



Github



Linkedin



Youtube



Rust Capítulo 2. Condicionales

Ejercicios del Tema

1. Diseñar una aplicación que solicite al usuario un numero e indique si es par o impar
2. Pedir dos números enteros y decir si son iguales o no
3. Solicitar dos números distintos y mostrar cuál es el mayor
4. Realizar de nuevo el ejercicio anterior considerando el caso de que los números introducidos sean iguales
5. Implementar un programa que pida por teclado un número decimal e indique si es un número casi-cero, que son aquellos, positivos o negativos, que se acercan a 0 por menos de 1 unidad, aunque curiosamente el 0 no se considera un número casi-cero

Ejemplos de números casi-cero son el 0.3, el -0.99 o el 0.123

Y números que no se consideran casi-ceros son: el 12.3, el 0 o el -1.

6. Pedir dos números y mostrarlos ordenados de forma decreciente.
7. Pedir tres números y mostrarlos ordenados de mayor a menor.
8. Pedir los coeficientes de una ecuación de 2º Grado, y mostrar sus soluciones reales

Si no existen, debe indicarlo

Recordemos que las soluciones de una ecuación de segundo grado.

$$ax^2 + bx + c = 0 \text{ son}$$

$$x = (-b (+-) \text{ raíz}(b^2 - 4ac)) / 2a$$

9. Escribir un aplicación que indique cuántas cifras tiene un número entero introducido por teclado, que estará comprendido entre 0 y 99999.
10. Pedir un número entre 0 y 9999, y decir si es capicúa.
11. Pedir una nota de 0 a 10 y mostrarla de la forma: Insuficiente (de 0 a 4), Suficiente (5), Bien (6), Notable (7 y 8) y Sobresaliente (9 y 10).
12. Pedir el día mes y año de una fecha e indicar si la fecha es correcta. Recordamos que existen meses con 28, 30 y 31 días. No consideramos los años bisiestos.

- 13. Escribir un programa que pida un hora de la forma: hora, minutos y segundos
El programa debe mostrar la hora un segundo más tarde**

Un ejemplo:

hora actual [10:41:59] hora actual + 1 segundo [10:42:00].

- 14. Crear una aplicación que solicite al usuario una fecha (día, mes y año) y muestre la fecha correspondiente al día siguiente.**
- 15. Idear un programa que solicite al usuario un número comprendido entre 1 y 7, correspondiente a un día de la semana. Se debe mostrar el nombre del día de la semana al que corresponde. Por ejemplo, el número 1 corresponde a “lunes” y el 6 a “sábado”.**
- 16. Solicitar un número comprendido entre 1 y 99. El programa debe mostrarlo escrito. Por ejemplo, para 56 mostrar “cincuenta y seis”.**



Github



Linkedin



Youtube

