

Tarea:

- ① Imprimir los #'s enteros del 1 al 10
- ② 90 al 100
- ③ pares del 90 al 100
- ④ enteros del 3 al 1.
- ⑤ impares del 100 al 90.
- ⑥ Imprimir los #'s enteros desde 1 hasta n , donde n se solicita al usuario.

⑦ Imprimir los #'s pares desde un rango inferior que se solicita al usuario hasta un rango superior que se solicita al usuario

⑧ Imprime la sigt. tabla en pant.

*	N	$N \times 10$	$N \times 100$	$N \times 1000$
*	1	10	100	1000
	2	20	200	2000
	3	30	300	3000

Todas las soluciones anteriores, así como los problemas asignados de tarea, tienen, en su solución, una variable contadora, que le hemos llamado i (pero que puede tener otro nombre).

¿Qué es una var. contadora?

Es una variable que se inicia en un cierto valor conveniente ($i=0$ o $i=1$, en el problema de imprimir 1, 2, 3) una sola vez antes de que un ciclo empiece, y que se incrementa o decrementa en una cierta cantidad adentro del ciclo.

PROBLEMAS DE CONTEO

Todos los problemas anteriores son problemas de conteo porque:

- 1) Tienen una var. contadora en su sol.
- 2) La var. contadora controla el ciclo (el valor de la contadora, i en nuestro caso, es lo que determina si el ciclo se repite una vez más o si se corta).

Un problema de conteo tiene, en su solución, las partes (a), (b) y (c) identificadas en las 'soluciones' anteriores:

(a) darle un valor inicial conveniente a la var. contadora, i .

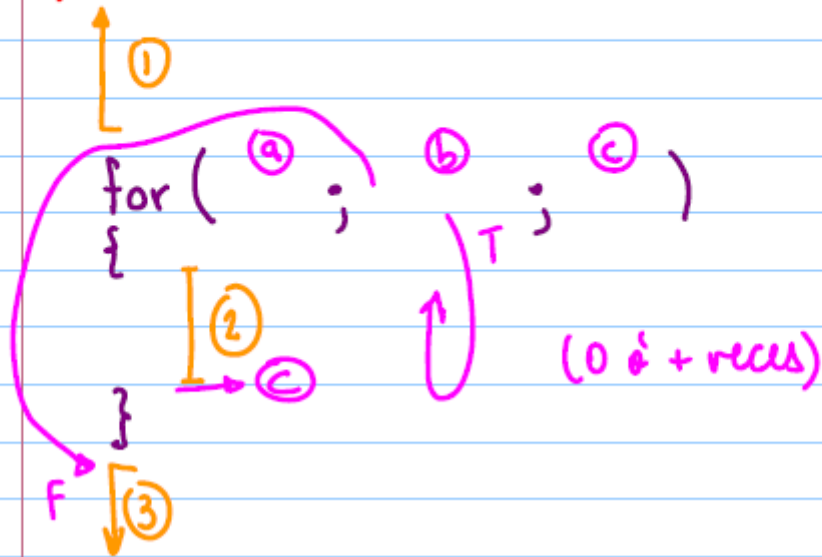
(b) una condición de fin (como en todos los problemas cíclicos) pero que depende de la var. contadora, i .

(c) el incremento o decremento, dependiendo del problema, de la var. contadora, i .

Cuando se determina que un problema es de conteo, también puedo usar la instrucción cíclica **for** para solucionarlo.

El **for** "reserva" un espacio para las partes (a), (b) y (c).

c) for



// Sol. #2 - con while
int main()
{

int i;
i = 1; (a)

while (i <= 3) (b)
{

cout << i << '\n';
i = i + 1; (c)
}

return 0;
}

// Sol. #2 - con for
1 int main()
{

2 int i;

3 for (i = 1; i <= 3; i = i + 1) (a) (b) (c)
{

4 cout << i << '\n';
}

5 return 0;
}



4	1
4	2
4	3

Ejemplos con for:

④

enteros del 3 al 1.

⑤

impares del 100 al 90.

⑥

Imprimir los #'s enteros desde 1 hasta n , donde n se solicita al usuario.

Solucion 4:

```
C:\MinGW\bin\myfor.cpp - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
[Icons]
[Taskbar]
1  /* Problema: imprimir los numeros enteros desde el 3 hasta el 1.
2  Fase 1: Analisis
3  - Crear variable contadora int i;
4  - For loop, inicializando en tres y finalizando en uno, decrementando
5  - Dentro del loop, escribir el número
6  - Fin.
7  */
8
9  //Fase 2: Programa
10 #include <iostream>
11 using namespace std;
12 int main() {
13     int i;
14     for(i = 3; i >= 1; i--){
15         cout << i << " ";
16     }
17     cout << endl;
18     return 0;
19 } //End of main
```

Fase 3: Ejecutar y validar

```
Administrator: C:\Windows\System32\cmd.exe
C:\MinGW\bin>g++ myfor.cpp -o myfor.exe
C:\MinGW\bin>myfor
3 2 1
C:\MinGW\bin>s_
```

Solucion 5:

```

1  /* Problema: imprimir los numeros impares desde el 100 hasta el 90.
2  Fase 1: Analisis
3  - Crear variable contadora int i;
4  - For loop, inicializando en 100 y finalizando en 90, decrementando
5  - Dentro del loop, verificar si es impar e imprimir
6     - par % 2 is zero, impar % 2 es uno!!!
7  - Fin.
8  */
9
10 //Fase 2: Programa
11 #include <iostream>
12 using namespace std;
13 int main() {
14     int i;
15     for(i = 100; i >= 90; i--) {
16         if(i % 2 == 1)
17             cout << i << " ";
18     }
19     cout << endl;
20     return 0;
21 } //End of main

```

```

C:\MinGW\bin>g++ myfor.cpp -o myfor.exe

C:\MinGW\bin>myfor
99 97 95 93 91

C:\MinGW\bin>

```

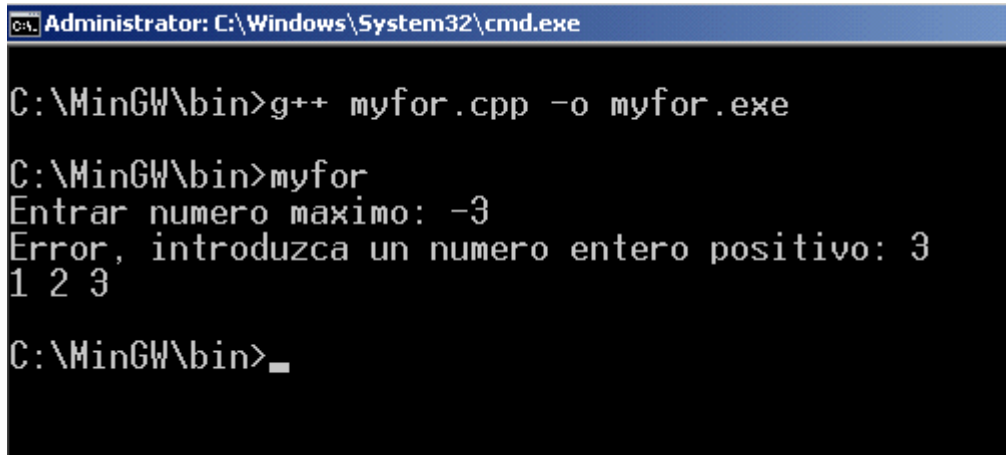
Solucion 6:

```

1  /* Problema: Imprimir numeros enteros del 1 al n donde n se pregunta al usuario.
2  Fase 1: Analisis
3  - Crear variable contadora int i;
4  - Crear variable para almacenar el numero maximo int n;
5  - Validar entrada, que sea entero y positivo
6  - For loop, inicializando en 1 y finalizando en n, incrementando
7  - Dentro del loop, imprimir usando el contador.
8  - Fin.
9  */

```

```
10
11 //Fase 2: Programa
12 #include <iostream>
13 using namespace std;
14 int main() {
15     int i, n;
16     bool isnumm;
17     cout << "Entrar numero maximo: ";
18     cin >> n;
19     .....
20 while(n < 1 ){
21     cout << "Error, introduzca un numero entero positivo: ";
22     cin >> n;
23 }
24 for(i = 1; i <= n; i++){
25     cout << i << " ";
26 }
27 cout << endl;
28     return 0;
29 }//End of main
```



The screenshot shows a Windows command prompt window titled "Administrator: C:\Windows\System32\cmd.exe". The prompt is at "C:\MinGW\bin>". The user enters "g++ myfor.cpp -o myfor.exe" to compile the program. The prompt then moves to "C:\MinGW\bin>myfor". The user enters "-3" as input. The program outputs "Entrar numero maximo: -3" and "Error, introduzca un numero entero positivo: 3". The user then enters "3" as input, and the program outputs "1 2 3". The prompt returns to "C:\MinGW\bin>".

```
Administrator: C:\Windows\System32\cmd.exe
C:\MinGW\bin>g++ myfor.cpp -o myfor.exe
C:\MinGW\bin>myfor
Entrar numero maximo: -3
Error, introduzca un numero entero positivo: 3
1 2 3
C:\MinGW\bin>_
```