|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **LENGUAJE C** | | | **No.** | **1** |
| **Asignatura:** | **METODOS NUMERICOS** | **Carrera:** | **ING. SISTEMAS** | **Duración de la práctica (Horas)** | **3** |

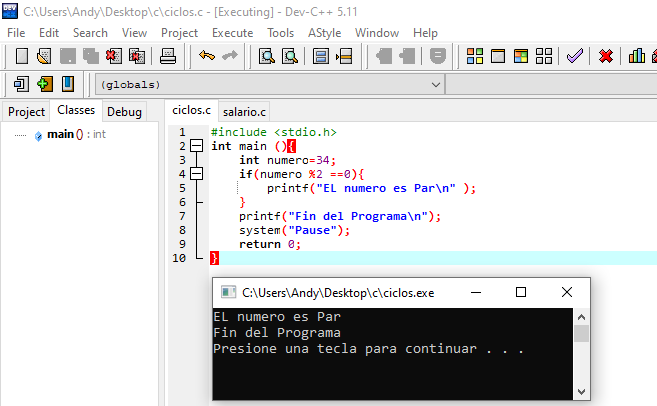
**ANDREA YOCELINE ALVAREZ SUAREZ**

1. **Competencia(s) específica(s):**
2. **Desarrollo de la Practica:**

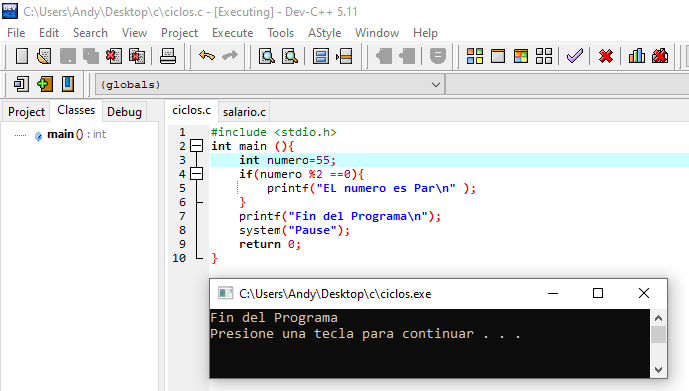
**EJERCICIOS:**

**Estructura de decisión simple**

1. Ejecuta una determinada acción cuando se cumple una cierta condición y en caso contrario sigue el orden secuencial.



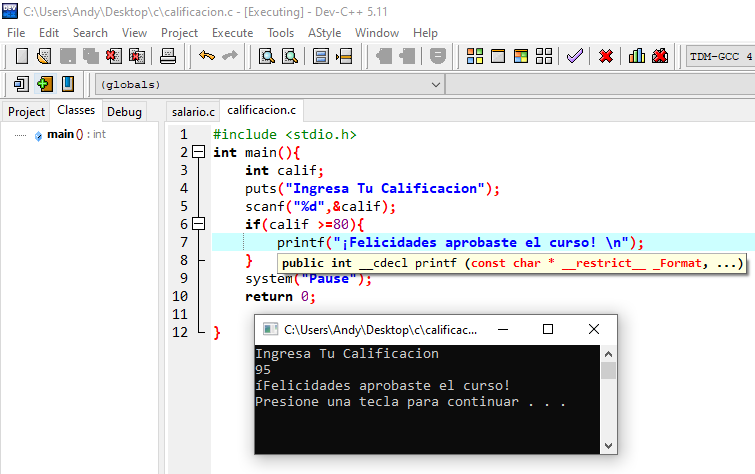
1. En este código cuando la condición no cumple se sale del ciclo y se imprime el resultado.



1. Escribe un programa que reciba de teclado la calificación del examen de un alumno e imprima el siguiente texto únicamente si la calificación fue aprobatoria.

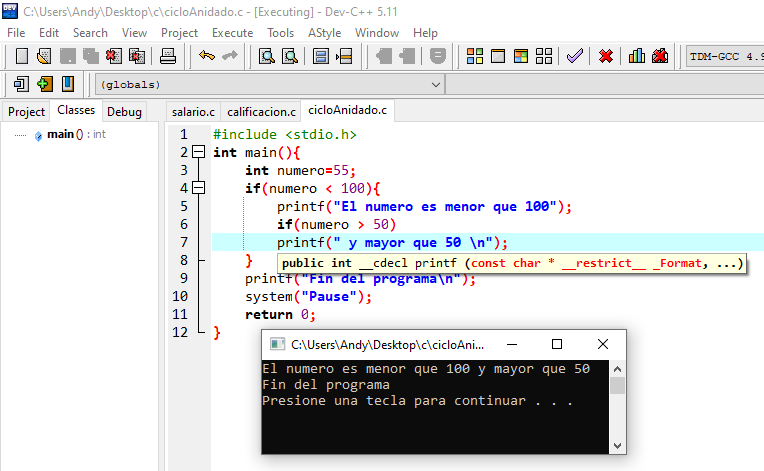
Mensaje: "¡Felicidades aprobaste el curso!"

La calificación mínima aprobatoria es: 80

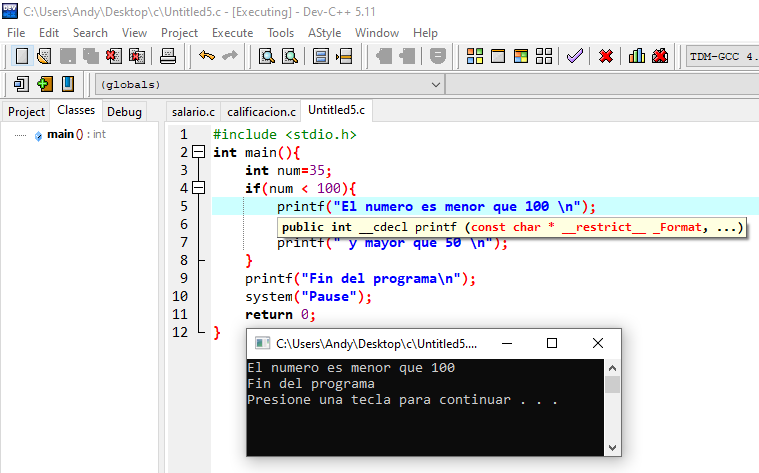


1. **Estructuras de Decisión Simple Anidadas**

Se dice que una estructura es anidada cuando existe una estructura dentro de otra.



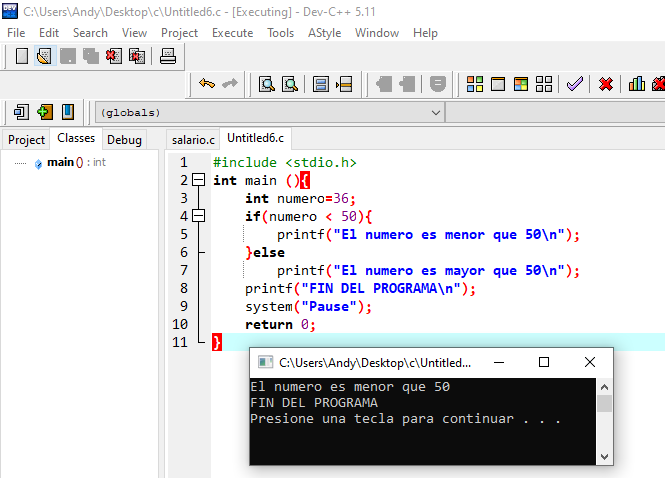
1. Depende de la cantidad que se le esté declarando a nuestra variable se verá si entra dentro de la condición, si cumple con la primera condición evaluara si puede entrar a la segunda condición si no solo se imprime la que se cumpla.

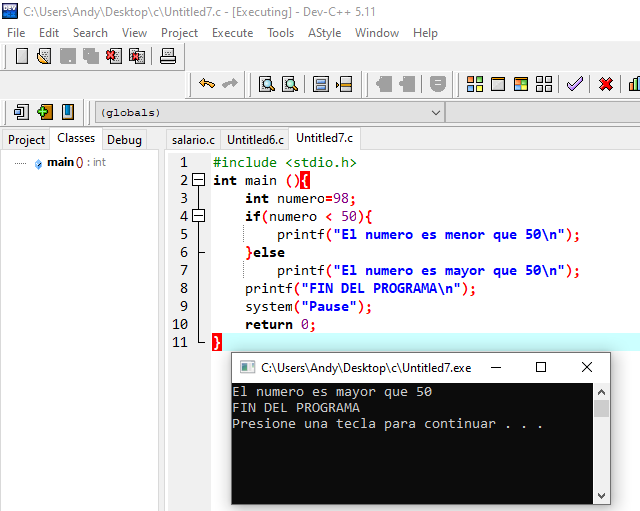


**Estructura de Decisión Doble**

Permiten ejecutar dos conjuntos de sentencias alternativos en función del resultado de la evaluación de una condición lógica.

1. Cuenta con una condición que si no cumple esta primera puede entrar a la segunda. Y si se cumple la primera ya no pasa a la segunda.





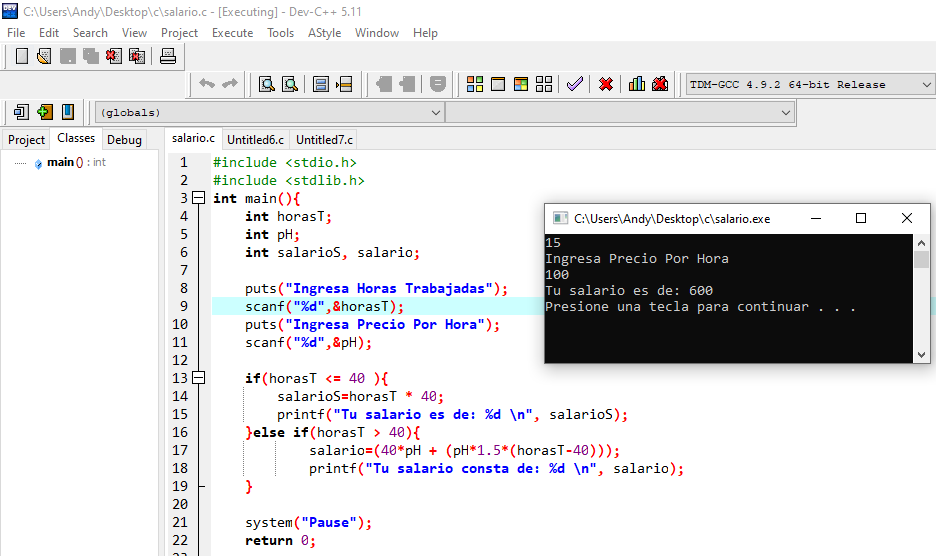
1. Calcular el salario semanal de un empleado, sabiendo que éste se calcula en base a las horas semanales trabajadas y de acuerdo a un precio especificado por hora. Si se pasa de las 40 horas semanales, las horas extras se pagarán a razón de 1.5 veces la hora ordinaria. ENTRADAS: Horas semanales trabajadas (HST) Precio por hora (PH) PROCESO:

Si las horas semanales trabajadas <= 40

Calcular salario semanal = (HST \* 40)

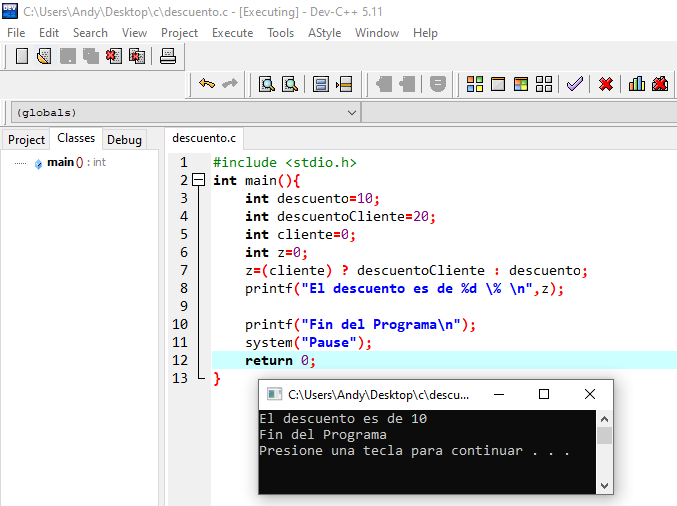
Si las horas semanales trabajadas > 40

Calcular salario semanal = (40 \* PH + (PH \* 1.5 \*(HST-40)))



**Operador condicional**

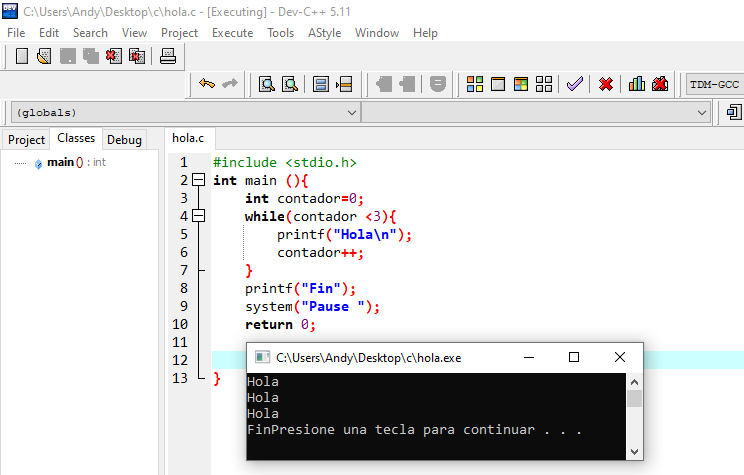
1. Es el único operador ternario de C, es decir usa tres operandos. Sintaxis: exp1? exp2: exp3



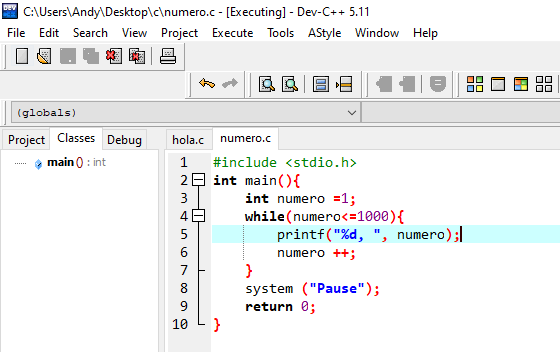
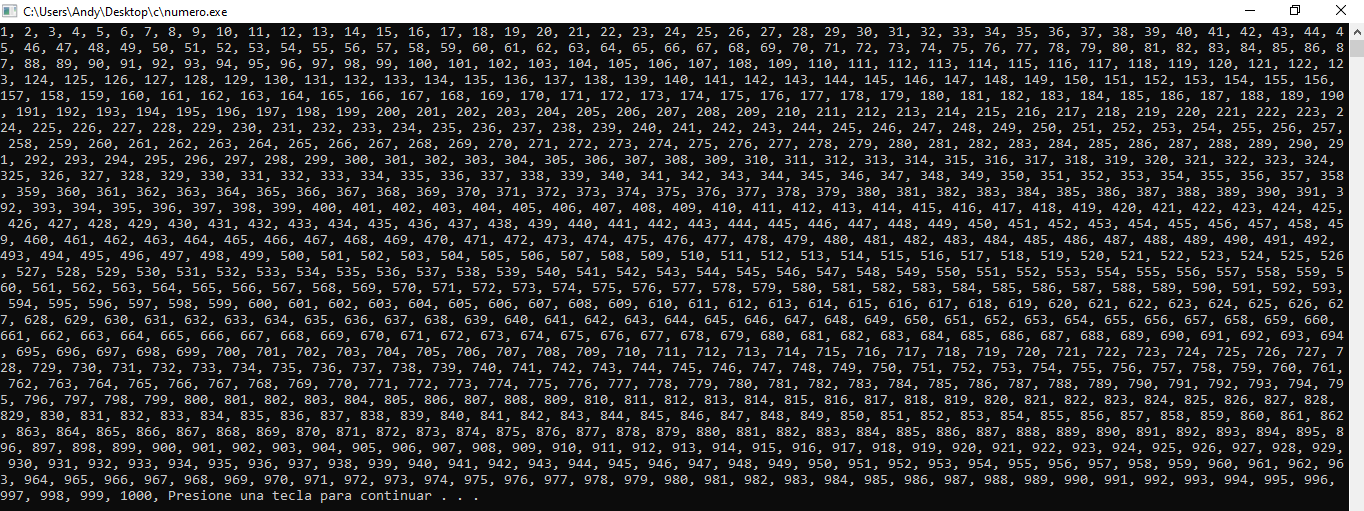
**Estructura de repetición WHILE**

Permite que un conjunto de sentencias puedan ser ejecutadas repetidamente según el estado de una expresión lógica (condición).

1. Imprimir 3 veces la palabra **HOLA**

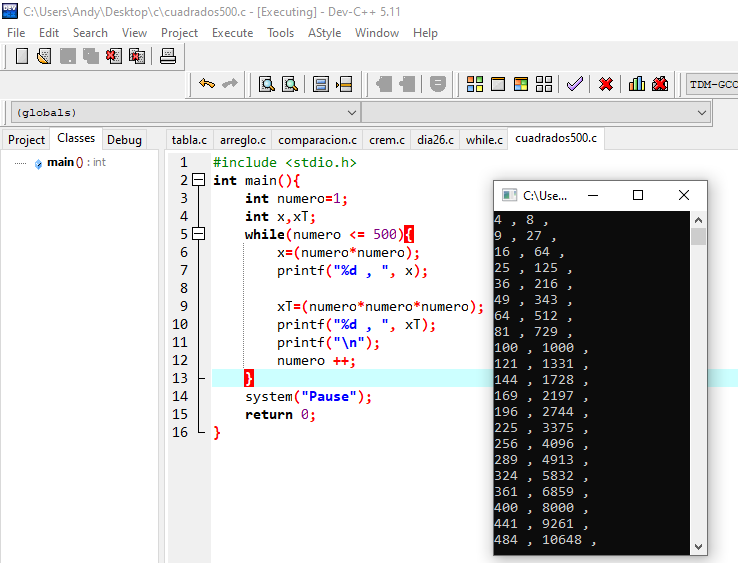


1. Imprime los números del 1 al 1000.



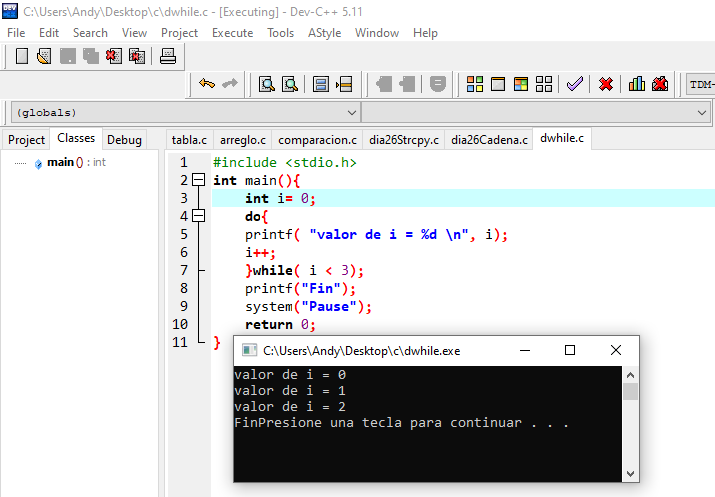
1. Imprimir los cuadrados y los cubos de los primeros quinientos números naturales.

* El cuadrado de un número se define como: x2= x\*x
* El cubo de un número se define como: x3= x\*x\*x



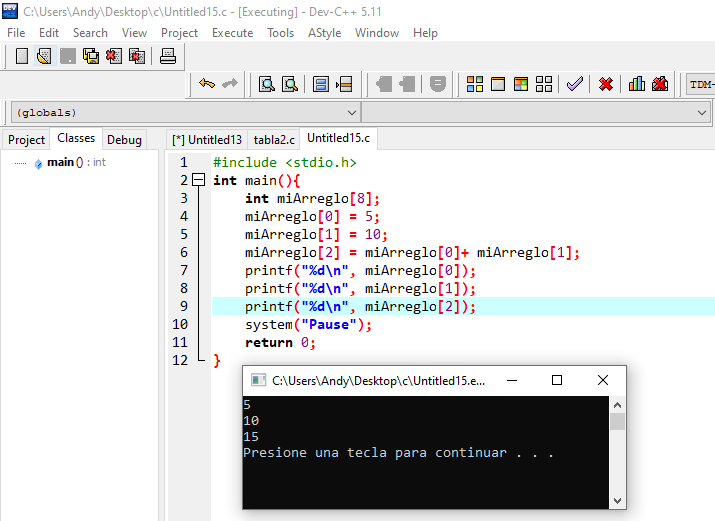
**Ciclo do -while**

1. A diferencia del while¸ esta estructura primero ejecuta el conjunto de instrucciones y después verifica que la condición se cumpla.



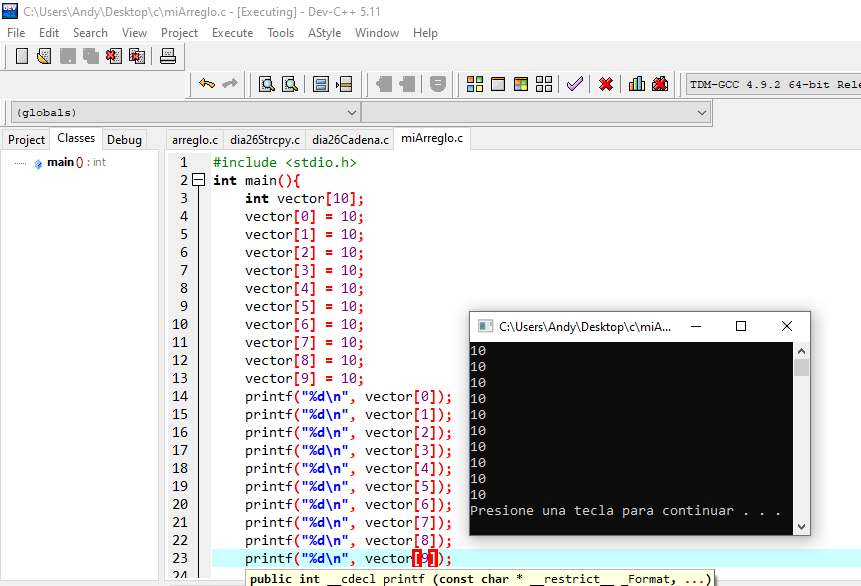
**Arreglo**

1. Variable que hace referencia a varias posiciones de memoria.

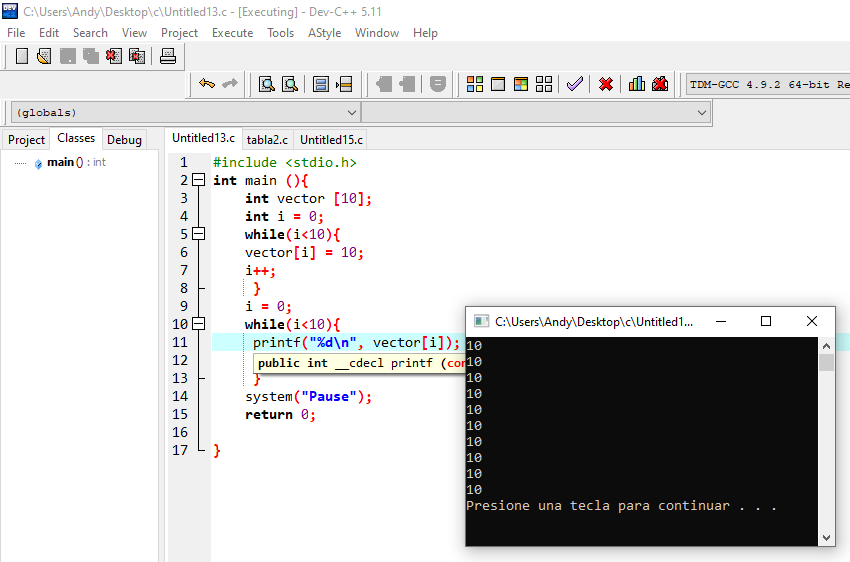


1. Crear un programa que declare un arreglo llamado "vector" de 10 posiciones.

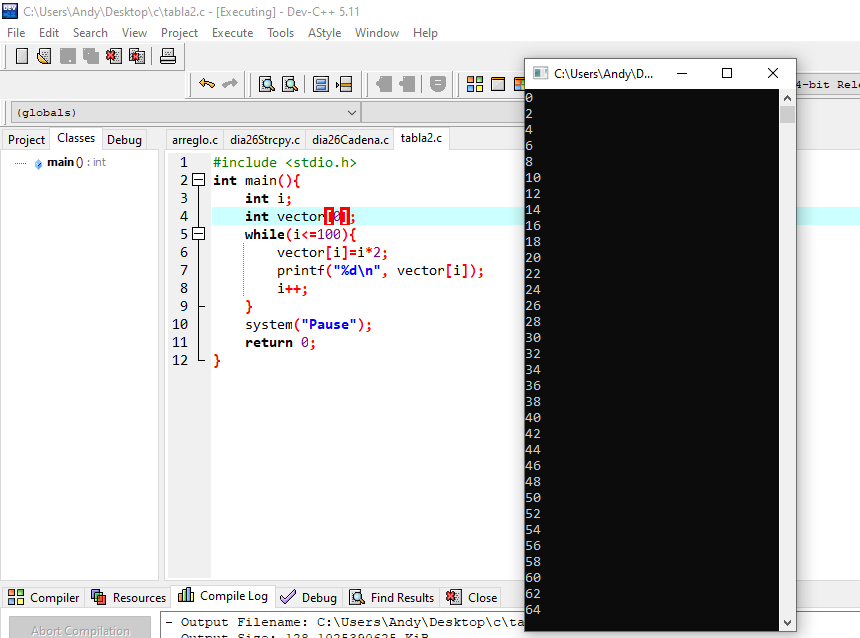
Asignar el valor de 10 a cada posición del arreglo.



1. Código para imprimir el valor 10:



1. Crear un arreglo de 100 posiciones. Llenar el arreglo con la tabla del 2.

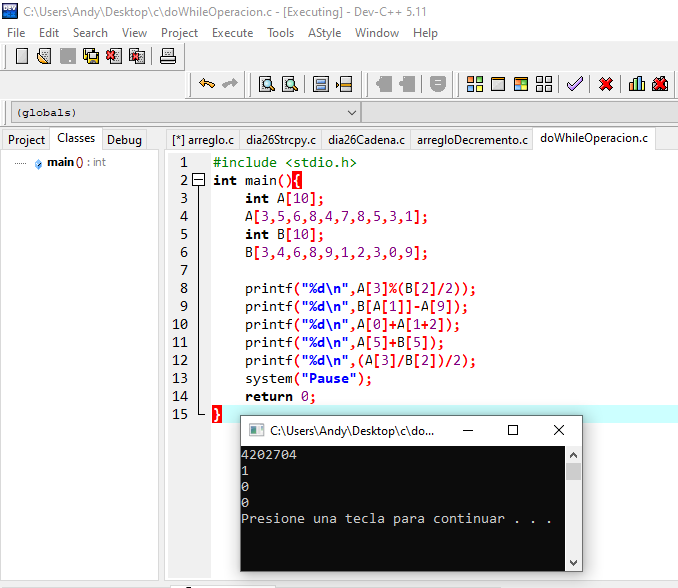


1. Crear un arreglo de 100 posiciones. Llenar el orden inverso al índice.
2. Dados A = [3, 5, 6, 8, 4, 7, 8, 5, 3, 1]

B = [3, 4, 6, 8, 9, 1, 2, 3, 0, 9]

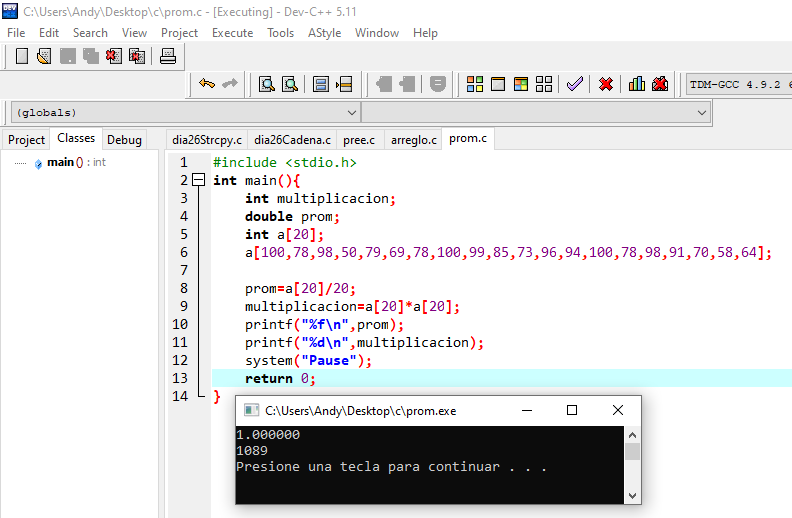
Realizar las siguientes operaciones

* A[3] mod(B[2]/2)
* B[A[1]] –A[9]
* A[0] + A[1+2]
* A[5] + B[5]
* (A[3]/B[2])/2



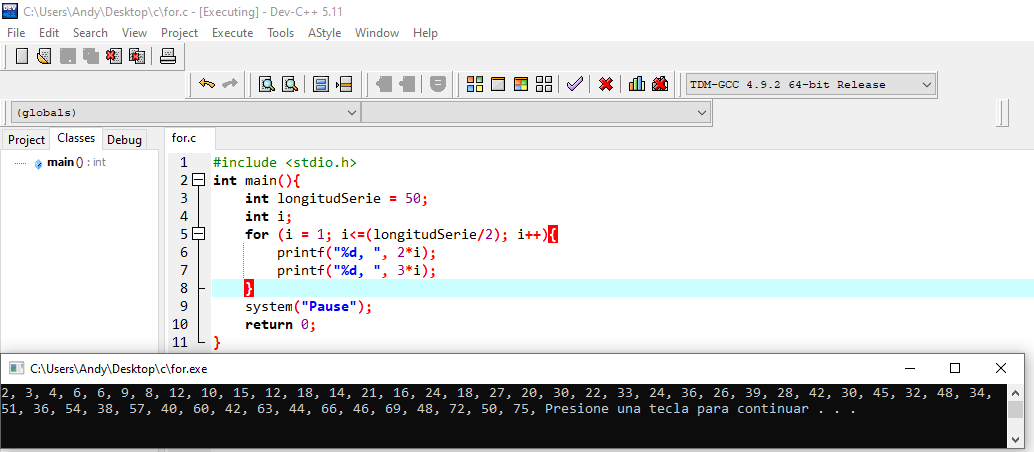
1. Crea un arreglo de 20 posiciones.

Asígnale a cada elemento un valor. Calcula el promedio de todos los elementos. Calcula la multiplicación de todos los elementos.



**Ciclo FOR**

1. Mostrar la serie: 2, 3, 4, 6, 6, 9, 8, 12, 10



1. Escribe un programa que reciba un número N del usuario y haga la suma de todos los números desde 1 hasta N.

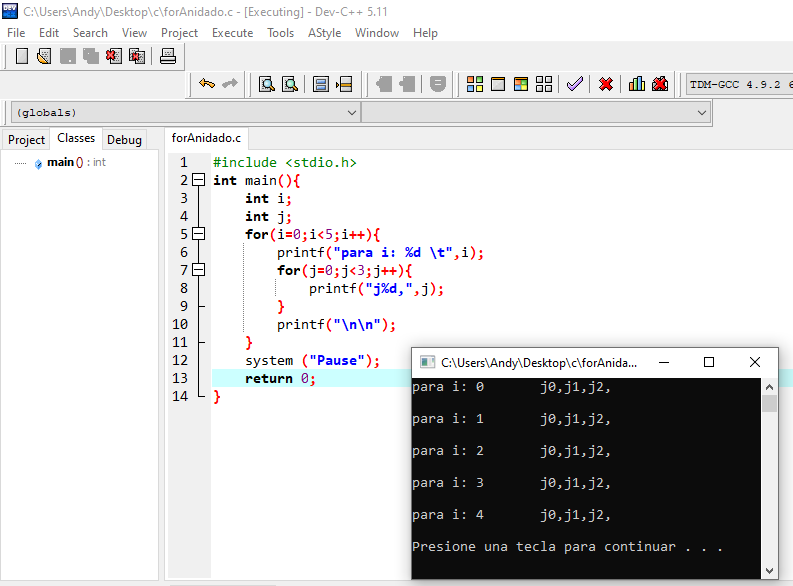
Ej. >> 5 1+2+3+4+5 = 15

1. Escriba un programa en C que utilice un ciclo para producir la siguiente tabla de valores.



**FOR Anidado**

1. Se encuentra un ciclo for dentro de otro ciclo for, se cumplen las condiciones siempre y cuando se cumpla la primera.



1. Escribe un programa que reciba un número entero N del usuario e imprima una escalinata de N pisos de asteriscos.

Asteriscos Incremento

