

## Indicaciones generales

### Desarrollo:

- Para el examen es obligatorio que se conecten por medio de Zoom con un dispositivo que tenga una cámara y audio (tal como indica la Resolución de la Vicerrectoría de Docencia No. VD-11502-2020). Además, se debe compartir pantalla durante la realización de la prueba (salvo casos previamente definidos con él o la docente). Debe mantenerse en Zoom durante toda la duración del examen. Se tomará asistencia.
- El examen es estrictamente individual. Está prohibido interactuar con cualquier otra persona que no sea él o la docente. En caso de sospechar lo contrario, se aplicará el debido proceso estipulado en el Reglamento de orden y disciplina de los estudiantes de la Universidad de Costa Rica, el cual plantea sanciones de suspensión hasta por seis años.
- Debe hacer uso de buenas prácticas para la programación orientada a objetos, por ejemplo: nombres significativos, indentación adecuada, clases con responsabilidades separadas, breve documentación de métodos y clases, y convenciones de nombres.
- En este examen se permite la reutilización de código propio o aquel provisto en el curso (pero debe acreditarse la autoría). También se tiene derecho a consultar material escrito en la internet. Todo código no realizado por él o la estudiante debe encontrarse debidamente referenciado.
- La comunicación entre docente y estudiante durante la realización de la prueba es por medio de Zoom. Para solicitar a él o la docente, de clic en el botón de “Pedir ayuda”.
- Cualquier imprevisto técnico durante el examen deberá comunicarlo inmediatamente a él o la docente por medio de un mensaje privado en Telegram.
- Si normalmente utiliza el internet de la casa con señal wifi y se va la electricidad, pero posee internet en el celular comuníquese inmediatamente a él o la docente por un mensaje privado de Telegram.

### Entrega:

- Cuenta con 3 horas para solucionar el examen.
- Una vez transcurridas las 3 horas tendrá 5 minutos adicionales para subir un archivo .zip (cuyo nombre de archivo sea el nombre y el apellido suyo) a Mediación Virtual. Debe subir dentro del archivo .zip todos los archivos de código fuente (.java). El enlace de mediación virtual se cerrará en punto. Note que estos 5 minutos adicionales se otorgan para solventar cualquier problema técnico para la entrega del archivo. Si desean usar Github (opcional), pueden subir el código del examen en un repositorio privado y agregar a la docente al repositorio. Si hace la entrega con Git debe subir a mediación un archivo (.pdf) que documente el enlace del repositorio y el hash (identificación del commit). **Solo se debe entregar el .zip con el código o el .pdf con los datos del repositorio, no ambos.**
- Debe subir el examen en Mediación Virtual al enlace correspondiente. En caso de problemas con la plataforma de Mediación Virtual debe mandar el examen por correo (sivana.hamer@ucr.ac.cr). En caso excepcional que fallen ambos medios, debe enviar a él o la docente por medio de un mensaje privado en Telegram. Se debe subir el examen siempre durante el período de entrega del examen.
- Si realiza el examen en una computadora de escritorio que depende de energía eléctrica y sucede un fallo en el suministro, debe continuar su examen en papel.

**Recomendaciones:**

- Antes de comenzar a implementar código, preocúpense por tener total claridad sobre cada pregunta y solo responda lo que se le pida, ya que responder otros aspectos no permite otorgarle puntuación alguna. Ejemplo: si no se les pide leer, usen constantes, si no se les pide usar el patrón MVC, no tienen por qué hacerlo.
- No se olviden de guardar el código antes de subirlo (en Visual Studio Code es: ctrl + s).
- Si posee una computadora portátil, es una buena alternativa que la utilice para realizar el examen, asegurándose de que la batería está cargada al 100 %, para que en caso de una falla eléctrica pueda continuar el examen. Tengan disponible siempre el cargador.
- Garanticen que su teléfono está con la batería cargada o tiene disponible el cargador.
- Eviten distracciones. Vayan al baño y a comer antes del examen. Intenten encontrarse en una zona con silencio. Avisen a sus familiares que van a realizar un examen.

**Pregunta 1 (60 %)**

Se desea crear un programa que, usando el paradigma orientado a objetos, sea capaz de manejar (crear, consultar, mostrar) distintas monedas y una billetera digital, con capacidad para manejar simultáneamente varias criptomonedas.

Las criptomonedas poseen un nombre, una popularidad y una tendencia representada por un arreglo de valores de la moneda del último minuto (cambios en la moneda). Se guarda un valor de tendencia por cada segundo. Los valores de la tendencia estarán generados aleatoriamente como valores de tipo real, que se guardarán en el arreglo para ello.

Se espera en un inicio, que el programa sea capaz de manejar distintas criptomonedas.

En un objeto billetera, defina y agregue las siguientes criptomonedas, donde se incluye como primer valor la popularidad de la misma, por ejemplo:

1. Bitcoin, con un rango de valores aleatorios entre 0,0 y 3,0.
2. Ethereum, con un rango de valores aleatorios entre -5,5 y 5,0.
3. Doge, con un rango de valores aleatorios entre -8,0 y 8,0.

Adicionalmente, cree su moneda propia, siendo esta la cuarta más popular. Defínale un nombre y el rango a usar para generar los valores.

Como parte de su implementación debe poderse determinar mediante un método: cuál de todas las criptomonedas posee un mejor rendimiento promedio (sumar sus tendencias de valores y dividir esta suma entre la cantidad total de valores).

**Desglose de evaluación:**

- Manejo de criptomoneda (25 %)
  - Crear (10 %)
  - Consultar (10 %)
  - Mostrar (5 %)
- Manejo de billetera (35 %)
  - Crear (5 %)
  - Medir rendimiento (20 %)
  - Controlador o main (10 %)

## Pregunta 2 (40 %)

Dada una matriz  $m$ , con dimensiones  $f * c$ , instanciada con valores aleatorios entre 0 y 99, por ejemplo:

43	85	53	20	25
8	79	90	29	94
38	25	24	94	93
39	84	93	52	20

Haga un método que permita crear  $n$  categorías para contabilizar rangos numéricos de dichos valores, de manera que si se tuviera por ejemplo un valor  $n = 3$  y la matriz dada anteriormente, se genere una matriz como parte del proceso de categorizar. Dicha matriz incluye como primera fila el rango de la categoría y como segunda fila la cantidad de valores que se hallan en ese rango. Por ejemplo, considerando  $n = 3$  se podría ver similar a la siguiente:

Cantidad de categorías deseadas?3		
Valores distribuidos en 3 categorías		
[0-33[	[33-66[	[66-99[
7	5	8

### Desglose de evaluación:

- Generación de matriz con valores aleatorios (5 %)
- Clasificación de valores en  $n$  categorías (25 %)
- Mostrar datos, tanto los originales, como el contador de casos por categoría (10 %)