

Estructuras repetitivas

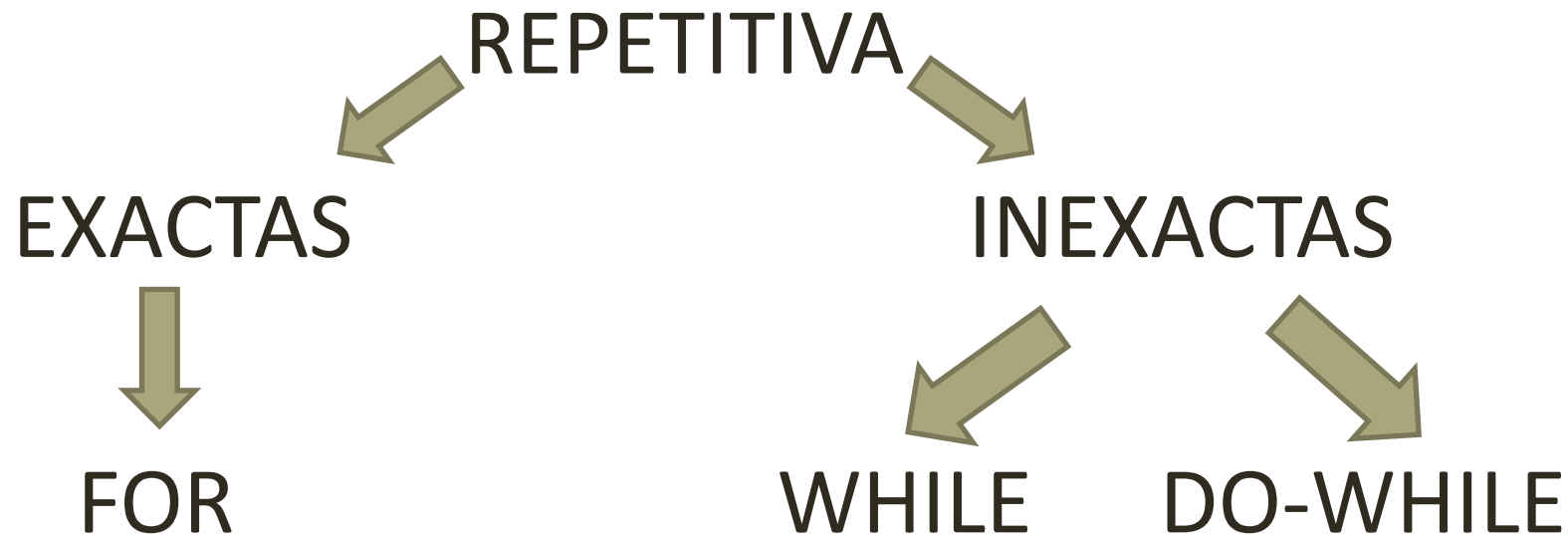
Uso de repetitivas For, While y Do-While



Desde 1936 educando para la vida.

¿QUÉ ES UNA ESTRUCTURA REPETITIVA?

- Mecanismos para **repetir** un conjunto de instrucciones **hasta que se cumple cierta condición**.



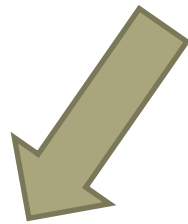
¿QUÉ PUEDE EJECUTARSE EN UNA REPETITIVA?

- Puedo repetir un ingreso de datos? **Sí.**
- Puedo incorporar una estructura condicional dentro de una repetitiva? **Sí.**
- Puedo realizar una repetitiva dentro de otra? **Sí.**
- Recordemos que, en el primer caso, luego de la primera repetición y al ingresar el segundo dato sobre la misma variable, lo ingresado anteriormente se pierde.
- ***Por esto es importante, conocer el significado y el uso de enteros contadores y acumuladores***

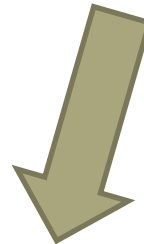
REPETITIVA EXACTA FOR

- Al ser exacta **conocemos de antemano** cuándo va a finalizar su ejecución
- Forma estándar de definir de la sentencia:

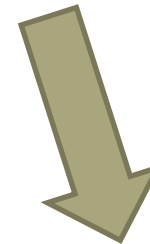
```
for ( int i = 0; i < tope; i++ )
```



Defino la variable que
Controla el FOR



i como máximo puede
valer tope



cuando termine de
ejecutarse cada ciclo del
For, i crece en 1

Este ciclo se ejecutaría **tope** veces (donde tope **es un número entero**)

EJEMPLO DE USO DEL FOR

- Se plantea la situación de, utilizando la estructura FOR, mostrar todos los números impares entre 1 y 100.
- Nuevamente, lo primero que tengo que hacer es bajar a concreto. ¿Qué variable/s necesito?
 - Mínimamente, necesito, la variable en la cual estarían los números impares. En este caso, ¿por qué no aprovechar que la variable *i* va a tener todos los valores? (Obviamente, en repeticiones diferentes). Podría mostrarla **sólo cuando tome un valor impar**.

```
for(int i=1;i<100;i++)  
{  
    if (i % 2 ==1)  
    {  
        cout<<i<<endl;  
    }  
}  
getchar();
```

Sencillo, no?

OTRO EJEMPLO DEL FOR

- La cantidad de repeticiones del For **puede ser dada por un ingreso**
- La variable (en este caso N) es la que determina cuántas veces se mostrará la palabra "Hola"

```
#include <iostream>
using namespace std;

void main()
{
    int N;
    cout << "Ingrese un numero"<<endl;
    cin >> N;

    for(int i=0; i<N; i++)
    {
        cout << i+1 << "- Hola"<<endl;
    }

    system("Pause");
}
```

```
Ingrese un numero
4
1- Hola
2- Hola
3- Hola
4- Hola
Presione una tecla para continuar . . .
```

REPETITIVA INEXACTA WHILE

- No sabemos cuántas veces se repetirá nuestro código, incluso puede que nunca lo haga.
- Mientras **se cumpla una condición**, se repetirá el ciclo
- Este programa muestra todos los enteros consecutivos del 0 a N, con repetitiva **while**

```
#include <iostream>
using namespace std;

void main()
{
    int N, cont=0;
    cout <<"Ingrese numero"<<endl;
    cin>>N;

    while(N>=cont)
    {
        cout <<cont;
        if(N!=cont)cout<<"-";
        cont++;
    }
    cout <<endl;
    system("Pause");
}
```

```
Ingrese numero
100
0-1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18-19-20-21-
30-31-32-33-34-35-36-37-38-39-40-41-42-43-44-45-46-47-4
-57-58-59-60-61-62-63-64-65-66-67-68-69-70-71-72-73-74-
3-84-85-86-87-88-89-90-91-92-93-94-95-96-97-98-99-100
Presione una tecla para continuar . . .
```

Si ingreso un 0, la ejecución del programa no entra en el ciclo while

REPETITIVA INEXACTA DO-WHILE

- En este caso, la traducción del ciclo es **hacer-mientras**. Esto quiere decir que el ciclo se ejecutará por lo menos 1 vez, ya que la condición **se consulta al final** del bloque
- Este es el mismo programa que el anterior, pero con un ciclo do-while en vez de while.

```
#include <iostream>
using namespace std;

void main()
{
    int N, cont=0;
    cout <<"Ingrese numero"<<endl;
    cin>>N;

    do
    {
        cout <<cont;
        if(N!=cont)cout<<"-";
        cont++;
    }
    while(N>=cont);

    cout <<endl;
    system("Pause");
}
```


VALIDACIÓN UTILIZANDO DO-WHILE

```
#include <iostream>
using namespace std;

void main(){

    //-----FORMA DE VALIDACIÓN DE TIPO DE DATO-----//

    int N;
    do{
        cin.clear(); //se recupera del error
        fflush(stdin); //borra el buffer
        cout << "Ingresa un numero: ";
        cin >> N;
        if (cin.fail()) //la entrada tiene un formato que no coincide?
            cout << "NO ES UN NUMERO " << endl;
    } while (cin.fail());
    system("pause");

    //-----FORMA DE VALIDACIÓN DE TIPO DE DATO-----//

}
```



ORT
אורט
ARGENTINA



Desde 1936 educando para la vida.