



# PROGRAMACIÓN I

TUP - 2024 - 2C - Turno noche - Presencial  
Comisión 111

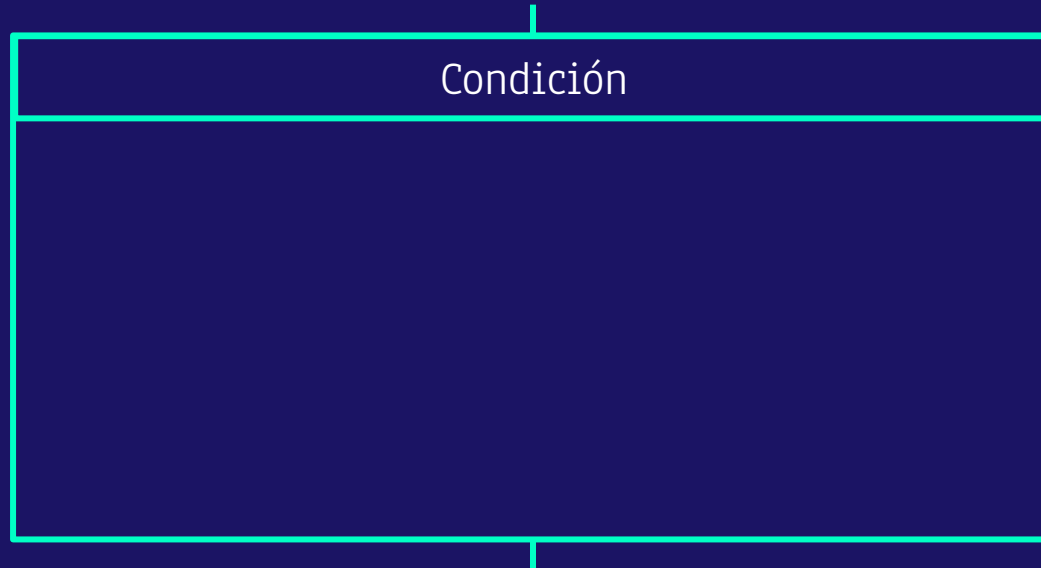


04

CICLOS – INEXACTO

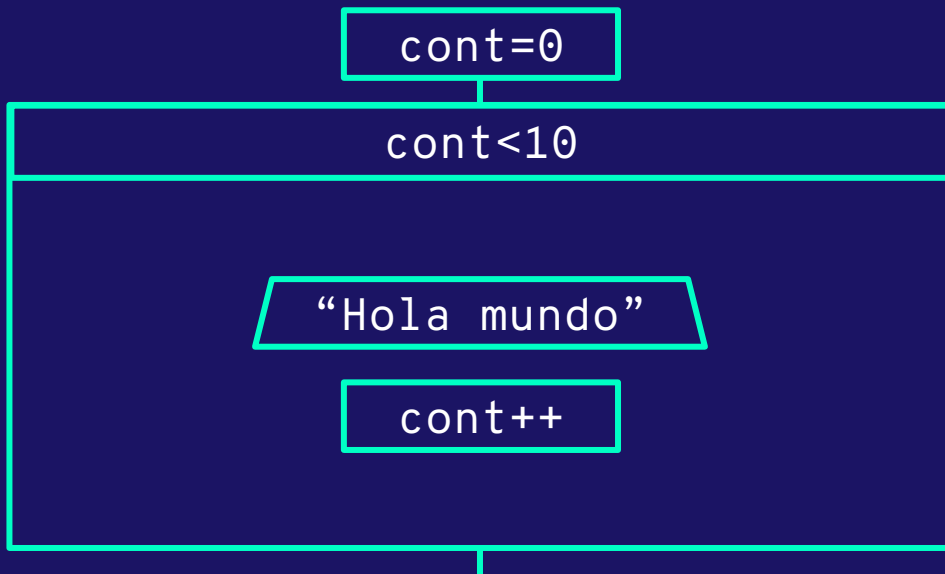
# CICLO INEXACTO – WHILE

El ciclo inexacto es una estructura repetitiva que nos permite realizar un conjunto de instrucciones una indeterminada cantidad de veces.



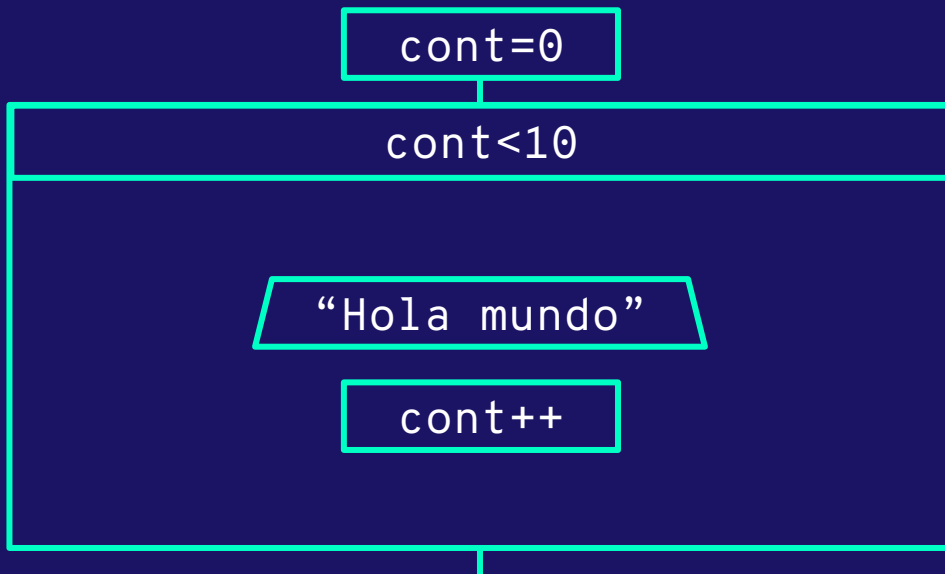
# DIAGRAMACIÓN

Ejemplo: Realizar un programa que nos muestre el mensaje “Hola mundo” por pantalla 10 veces.



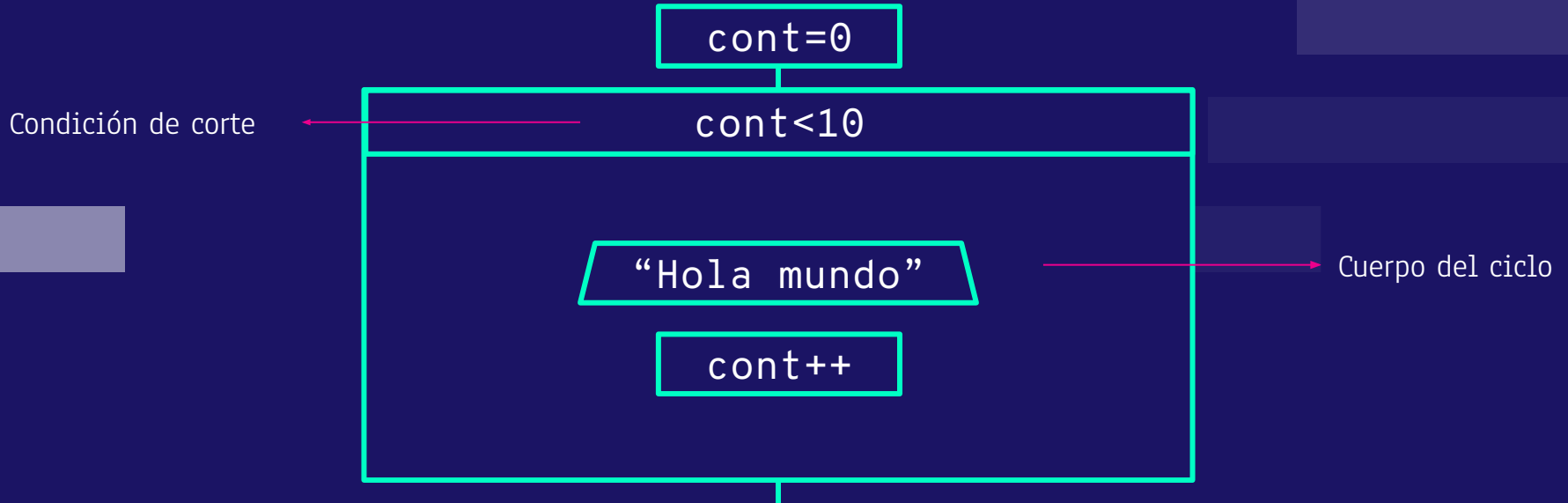
# DIAGRAMACIÓN

Ejemplo: Realizar un programa que nos muestre el mensaje “Hola mundo” por pantalla 10 veces.



# DIAGRAMACIÓN

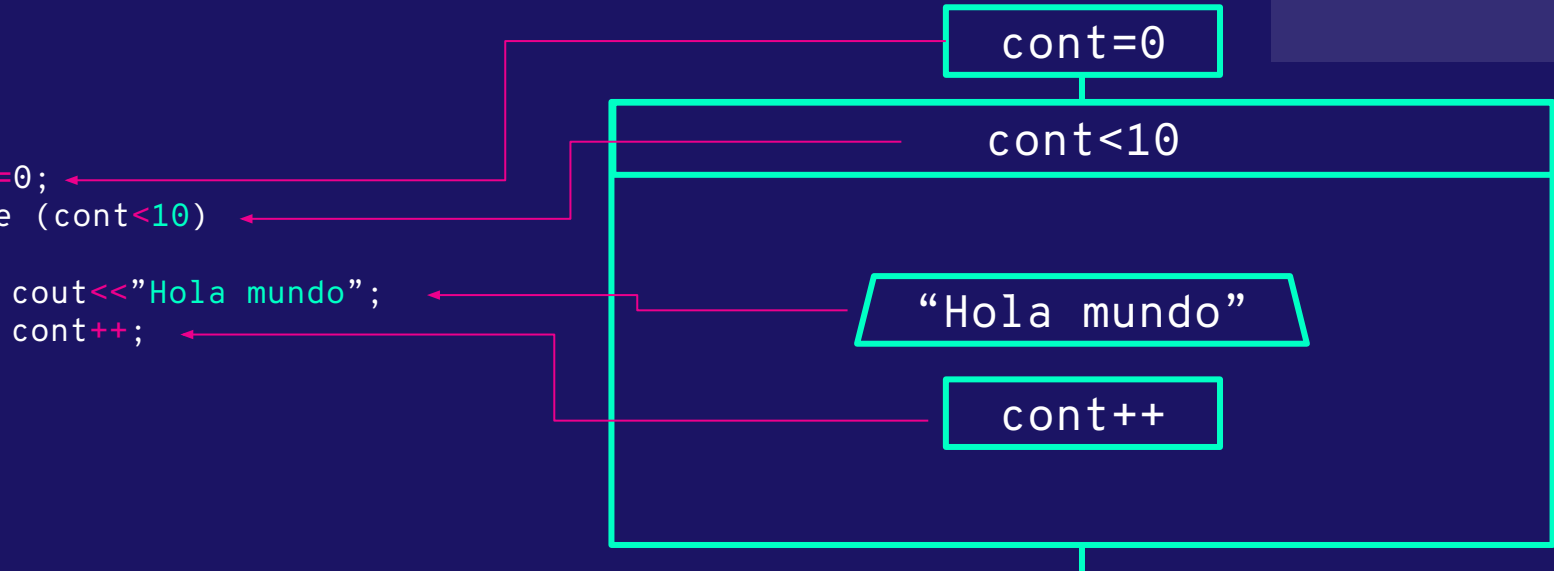
Ejemplo: Realizar un programa que nos muestre el mensaje "Hola mundo" por pantalla 10 veces.



# DEL DIAGRAMA AL CÓDIGO

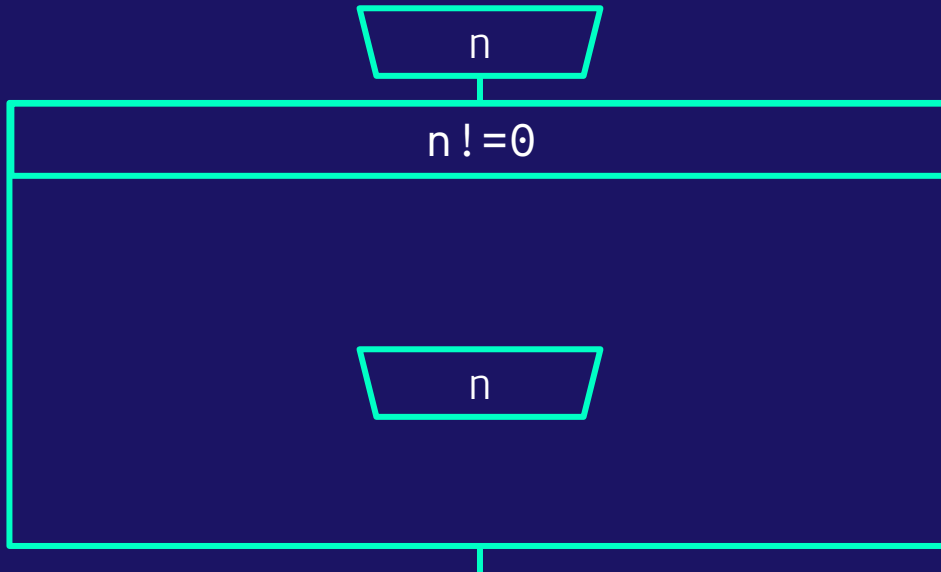
Ejemplo: Realizar un programa que nos muestre el mensaje "Hola mundo" por pantalla 10 veces.

```
cont=0;  
while (cont<10)  
{  
    cout<<"Hola mundo";  
    cont++;  
}
```



# DIAGRAMACIÓN

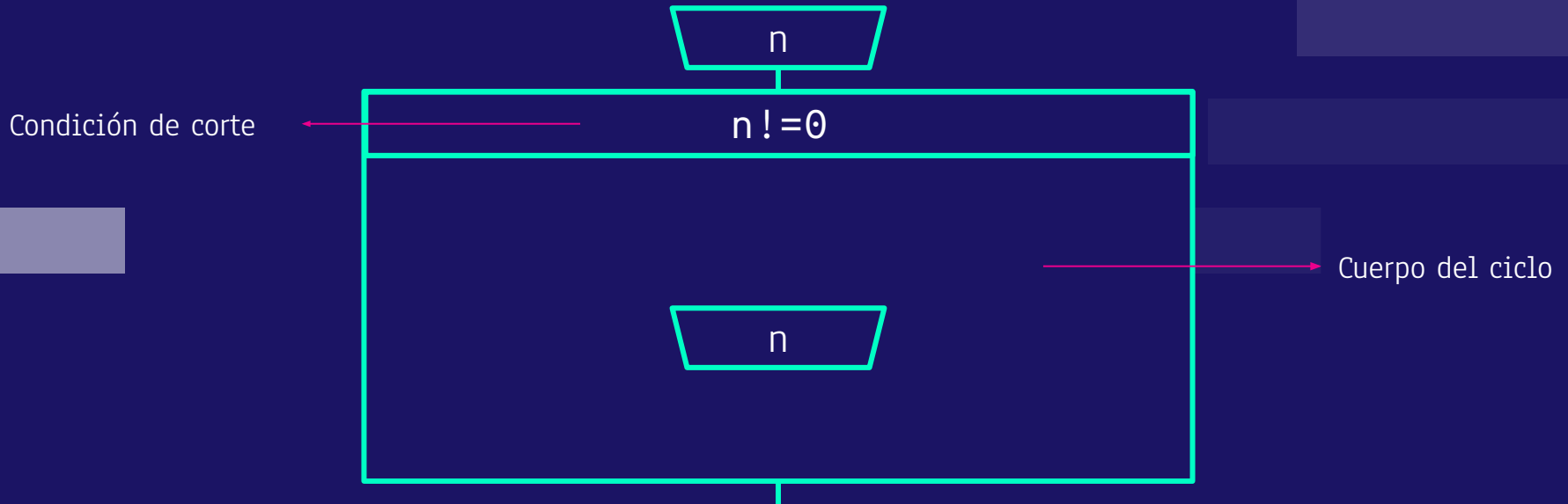
Otro ejemplo: Realizar un programa que solicite números y finalice cuando se ingrese un cero.





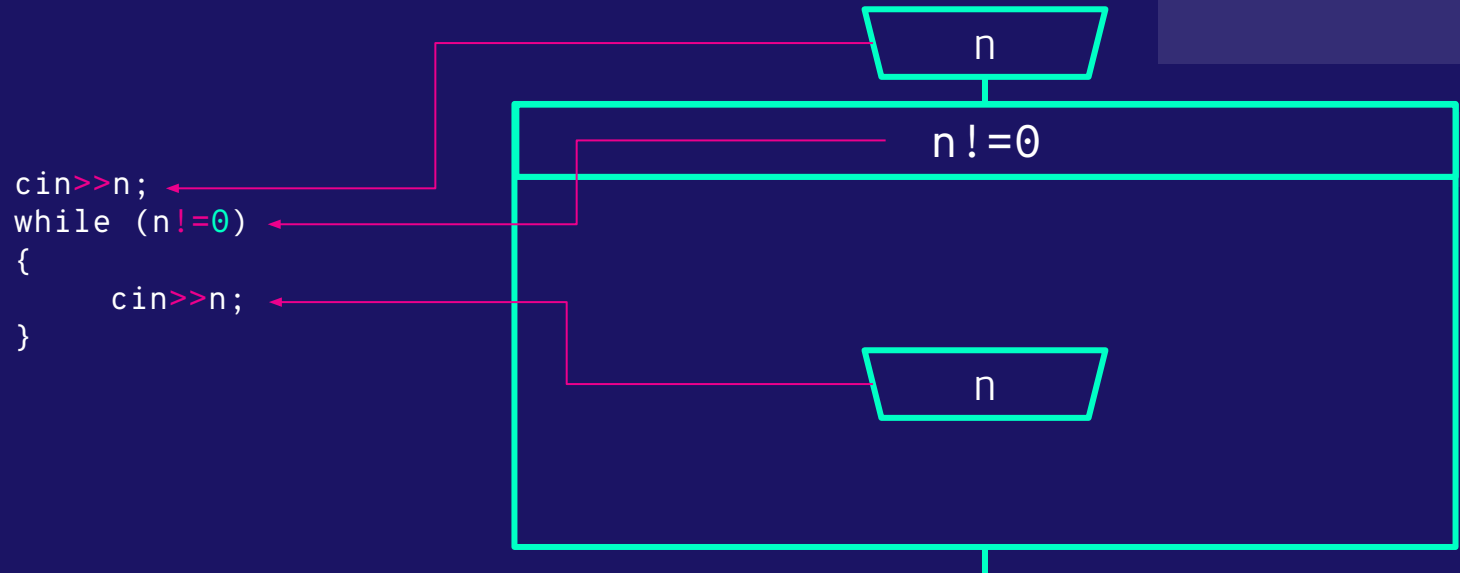
# DIAGRAMACIÓN

Otro ejemplo: Realizar un programa que solicite números y finalice cuando se ingrese un cero.



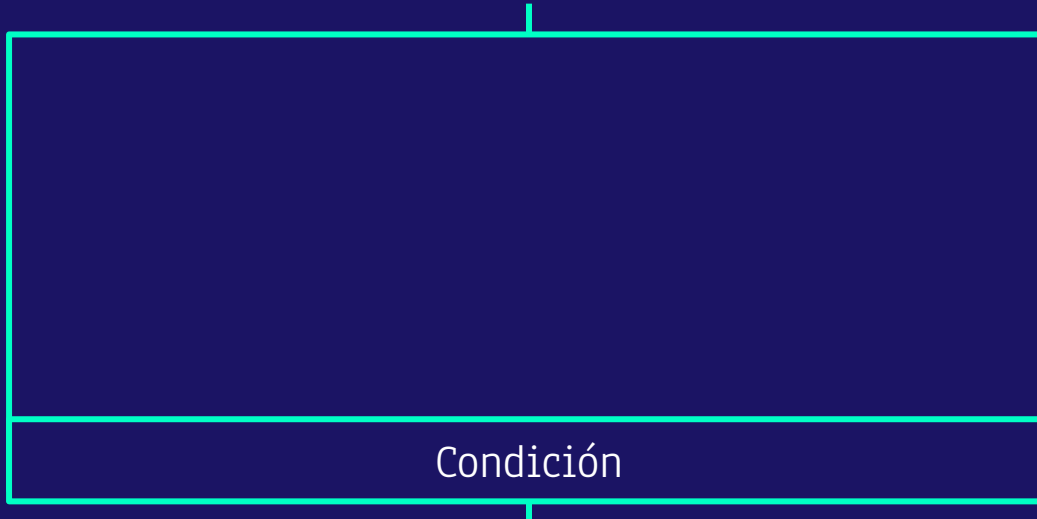
# DEL DIAGRAMA AL CÓDIGO

Otro ejemplo: Realizar un programa que solicite números y finalice cuando se ingrese un cero.



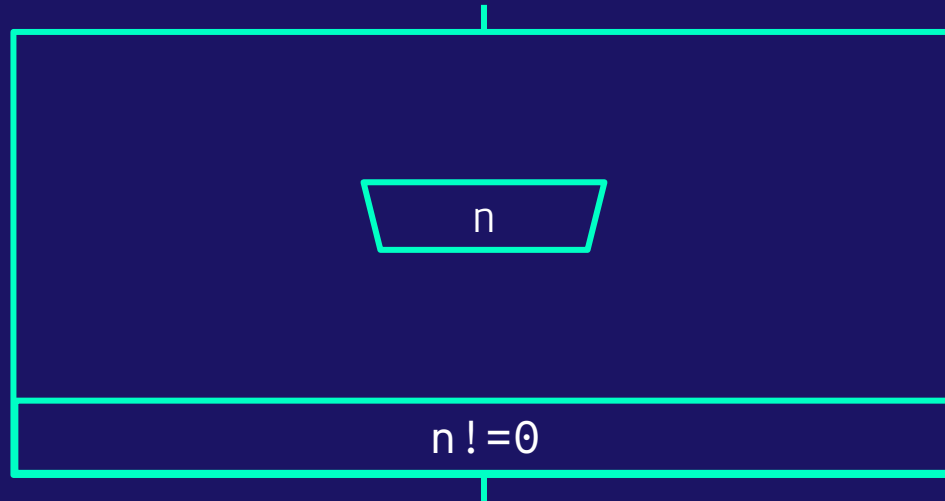
# CICLO INEXACTO – DO WHILE

Este ciclo funciona casi de la misma manera que el ciclo While tradicional. La diferencia fundamental es que la primera vuelta se ejecutará siempre, y recién para la segunda es que la condición será evaluada para determinar si continuar ciclando o concluir el bloque.



# DIAGRAMACIÓN

Otro ejemplo: Realizar un programa que solicite números y finalice cuando se ingrese un cero.



# DEL DIAGRAMA AL CÓDIGO

Otro ejemplo: Realizar un programa que solicite números y finalice cuando se ingrese un cero.

```
do  
{
```

```
    cin>>n;
```

```
}while (n!=0)
```

