
do
{
    cout<<i<<" ";
    i++;
}
while (i<=5);
```

- 1, 2, 3,
- 1, 3, 5,
- 1, 2, 3, 4, 5,



Mostrar los valores de i en los siguientes ciclos:

c)

```
int i;
for (i=5; i<=15; i+=3)
{
    cout<<i<<" ";
}
```

- 5, 8, 11, 14,
- 5, 8, 11, 14, 17
- 5, 10, 15,

d)

```
int i;
for (i=5; i<=15; i++)
{
    if (i%5==0)
        cout<<i<<" ";
}
```

- 5, 7, 9, 11, 13, 15,
- 5, 10, 15, 20,
- 5, 10, 15,



Ejercicios





Elaborar los siguientes programas

1. Que solicite el ingreso de un número y muestre números múltiplos de 3 desde 0 hasta el número ingresado (`while`).
2. Que permita el ingreso de notas mientras sean entre 0 y 20, si se ingresa un nota no válida, finaliza mostrando la suma, cantidad de notas y el promedio de las notas ingresadas (`do-while`).



1.

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      int N,i=0;
6      cout<<"Ingrese numero entero: ";
7      cin>>N;
8      cout<<"Multiplos de 3 desde 0 hasta "<<N<<": \n";
9      while (i<=N)
10     {
11         if (i%3==0)
12             cout<<i<<", ";
13         i=i+1;
14     }
15     cout<<endl<<endl;
16
17     return 0;
}
```




2.

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      int can=0;
6      float prom,nota,sum=0;
7      do
8      {
9          cout<<"Ingrese nota (0-20): ";
10         cin>>nota;
11         sum=sum+nota;
12         can=can+1;
13     }
14     while (nota>=0 && nota<=20);
15     cout<<"\nFin...Ingreso una nota no valida\n";
16     cout<<"Suma de notas "<<sum<<endl;
17     cout<<"Cantidad de notas ingresadas "<<can-1<<endl;
18     cout<<"Promedio "<<(sum-nota)/(can-1);
19     cout<<endl<<endl;
20
21     return 0;
```



Elabora los siguientes programas (for)

1. Que solicite el ingreso de un número y muestre números múltiplos de 7 desde 10 hasta el número ingresado (for).
2. Que permita el ingreso de N notas (de 0 a 20), mostrar el promedio de las notas ingresadas (for).
3. Que solicite el ingreso de N números y luego muestre la cantidad de pares, impares, positivos y negativos (for).



1.

```
1  #include<iostream>
2  using namespace std;
3
4  int main()
5  {
6      int n;
7      float nota,sum=0;
8      cout<<"Ingrese numero mayor de 10: ";
9      cin>>n;
10     cout<<"\nSerie de 10 a "<<n<<" , multiplos de 7: \n";
11     for (int i=10; i<=n; i++)
12     {
13         if (i%7==0)
14             cout<<i<<" , ";
15     }
16     cout<<endl;
17     return 0;
18 }
```



2.

```
1  #include<iostream>
2  using namespace std;
3
4  int main()
5  {
6      int N;
7      float nota,sum=0;
8      cout<<"Ingrese cantidad de notas: ";
9      cin>>N;
10     cout<<endl;
11     for (int i=1; i<=N; i++)
12     {
13         cout<<"Ingrese nota "<<i<<": ";
14         cin>>nota;
15         if (nota<0||nota>20)
16         {
17             cout<<"Nota no valida, ingrese nota correcta \n";
18             i--;
19         }
20         else
21             sum+=nota;
22     }
23     cout<<"\nEl promedio es "<<sum/N;
24     cout<<endl;
25     return 0;
26 }
```



3.

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      int can,i,N,par=0,imp=0,posi=0,nega=0;
6      cout<<"Cuantos numeros desea ingresar?: ";
7      cin>>can;
8      for (i=1; i<=can; i++)
9      {
10         cout<<"Ingrese numero "<<i<<": ";
11         cin>>N;
12         if (N%2==0)
13             par++;
14         else
15             imp++;
16         if (N>0)
17             posi++;
18         else
19             nega++;
20     }
21     cout<<"\nCantidad de pares: "<<par;
22     cout<<"\nCantidad de impares: "<<imp;
23     cout<<"\nCantidad de positivos: "<<posi;
24     cout<<"\nCantidad de negativos: "<<nega;
25     cout<<endl<<endl;
26     system("pause");
27     return 0;
28 }
```



Preguntas





Reflexionemos



ucontinental.edu.pe

