

# Módulo Teórico-Práctico

# Entrega

Módulo
Programación de Computadores
Tipo de entrega
Taller de ejercicios

## Nota

Tenga en cuenta que el tutor le indicará qué herramienta requiere y qué estrategia deberá desarrollar para evidenciar su participación individual en un trabajo colaborativo.



## INSTRUCCIONES PARA REALIZAR LA ENTREGA

Para realizar este taller de ejercicios, es fundamental que realices una lectura previa exhaustiva de todos los documentos y revises detenidamente todos los recursos didácticos proporcionados en los Escenarios hasta la fecha de cada una de las entregas. El tutor estará disponible para resolver cualquier pregunta o inquietud que pueda surgir, y podrás comunicarte con él a través de los diferentes canales de comunicación disponibles tanto de manera sincrónica como asincrónica en el aula.

El primer paso que debes dar para comenzar con este taller es formar un equipo de trabajo compuesto por tres o cuatro personas. Es importante que todos los miembros del equipo sean parte del mismo curso, es decir, pertenezcan al grupo tutorizado por el mismo tutor. Solo uno de los integrantes del equipo deberá cargar el trabajo en el aula virtual, pero asegúrense de incluir los nombres completos de todos los miembros del equipo en la entrega. No hacerlo resultará en que los estudiantes no mencionados no recibirán una calificación por su participación en el proyecto.

Recuerda que el plazo y la forma de entrega de esta actividad están claramente especificados en el aula virtual. Cualquier entrega realizada después de la fecha límite o a través de un medio diferente al indicado será considerada inválida.

# ENTREGA PREVIA 1

## SEMANA 3

Desarrolle uno de los siguientes ejercicios propuestos. El ejercicio que debe resolver cada equipo será asignado por el tutor del módulo. La entrega debe consistir en un archivo java con la solución del ejercicio.

### Ejercicios

#### 1. Siglo

En la clase de historia de Juan con frecuencia se habla de los siglos en que ocurrieron ciertos hechos. Esto hace que a Juan se le dificulte un poco ubicar los hechos en una línea de tiempo. Su tarea es escribir un programa java que ayude a Juan.

- A. Escriba un método java llamado `siglo`, el cual recibe como entrada el número de un año (positivo), y retorna el número del siglo al que pertenece el año.
- B. Escriba un método java llamado `primer_anho`, el cual recibe como entrada el número de un siglo (positivo), y retorna el número del primer año de dicho siglo.
- C. Complete el programa de forma que lea del teclado un valor `n` correspondiente a un año, e imprima en pantalla el número del siglo al que pertenece el año `n` y el primer año de dicho siglo.

El programa debe usar los métodos creados en los literales a y b.

#### 2. Conversión de moneda

Juan está planificando un viaje a un país extranjero y necesita calcular la cantidad de dinero que tendrá en la moneda local en diferentes momentos del día debido a las fluctuaciones en el tipo de cambio.

- A. Escribe un método en Java llamado "conversionA las8AM" que reciba como entrada la cantidad de dinero en moneda local (P), y retorne la cantidad de dinero en moneda local a las 8:00 a.m., antes de cualquier cambio en el tipo de cambio.
- B. Escribe un método en Java llamado "conversionAlMediodía" que reciba como entrada la cantidad de dinero en moneda local (P), y retorne la cantidad de dinero en moneda local al mediodía, después de aplicar una disminución del 10% en el tipo de cambio.
- C. Escribe un programa principal en Java que solicite al usuario ingresar la cantidad de dinero en moneda local y luego muestre en pantalla la cantidad de dinero en moneda local a las 8:00 a.m. y al mediodía.

El programa debe utilizar los métodos creados en los literales A y B.

### 3. Conversión de medidas de longitud

Escriba un programa que pida al usuario el número de metros de un objeto, y muestre en pantalla las conversiones de dicha cantidad a pies, pulgadas y centímetros.

## ENTREGA PREVIA 2 SEMANA 5

Desarrolle uno de los siguientes ejercicios propuestos. El ejercicio que debe resolver cada equipo será asignado por el tutor del módulo. La entrega debe consistir en un archivo java con la solución del ejercicio.

### Ejercicios

1. Adivina el personaje

Escriba un programa que pida al usuario elegir uno de los personajes de la lista de abajo (sin informar de su elección a su programa). Posteriormente, su programa debe hacer al usuario un conjunto de preguntas (máximo 5) de tipo Sí/No hasta determinar el personaje escogido por el usuario. Personajes: Radamel Falcao García, Goku, Michael Jordan, Eminem, Darth Vader, Adam Sandler, Bruce Wayne, Tin Tin, Ayudante de Santa, Joe Biden, José Saramago, Günter Grass y Kim Jong Un.

## 2. Caballos

- A. Escriba un método java que reciba como entrada cuatro valores enteros, fila\_cab (fila caballo), col\_cab (columna caballo), fila\_rey y col\_rey; y decida si, en un tablero de ajedrez, un caballo en la posición (fila\_cab, col\_cab) ataca a un rey en la posición (fila\_rey, col\_rey).
- B. Escriba un programa que pida al usuario las posiciones de un caballo y un rey en un tablero de ajedrez, y muestre en consola si el rey es atacado por el caballo. El programa debe usar el método creado en el literal a.

## 3. Suma de números primos

Escribe un programa que reciba como entrada un número entero positivo, n, y calcule la suma de todos los números primos menores o iguales que n. Luego, imprime el resultado de esta suma.

# ENTREGA FINAL SEMANA 7

Desarrolle uno de los siguientes ejercicios propuestos. El ejercicio que debe resolver cada equipo será asignado por el tutor del módulo. La entrega debe consistir en un archivo java con la solución del ejercicio.

## Ejercicios

### 1. Mediana

Investigue como se calcula la mediana y escriba un programa que reciba como entrada un arreglo de reales  $x$  e imprima el elemento mediano de  $x$ .

### 2. Contraseña segura

Una contraseña se considera segura si su longitud es mayor o igual a 10, tiene al menos una letra (A-Z, a-z), tiene al menos un dígito (0-9), y tiene al menos un símbolo diferente a letras o dígitos. Escriba un programa que reciba como entrada una cadena y decida si esta corresponde a una contraseña segura.

### 3. Cadenas amigas

Se dice que dos cadenas  $S$  y  $T$  son amigas si existen dos cadenas no vacías  $u$  y  $v$  tales que  $S = uv$  y  $T = vu$ . Por ejemplo, “tokyo” y “kyoto” son amigas, siendo  $u = \text{“to”}$  y  $v = \text{“kyo”}$ . Escriba un programa java que reciba como entrada dos cadenas  $S$  y  $T$ , e imprima si  $S$  y  $T$  son amigas.

## Criterios de evaluación

Es importante que los programas estén bien estructurados, sigan las mejores prácticas de programación y produzcan resultados correctos y precisos en todas las situaciones. Además, se debe prestar atención a la claridad y legibilidad del código, así como a la gestión de errores.

### Correctitud del Código:

- El código debe ser funcional y realizar la tarea especificada en el enunciado sin errores.
- Debe manejar de manera adecuada los valores de entrada y los posibles casos límite.

- El código debe estar bien estructurado, utilizando nombres de variables y funciones descriptivos.
- El código debe ejecutarse de manera eficiente y sin excesivo consumo de recursos, especialmente en ejercicios que involucran cálculos repetitivos.

## Legibilidad y Comentarios:

- Deben incluirse comentarios apropiados que expliquen la lógica del programa, especialmente en secciones complicadas o algoritmos importantes.

## Manejo de Errores:

- El programa debe manejar errores de entrada del usuario de manera adecuada, proporcionando mensajes de error claros y sugerencias para corregirlos.

## Precisión de los Cálculos:

- Los cálculos matemáticos deben realizarse con precisión y exactitud, especialmente en ejercicios que involucran conversiones numéricas.

## Cumplimiento de Requisitos:

- El programa debe cumplir con todos los requisitos especificados en el enunciado del ejercicio.
- Deben implementarse todos los métodos y funciones requeridos.

## Interacción con el Usuario:

- Cuando se requiera, el programa debe interactuar de manera clara y amigable con el usuario, proporcionando instrucciones claras y solicitando la información necesaria.



Apreciado estudiante, haz tus entregas como **DAVID EL RINOCERONTE**, quien con transparencia y calidad, presenta siempre sus trabajos puntual.

Antes de subir tus archivos a la plataforma, lee detenidamente las siguientes indicaciones y minimiza inconvenientes:

1. Por tu seguridad y la seguridad de tus compañeros, nunca incluyas datos personales.
2. Antes de la fecha de cierre, podrás guardar el archivo en caso que sea necesario borrarlo o cambiarlo. Si lo envías para calificar, no podrás realizar ningún cambio sobre el mismo.
3. Cuando hayas cargado tu trabajo correctamente, podrás ver el nombre del archivo al lado derecho de tu pantalla.
4. Recuerda: debes cargar el archivo por lo menos dos horas antes de la hora de cierre. Evita problemas de saturación en el servidor.
5. Asegúrate de tener buena conexión a internet, cierra cualquier programa que pueda consumir el ancho de banda y no utilices internet móvil.
6. El uso de teléfonos inteligentes o tabletas está permitido únicamente para la visualización de tu material. No la uses para la presentación de actividades evaluativas, ni para cargar archivos de entregas grupales.
7. Si presentas algún inconveniente de la plataforma al momento de cargar el archivo, te recomendamos crear un caso adjuntando imágenes de soporte, donde se evidencie nombre de la actividad y/o URL respectiva, el error, la fecha y hora en que ocurrió.

¡Confiamos en que sigas, paso a paso, en el camino hacia la excelencia académica!  
¿Das tu palabra de que realizarás esta actividad asumiendo de corazón nuestro

**PACTO DE HONOR?** 