



Estructuras de control para la programación: Estructuras de control repetitiva (Hacer- Hacer Mientras-Desde o Para)

FUNDAMENTOS DE PROGRAMACIÓN



Teoría

Reconoce la sintaxis de la estructura de control para la programación repetitivas: Mientras, Hacer Mientras y Para/Desde, en el lenguaje de programación



RECUERDA: Estructura de Control Repetitiva Mientras

Ejemplo:

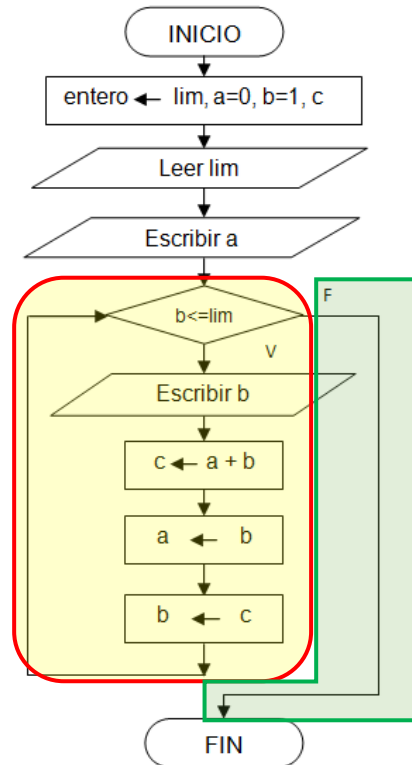
Mostrar la serie Fibonacci, menor a un límite dado:

Solución:

Variables:

a: valor de primer numero
b: valor de segundo numero
c: valor de la suma de la serie

Diagrama de Flujo:



Programa en C/C++: Semana5Programa1.cpp

Tenemos:

0	1	1	2	3
---	---	---	---	---

```
1  #include<iostream>
2  using namespace std;
3
4  int main( )
5  {
6      setlocale(LC_CTYPE, "Spanish");
7
8      int lim, a=0, b=1;      a=0  b=1
9      int c;
10
11     cout<<"Ingrese límite: ";
12     cin>>lim;               4
13     cout<<a;
14
15     while( b <= lim )
16     {
17         cout<<b;
18         c=a+b;
19         a=b;
20         b=c;
21     }
22
23     return 0;
24 }
```

while(1<=4)	while(1<=4)	while(2<=4)	while(3<=4)	while(5<=4)
1	1	2	3	
c=0+1=1	c=1+1=2	c=1+2=3	c=2+3=5	
a=1	a=1	a=2	a=3	
b=1	b=2	b=3	b=5	

***fin de programa



RECUERDA: Estructura de Control Repetitiva Hacer-Mientras

Ejemplo:

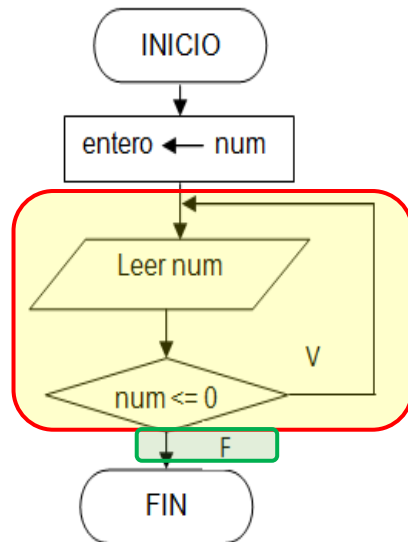
Leer un número entero mayor que cero, sino es así, volver a leer.

Solución:

Variables:

num: valor de numero

Diagrama de Flujo:



Programa en C/C++: Semana5Programa2.cpp

```
1 #include<iostream>
2 using namespace std;
3
4 int main( )
5 {
6     setlocale(LC_CTYPE,"Spanish");
7
8     int num;
9
10    do{
11        cout<<"Ingrese un numero:\t";
12        cin>>num;
13        if(num <= 0)
14            cout <<"\nERROR. Ingrese nuevamente.\n\n";
15        }while(num <= 0);
16
17    return 0;
18 }
```

CASO I

CASO II

do

do

-4

8

if(-4<=0)

if(8<=0)

ERROR.

while(-4<=0)

while(8<=0)

V

F

***fin de programa



RECUERDA: Estructura de Control Repetitiva Desde o Para

Ejemplo:

Calcular el Factorial de un número.

Solución:

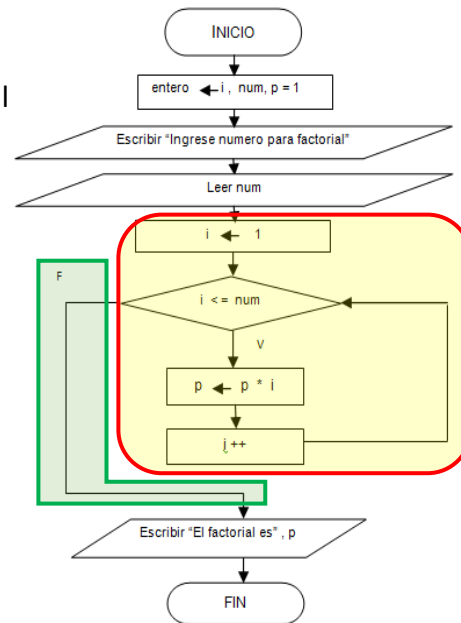
Variables:

num: valor de numero para factorial

p = valor del producto

i = valor inicial de recorrido

Diagrama de Flujo:



Programa en C/C++: Semana5Programa3.cpp

```
1 #include<iostream>
2 using namespace std;
3
4 int main( )
5 {
6     setlocale(LC_CTYPE,"Spanish");
7
8     int num, i, p=1;
9
10    cout<<"Ingrese numero para factorial: ";
11    cin>>num;
12
13    for( i=1; i<=num; i++ )
14    {
15        p = p * i;
16    }
17
18    cout<<"\nEl factorial de "<< num << " es: " << p;
19
20    return 0;
21 }
```

i= 1; 1<=4

p =1*1=1

i= 2; 2<=4

p =1*2=2

i= 3; 3<=4

p =2*3=6

i= 4; 4<=4

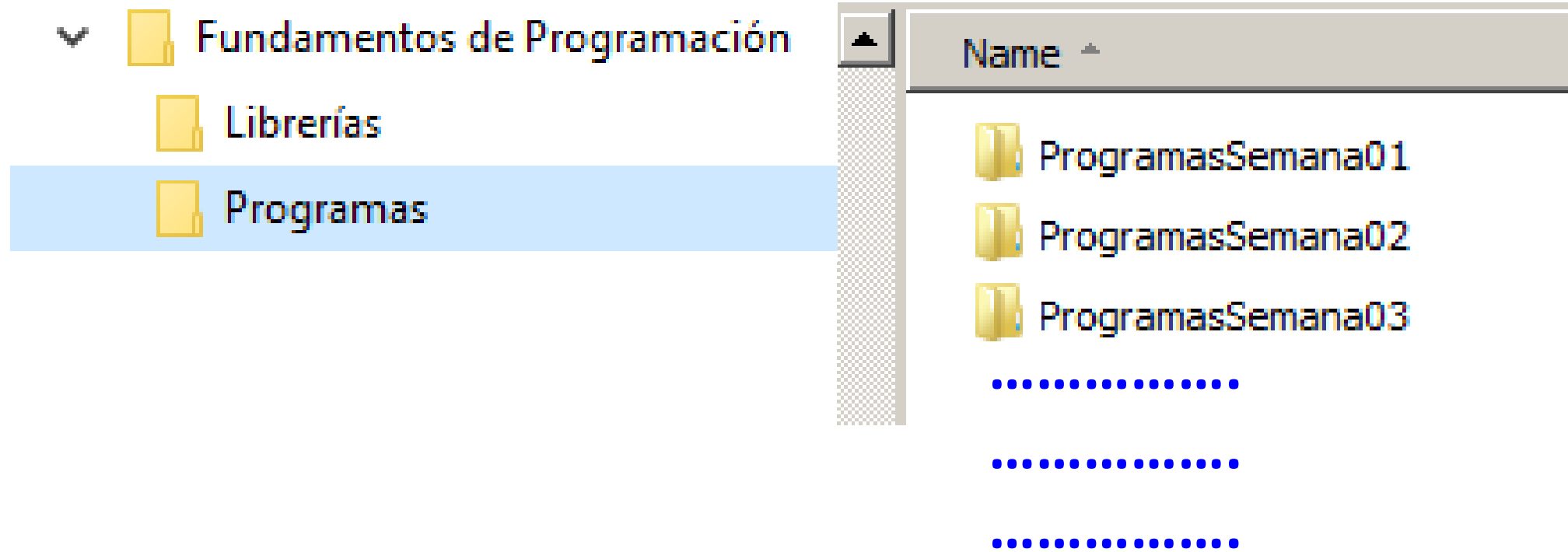
p =6*4=24

i= 5; 5<=4

Factorial es 24



Recomendación para crear los programas:



Ahora, elaboremos los programas ejemplos..!



Ejemplo: Ventas Repetitivas

```
1  #include<iostream>
2  using namespace std;
3
4  int main( )
5  {
6      double Cant, Prec, dcto, MBruto, MDcto, MFinal;
7
8      do{
9          cout<<"Ingrese cantidad:\t";
10         cin>>Cant;
11
12         if(Cant <= 0 )
13             cout<<"ERROR. Vuelva a Ingresa >0.\n";
14     }while(Cant <=0 );
15
16     do{
17         cout<<"Ingrese precio:\t\t";
18         cin>>Prec;
19
20         if(Prec <= 0 )
21             cout<<"ERROR. Vuelva a Ingresa >0.\n";
22     }while(Prec <=0 );
23
24     if(Cant >0 && Cant <=2 )
25         dcto = 0;
26     else
27         if(Cant >2 && Cant <=5 )
28             dcto = 0.1;
29         else
30             if(Cant >5 && Cant <=10 )
31                 dcto = 0.15;
32             else
33                 if(Cant >10 )
34                     dcto = 0.2;
35
36     MBruto= Prec * Cant;
37     MDcto = MBruto * dcto;
38     MFinal = MBruto - MDcto;
39
40     cout<<"El monto Final es:\t" << MFinal<<endl;
41     cout<<"\n";
42
43     return 0;
44 }
45
```

11 Sentencias Secuenciales
02 Sentencias Repetitivas Hacer-Mientras
01 Sentencia Selectiva Compuesta

Valida Datos

Valida Datos

Verifica Datos

Verifica Datos

Verifica Datos

Verifica Datos



Ejemplo: Ventas por Genero

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      double precioProd, MBruto, MBono, MDcto, MTtotal, bono, dcto;
7      int cantProd;   char genero;
8
9      do{
10         cout << "Ingrese cantidad:\t\t";
11         cin >> cantProd;
12
13         if(cantProd <= 0)
14             cout<<"ERROR. Vuelva a Ingresar >0."<<"\n\n";
15
16     }while(cantProd <= 0);
17
18     do{
19         cout << "Ingrese precio:\t\t\t";
20         cin >> precioProd;
21
22         if (precioProd <= 0)
23             cout<<"ERROR. Vuelva a Ingresar >0."<<"\n\n";
24
25     }while(precioProd <= 0);
26
27     do{
28         cout << "Ingrese inicial de genero:\t";
29         cin >> genero;
30
31         if (genero != 'F' && genero != 'f' && genero != 'M' && genero != 'm')
32             cout<<"ERROR. Vuelva a Ingresar F o f o M o m."<<"\n\n";
33
34     }while(genero != 'F' && genero != 'f' && genero != 'M' && genero != 'm');
35
36     switch (genero)
37     {
38         case 'M':
39         case 'm': {   bono = 0.18;
40                     if(cantProd <= 10)
41                         dcto = 0.2;
42                     else
43                         if(cantProd > 10)
44                             dcto = 0.5;
45                     } break;
46
47         case 'F':
48         case 'f': {   bono = 0.25;
49                     if(cantProd <= 10)
50                         dcto = 0.3;
51                     else
52                         if(cantProd > 10)
53                             dcto = 0.4;
54                     } break;
55     }
56
57     MBruto = cantProd * precioProd;
58     MBono  = MBruto * bono;
59     MDcto  = MBruto * dcto;
60     MTtotal = MBruto + MBono - MDcto;
61     cout << "El monto bruto es:\t\t" << MBruto << endl;
62     cout << "El monto de bonificacion es:\t" << MBono << endl;
63     cout << "El monto de descuento es:\t" << MDcto << endl;
64     cout << "El monto total de pago es:\t" << MTtotal << endl;
65     cout<<"\n";
66
67     return 0;
68 }
```

17 Sentencias Secuenciales
03 Sentencias Repetitivas Hacer-Mientras
01 Sentencia Selectiva Múltiple

Valida Datos

Valida Datos

Valida Datos

Verifica Datos

Verifica Datos

Verifica Datos

Verifica Datos



Conclusiones

Estructura de Control Repetitiva

Puede ser:

- Mientras
- Hacer Mientras
- Desde o Para

Se usa para validar el ingreso de datos

Se usa para repetir (menú de opciones) acciones

Programa

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     double precioProd, MBruto, MBono, MDcto, MTTotal, bono, dcto;
7     int cantProd;   char genero;
8
9     do{
10         cout << "Ingrese cantidad:\t\t";
11         cin >> cantProd;
12
13         if(cantProd <= 0)
14             cout<<"ERROR. Vuelva a Ingrese >0."<<"\n\n";
15     }while(cantProd <= 0);
16
17     do{
18         cout << "Ingrese precio:\t\t\t";
19         cin >> precioProd;
20
21         if (precioProd <= 0)
22             cout<<"ERROR. Vuelva a Ingrese >0."<<"\n\n";
23     }while(precioProd <= 0);
24
25     do{
26         cout << "Ingrese inicial de genero:\t";
27         cin >> genero;
28
29         if (genero != 'F' && genero != 'f' && genero != 'M' && genero != 'm')
30             cout<<"ERROR. Vuelva a Ingrese F o f o M o m."<<"\n\n";
31     }while(genero != 'F' && genero != 'f' && genero != 'M' && genero != 'm');
32
33     switch (genero)
34     {
35         case 'M':
36         case 'm': {   bono = 0.18;
37                     if(cantProd <= 10)
38                         dcto = 0.2;
39                     else
40                         if(cantProd > 10)
41                             dcto = 0.5;
42                     } break;
43
44         case 'F':
45         case 'f': {   bono = 0.25;
46                     if(cantProd <= 10)
47                         dcto = 0.3;
48                     else
49                         if(cantProd > 10)
50                             dcto = 0.4;
51                     } break;
52     }
53
54     MBruto = cantProd * precioProd;
55     MBono = MBruto * bono;
56     MDcto = MBruto * dcto;
57     MTTotal = MBruto - MDcto;
58     cout << "El monto bruto es:\t\t" << MBruto << endl;
59     cout << "El monto de bonificacion es:\t" << MBono << endl;
60     cout << "El monto de descuento es:\t" << MDcto << endl;
61     cout << "El monto total de pago es:\t" << MTTotal << endl;
62     cout<<"\n";
63
64     return 0;
65 }
```





Práctica

Crea programas con estructura de control para la programación repetitivas: Mientras, Hacer Mientras y Para/Desde, en el lenguaje de programación C/ C++.



Desarrollamos la Guía Práctica 05

Guía práctica N°05

Fundamentos de Programación

Segunda Unidad: Estructuras de control para la programación: repetitivas

Sección :	Apellidos :
Docente :	Nombres :
	Fecha : / / Duración: 100 min
	Tipo de práctica: Individual (x) Grupal ()

Instrucciones: Lea detenidamente cada enunciado y desarrolle lo solicitado.
Utilizar el Dev C++ para el desarrollo de los siguientes programas

1. Elaborar un programa para ingresar el valor de la variable x y de la variable y, validando con DO WHILE (que sean mayor igual a cero), y permita calcular:

$$z = \sqrt{x + y^3}$$

2. Elaborar un programa para ingresar el precio de 10 artículos y calcule el total de los precios, si este es mayor a 200 se le aplicará un descuento del 15% muestre el total a pagar y el descuento:
3. Elaborar un programa para ayudar a un padre que, con la intención de motivar el estudio en su hijo, le dice que será compensado según su promedio de tres ciclos consecutivos. Calcular y mostrar la compensación monetaria, según el siguiente cuadro:

RANGO DE NOTA	FACTOR A MULTIPLICAR
0 - 5	50
6 - 12	80
13 - 17	120
18 - 20	150

Ejemplo: con tres ciclos consecutivos, cuyos promedios son 05, 11 y 20.
La compensación es : $50 + 80 + 150 = 280$

4. Elaborar un programa para ingresar la edad de n personas e indique cuantos son mayores de edad y cuantos son menores de edad.

Nota: Ya que "n" es una incógnita, se debe considerar el uso del DO WHILE, para indicar si desea seguir ingresando otro valor de edad, a través de una respuesta SI: S o s - NO: N o n.

5. Elaborar un programa para ingresar n salarios y aplique una bonificación del 15% a aquellos salarios menores a 3000, muestre cada nuevo valor del salario.

Nota: Ya que "n" es una incógnita, se debe considerar el uso del DO WHILE, para indicar si desea seguir ingresando otro valor de salario, a través de una respuesta SI: S o s - NO: N o n.

Carpeta: Programas Semana 05

Nombre Proyecto: Proyecto3_EcuacionValidada
NombrePrograma: Programa3_EcuacionValidada

Carpeta: Programas Semana 05

Nombre Proyecto: Proyecto4_DescuentoArticulo
NombrePrograma: Programa4_DescuentoArticulo

Carpeta: Programas Semana 05

Nombre Proyecto: Proyecto5_Compensacion
NombrePrograma: Programa5_Compensacion

Carpeta: Programas Semana 05

Nombre Proyecto: Proyecto6_ConjuntoEdades
NombrePrograma: Programa6_ConjuntoEdades

Carpeta: Programas Semana 05

Nombre Proyecto: Proyecto7_ConjuntoSalarios
NombrePrograma: Programa7_ConjuntoSalarios

Referencias bibliográficas consultadas y/o enlaces recomendados

- JOYANES, L. (2008). Fundamentos de Programación. Algoritmos, estructura de datos y objetos, 4ª Edición. Madrid: McGraw-Hill.
- Beekman, G., Pacheco, R. y Tábora, A. (2008). Introducción a la computación. México: Pearson Educación.



Conclusiones

Estructura de Control Repetitiva

Puede ser:

- Mientras
- Hacer Mientras
- Desde o Para

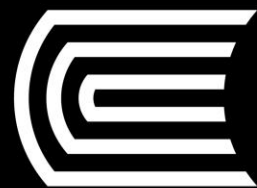
Se usa para validar el ingreso de datos

Se usa para repetir (menú de opciones) acciones

Programa

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     double precioProd, MBruto, MBono, MDcto, MTTotal, bono, dcto;
7     int cantProd;   char genero;
8
9     do{
10         cout << "Ingrese cantidad:\t\t";
11         cin >> cantProd;
12
13         if(cantProd <= 0)
14             cout<<"ERROR. Vuelva a Ingresar >0."<<"\n\n";
15     }while(cantProd <= 0);
16
17     do{
18         cout << "Ingrese precio:\t\t\t";
19         cin >> precioProd;
20
21         if (precioProd <= 0)
22             cout<<"ERROR. Vuelva a Ingresar >0."<<"\n\n";
23     }while(precioProd <= 0);
24
25     do{
26         cout << "Ingrese inicial de genero:\t";
27         cin >> genero;
28
29         if (genero != 'F' && genero != 'f' && genero != 'M' && genero != 'm')
30             cout<<"ERROR. Vuelva a Ingresar F o f o M o m."<<"\n\n";
31     }while(genero != 'F' && genero != 'f' && genero != 'M' && genero != 'm');
32
33     switch (genero)
34     {
35         case 'M':
36         case 'm': {   bono = 0.18;
37                     if(cantProd <= 10)
38                         dcto = 0.2;
39                     else
40                         if(cantProd > 10)
41                             dcto = 0.5;
42                     } break;
43
44         case 'F':
45         case 'f': {   bono = 0.25;
46                     if(cantProd <= 10)
47                         dcto = 0.3;
48                     else
49                         if(cantProd > 10)
50                             dcto = 0.4;
51                     } break;
52     }
53
54     MBruto = cantProd * precioProd;
55     MBono = MBruto * bono;
56     MDcto = MBruto * dcto;
57     MTTotal = MBruto - MDcto;
58     cout << "El monto bruto es:\t\t" << MBruto << endl;
59     cout << "El monto de bonificacion es:\t\t" << MBono << endl;
60     cout << "El monto de descuento es:\t\t" << MDcto << endl;
61     cout << "El monto total de pago es:\t\t" << MTTotal << endl;
62     cout<<"\n";
63     return 0;
64 }
```





ucontinental.edu.pe