

Ciclo inexacto

El ciclo inexacto permite ejecutar un conjunto de instrucciones una y otra vez mientras que una condición dada sea cierta.

```
while(condicion){  
    instruccion_1;  
    instruccion_2;  
    .  
    .  
    .  
    instruccion_N;  
}
```

Ciclo inexacto

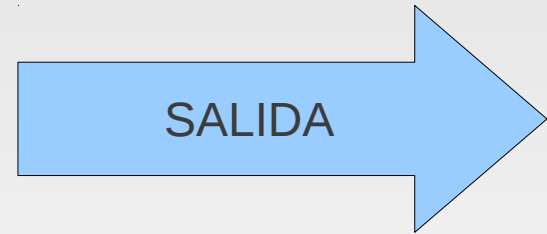
Supongamos que queremos ingresar una lista de números hasta que alguno de ellos sea primo.

Al finalizar el programa, el mismo deberá listar la cantidad de números pares e impares que tuvo la lista.

Ciclo inexacto



?



Ciclo inexacto

VARIABLES NECESARIAS

INT:

n → Para ingresar cada uno de los números

cPar → Para contar la cantidad de pares

cImpar → Para contar la cantidad de impares

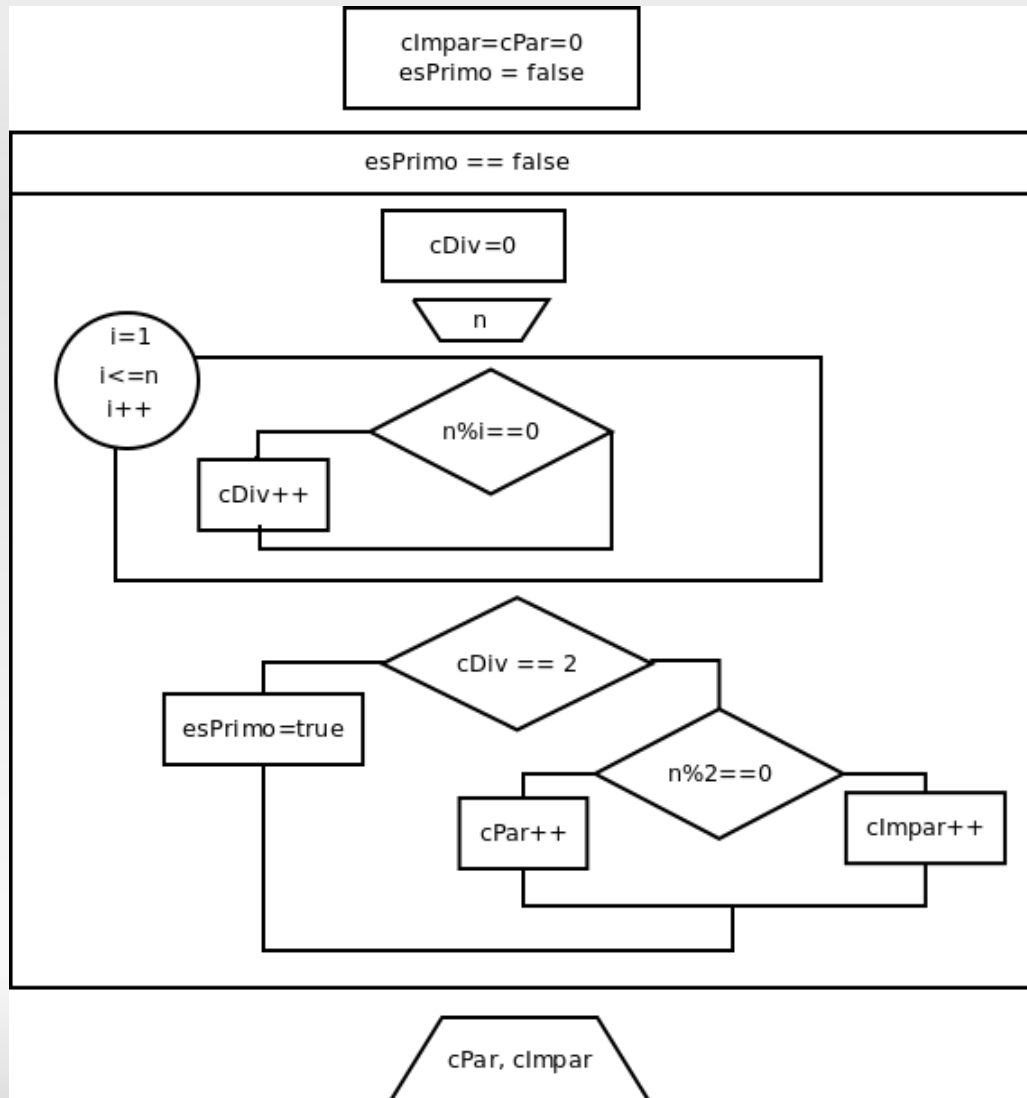
cDiv → La cantidad de divisores para determinar si es primo

i → Para el ciclo exacto necesario para calcular si el número es primo.

BOOL:

esPrimo → Bandera utilizada para indicar que se ingresó un número primo.

Ciclo inexacto



Ingresos:

$n \rightarrow 10$

$n \rightarrow 20$

$n \rightarrow 15$

$n \rightarrow 7$

Salida:

$cPar \rightarrow 2$

$cImpar \rightarrow 1$

Ciclo inexacto

OTROS EJEMPLOS:

- Se desea ingresar un número entero entre 1 y 10, en caso contrario, volver a pedir el ingreso del número hasta que se ingrese correctamente. Hay que tener en cuenta que los números correspondidos entre el intervalo $[1; 10]$ son los que el programa debe permitir, por lo tanto deberá pedir nuevamente el ingreso del número si el mismo se encuentra dentro de los intervalos $(-\infty; 1)$ ó $(10; \infty)$

```
cin >> n;  
while(condicion){  
    cin >> n;  
}
```

Ciclo inexacto

while(n>=1 && n<=10)

N	(N >= 1	&&	N <= 10)	Resultado lógico
-5	false	&&	true	false
5	true	&&	true	true
100	true	&&	false	false

while(n<1 && n>10)

N	(N < 1	&&	N > 10)	Resultado lógico
-5	true	&&	false	false
5	false	&&	false	false
100	false	&&	true	false

Ciclo inexacto

while(n>=1 || n<=10)

N	(N >= 1		N <= 10)	Resultado lógico
-5	false		true	true
5	true		true	true
100	true		false	true

while(n<1 || n>10)

N	(N < 1		N > 10)	Resultado lógico
-5	true		false	true
5	false		false	false
100	false		true	true

Ciclo inexacto

while(!(n>=1 && n<=10))

N	!	(N >= 1	&&	N <= 10)	Resultado l3gico
-5	!	false	&&	true	!false → true
5	!	true	&&	true	!true → false
100	!	true	&&	false	!false → true

Ciclo inexacto

OTROS EJEMPLOS:

- Se desea ingresar la forma de pago de una compra, la misma puede ser en efectivo → 'e' ó tarjeta de crédito → 't'. Si se ingresara un valor distinto a 'e' ó 't' se deberá pedir el ingreso de la forma de pago nuevamente. Hay que tener en claro que tanto 'e' como 't' son los valores aceptados. Por lo que si se ingresa otro valor, se deberá pedir el ingreso nuevamente.

```
cin >> fp;  
while(condicion){  
    cin >> fp;  
}
```

Ciclo inexacto

while(fp!='e' || fp!='t')

fp	(fp!='e'		fp!='t')	Resultado lógico
'h'	true		true	true
't'	true		false	true
'e'	false		true	true

while(fp!='e' && fp!='t')

N	(fp!='e'	&&	fp!='t')	Resultado lógico
'h'	true	&&	true	true
't'	true	&&	false	false
'e'	false	&&	true	false

Ciclo inexacto

while(!(fp=='e' && fp=='t'))

fp	!	(fp=='e'	&&	fp=='t')	Resultado lógico
'e'	!	true	&&	false	!false → true
't'	!	false	&&	true	!false → true
'h'	!	false	&&	false	!false → true

while(!(fp=='e' || fp=='t'))

fp	!	(fp=='e'		fp=='t')	Resultado lógico
'e'	!	true		false	!true → false
't'	!	false		true	!true → false
'h'	!	false		false	!false → true