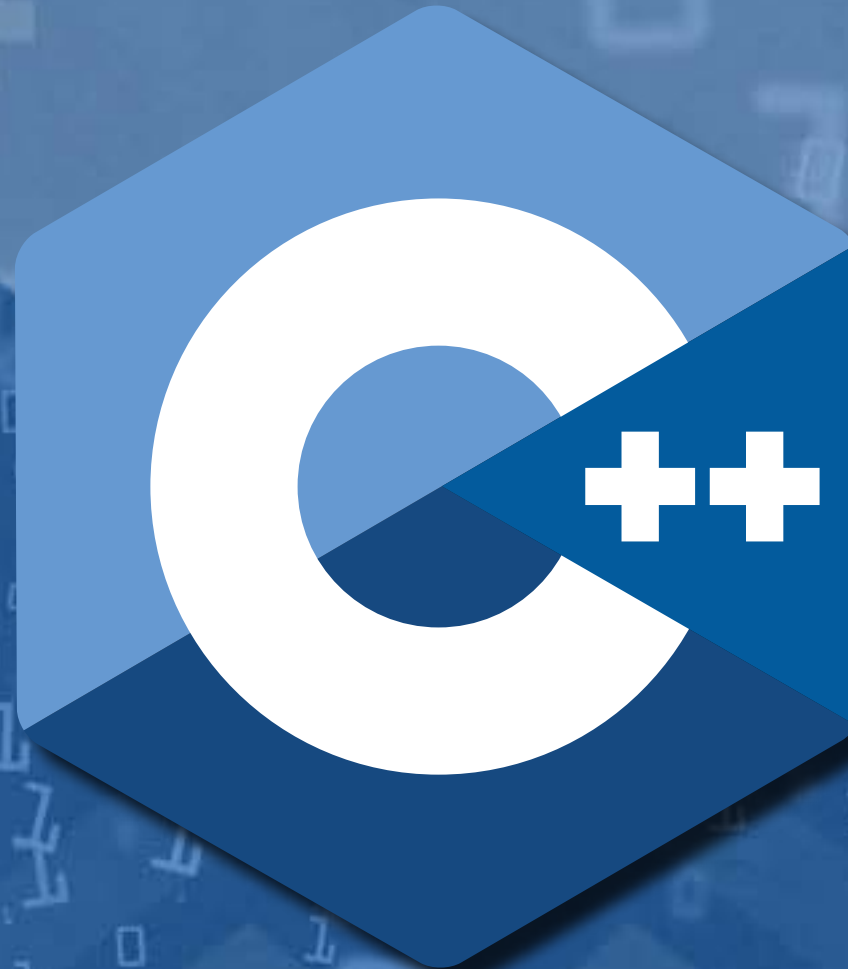


CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.



Material de Actividad de Aprendizaje 3



CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Material Actividad de Aprendizaje 3.

3. Estructuras de control por decisión o selección.

3.1 Estructuras de control por decisión simples.

3.2 Estructuras de control por decisión múltiples.



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Material Actividad de Aprendizaje 3.

◆ 3. Estructuras de control por decisión o selección.

En C++ la estructura de selección o decisión más importante es la sentencia.

if

Que denota una pregunta que debe ser contestada en términos de si o no solamente.

◆ 3.1 Estructuras de control por decisión simples.

Esta sentencia tiene varios formatos o formas de uso según la necesidad. La más simple es la siguiente.

if(expresion)Accion



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.



Donde la expresión, se refiere a cualquier combinación de valores, variables y operadores que respeten las normas sintácticas y que dan como resultado un valor de tipo booleano es decir falso o verdadero, que indica si la acción se debe ejecutar o no.

- Esta se lee de la siguiente manera;

```
if(a>b) c=3;  
si (a>b) entonces c=3;
```



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Si la acción que sigue al entonces, se realiza en una única sentencia, se puede usar sin llaves, de lo contrario se agrupa las sentencias que pertenecen a la decisión en cuestión así;

```
if(a>b)
{
    c=3;

    a=a+1;
}
```



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

◆ 3.2 Estructuras de control por decisión múltiples.

Las estructuras condicionales múltiples, permiten escoger una acción a seguir de entre varias opciones, al igual que el if sencillo, si la acción se cumple en una sola línea se puede omitir las llaves, de lo contrario son obligatorias.

```
if(a>b)
c=3;
else
h=5;
```

```
if(a>b)
{
c=3;
a=0;
}
else
{
h=5;
b=0;
}
```




C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

La palabra reservada **else**, significa de lo contrario y quiere decir que si no se cumple la condición, se ejecutará esa instrucción o ese grupo de instrucciones que preceden al (else). Las llaves al igual que en la función agrupa sentencias que pertenecen a un mismo bloque funcional.

```
if(a>b)
c=3;
else
h=5;
```

```
ifa>b
{
c=3;
a=0;
}
else
{
ifa>b
c=3;
else
h=5;
}
```

```
if(a>b)
{
if(a>b)
c=3;
else
h=5;
}
else
{
h=5;
b=0;
}
```



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Otras formas de uso del (if) son el anidamiento, que es cuando se incluye una estructura (if) dentro de otra estructura (if), como se puede ver en la tabla, indistintamente de donde se use. El nivel de anidamiento, prácticamente no tiene límite, más que la capacidad de seguir la lógica del programa, que suele complicarse luego de varios niveles:

Como se ve en la tabla, el (if) tiene una gran flexibilidad a la hora de implementarse, se puede usar adicionalmente la cláusula else y unirla a otra instrucción (if) y de esta manera volver a evaluar una expresión, o simplemente anidarse, es decir dentro de una estructura (if) otra estructura (if).





C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

```
if(a>b)
{
    c=3;
    a=0;
}
else if (a=4)
{
    h=5;
    b=0;
}
```

```
if(a>b)
{
    if(a>b)
    c=3;
    else
    h=5;
}
else if (a=4)
{
    h=5;
    b=0;
}
```



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.



Las instrucciones abajo mostradas, son equivalentes, sin embargo una expone un rendimiento superior, revisar y probar cuál es la opción a usar, ya que no se puede hablar de mejor opción, puesto que cada una cuenta con ventajas y depende del programador escoger la forma que lo haga sentir más cómodo.



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

▪ **Nota:** Para probar estos fragmentos de código, se recomienda asignar valores a las variables, y mantener un registro de ellas a medida que transcurre la ejecución del programa. Aquí se puede ver como se haría este proceso.

```
if(a>b)
{
    c=3;
    a=0;
}
else if (a==4)
{
    h=5;
    b=0;
}
```

Valores Iniciales

a	b	c	h
4	5	0	0

entonces

a > b ?	
↓ ↓	→ Resp
4 5	No

Como no se cumple la condición

→ de lo contrario

Si a = 4 ? entonces

Se Asigna a
h = 5; el valor 5
y a b

b = 0; el valor 0



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

```
if(a>b)
{
c=3;
a=0;
}
else if (a==4)
{
h=5;
b=0;
}
```

```
if(a>b)
{
c=3;
a=0;
}
else
{
if(a==4)
{
h=5;
b=0;
}
}
```



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Interpretar estos códigos hechos en C++, para que luego elaboren sus propios ejercicios.

```
#include <iostream>

// se incluye la librería iostream que maneja los flujos de entrada/salida

using namespace std;
//se usa el espacio de nombre std para evitar confusiones en los nombres

int main() //la función principal
{
    int a,b,c,h; //se declaran las variables
    //recuerda que C++ es un lenguaje fuertemente tipado
    a=4;
    b=0;
    c=0;
    h=0;

    if(a>b)
    {
        c=3;
        a=0;
    }
}
```





C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

```
else if(a==4)
{
    h=5;
    b=0;

}

cout<<"a="<<a<<" b="<<b<<" c="<<c<<" h="<<h<<endl;

cin>>a;

return 0;
}
```




C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Con los valores originales obtendrán una salida como esta.

A screenshot of a Windows command prompt window titled "M:\codeBlock\jtPrueba\bin\Debug\jtPrueba.exe". The window shows the output of a C++ program. The text displayed is: "a=0 b=0 c=3 h=0", "Estos son los valores de las variables", "Process returned 0 (0x0) execution time : 0.071 s", and "Press any key to continue.". A cursor is visible on the line "Press any key to continue.".

```
M:\codeBlock\jtPrueba\bin\Debug\jtPrueba.exe
a=0 b=0 c=3 h=0
Estos son los valores de las variables
Process returned 0 (0x0) execution time : 0.071 s
Press any key to continue.
-
```

Modifiquen los valores de las “variables” y observen el resultado. Intenten deducir el porqué de ese comportamiento.



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Otra forma particularmente útil de seleccionar una acción entre varias alternativas que dependen del valor de una variable; es la sentencia switch cuya sintaxis se describe a continuación.

```
switch(variable)
{
case 1:
Accion 1
break;
case 2:
Accion 2
break;
case 3:
Accion 3
break;
case n:
Accion n
break;
}
```



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Donde la variable controla la opción de la acción, por la que se decidirá continuar la ejecución del programa.

▪ **Aquí un ejemplo:**

```
#include <iostream>
// se incluye la libreria iostream que maneja los flujos de entrada/salida

using namespace std;
//se usa el espacio de nombre std para evitar confuciones en los nombres

int main() //la funcion principal
{
    int a;

    a=3;

    switch(a)
    {
```





C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

```
case 0:  
    cout<<"caso 0";  
    break;  
case 1:  
    cout<<"caso 1";  
    break;  
case 2:  
    cout<<"caso 2";  
    break;  
case 3:  
    cout<<"caso 3";  
    break;  
  
}  
return 0;  
}
```



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

Ahora pidiendo el valor de la variable al usuario, solo se usa la sentencia `cin>>a` y listo.

```
#include <iostream>
// se incluye la libreria iostream que maneja los flujos de entrada/salida

using namespace std;
//se usa el espacio de nombre std para evitar confuciones en los nombres

int main() //la funcion principal
{

    int a;
    cout<<"Digite un valor para la variable a ";
    cin>>a;

    switch(a)
    {
    case 0:
```





C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

```
    cout<<"caso 0";  
    break;  
case 1:  
    cout<<"caso 1";  
    break;  
case 2:  
    cout<<"caso 2";  
    break;  
case 3:  
    cout<<"caso 3";  
    break;  
}  
return 0;  
}
```




C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.

- La salida que obtendrán es esta:

```
Mi\codeBlock\ytPrueba\bin\Debug\ytPrueba.exe
Digite un valor para la variable a 2
caso 2
Process returned 0 (0x0) execution time : 11.206 s
Press any key to continue.
```

Fueron probados los conceptos de estructuras de control por decisión. Ahora apliquen esos conocimientos en la resolución de problemas simples.

- **Nota:** todos los ejemplos se implementaron en codeblocks.



C CONCEPTUALIZACIÓN DEL LENGUAJE DE PROGRAMACIÓN C++.



◆ Webgrafía:

Codeblocks.org. (2013). Codeblocks-13. Consultado el 19 de abril de 2014, en <http://www.codeblocks.org/downloads/26#windows>.