

# Programación Avanzada

## Parcial 1

Departamento de Ingeniería de sistemas

15 de marzo 2024

En el reglamento de estudiantes, en el numeral 116 se indica que: "En la Universidad Javeriana la función disciplinaria tiene como finalidad el fomento de la honestidad, la buena fe y el respeto entre los miembros de la Comunidad Educativa y entre éstos y la Universidad, así como también la defensa y conservación de sus bienes y derechos." De acuerdo con el numeral 123 del reglamento de estudiantes, constituye una falta grave "el fraude en actividades, trabajos y evaluaciones académicos y la posesión o utilización de material no autorizado en los mismos". Por lo tanto, todos los casos de fraude identificados serán reportados.

No se pueden utilizar dispositivos electrónicos (Computadores, Celulares, Tablets, ...).

### 1. (20 puntos) Conceptos

Marque la respuesta correcta, y sólo la respuesta correcta, a las siguientes preguntas:

- Un apuntador es una variable que solamente puede contener una dirección de memoria
  - ☐ Falso
  - ☐ Verdadero
- Hay dos formas de enviar parámetros a una función, que se conocen como paso de parámetros por valor y paso de parámetros por referencia
  - ☐ Falso
  - ☐ Verdadero
- ¿Cuál es el propósito principal de utilizar estructuras (struct) en C++?
  - ☐ Para crear bibliotecas de funciones
  - ☐ Para agrupar variables relacionadas bajo un mismo tipo
  - ☐ Para implementar bucles
  - ☐ Para manejar excepciones
- ¿Qué operador se utiliza en C++ para asignar memoria dinámicamente?
  - ☐ sizeof
  - ☐ &
  - ☐ \*
  - ☐ new
- ¿Cómo se libera la memoria asignada a un arreglo dinámico en C++?
  - ☐ delete
  - ☐ delete[]
  - ☐ free()
  - ☐ remove()
- Cuando se incrementa un apuntador que apunta a un arreglo de enteros, ¿cuántos bytes se incrementa el apuntador?
  - ☐ 1
  - ☐ 2
  - ☐ El tamaño de un int en la plataforma
  - ☐ 8
- ¿Cuál es el caracter nulo que indica el final de una cadena de caracteres en C++?
  - ☐ \n
  - ☐ \0
  - ☐ EOF
  - ☐ NULL
- ¿Qué modo de apertura de archivo se debe utilizar para añadir datos al final de un archivo binario en C++?
  - ☐ ios::in
  - ☐ ios::out
  - ☐ ios::app | ios::binary
  - ☐ ios::trunc | ios::binary
- ¿Cómo se lee una línea completa de texto de un archivo en C++?
  - ☐ read()
  - ☐ getline(cin, variable)
  - ☐ getline(archivo, variable)
  - ☐ archivo > variable
- ¿Cuál de las siguientes opciones inicializa correctamente un arreglo dinámico de 10 float a 0.0?
  - ☐ float\* arr = new float[10]();
  - ☐ float\* arr = new float[10]{0.0};
  - ☐ A y B son correctas
  - ☐ Ninguna de las anteriores

## 2. (80 puntos) Hellbank

¡Saludos, valiente programador! Te damos la bienvenida al desafío definitivo propuesto por HellBank, el banco de las profundidades, donde las riquezas y secretos del inframundo esperan ser gestionados con tu talento. Este banco único en su especie te invita a desarrollar una herramienta crucial para procesar archivos `.hbt` (Hell Bank Transactions). Estos archivos, esenciales para el funcionamiento del banco, son un compendio binario de las transacciones realizadas en cuentas específicas y requieren de tu habilidad para ser decodificados.

Un archivo `.hbt` representa las transacciones de una única cuenta y organiza su información en el siguiente formato binario:

- Nombre del Cliente (20 bytes): Codifica el nombre del cliente en 20 caracteres. Si el nombre es más corto, se completa con espacios.
- Número de Cuenta (10 bytes): Almacena el número de la cuenta bancaria como una cadena de 10 dígitos numéricos.
- Tipo de Cuenta (1 byte): Indica el tipo de cuenta: 'A' corresponde a una cuenta de ahorros, mientras que 'C' identifica una cuenta corriente.
- Número de Transacciones (1 byte): Especifica la cantidad de transacciones que se detallan a continuación en el archivo.
- Transacciones (variable): Cada transacción se inicia con un byte que marca el tipo ('C' para crédito, 'D' para débito), seguido de 8 bytes que representan el monto de la transacción en centavos, almacenado como un valor `long`.

Desarrollarás un programa en C++ que pueda leer estos archivos `.hbt`, procesando la información para calcular y mostrar:

- Nombre del Cliente.
- Número de Cuenta.
- Tipo de Cuenta.
- Saldo Total en la Cuenta (sumando créditos y restando débitos).
- Totales de Créditos y Débitos.
- Número de Transacciones Procesadas.
- Basado en esta idea, implemente los siguientes puntos

Para esto:

1. (5 puntos) Defina una estructura que represente una transacción según la especificación del archivo `.hbt` mencionado previamente. Esta estructura sólo debe contener el tipo (crédito o débito) y el monto.
2. (5 puntos) Defina una estructura que represente la información de un cliente según la especificación del archivo `.hbt` mencionado previamente. Esta estructura debe incluir el nombre del cliente, el número de cuenta, el tipo de cuenta, el número de transacciones de la cuenta, y un arreglo dinámico de transacciones usando la estructura del punto anterior.
3. (35 puntos) Desarrolle una función que lea un archivo binario `.hbt` y retorne una estructura como la definida en el punto anterior con toda la información del archivo.
4. (20 puntos) Desarrolle una función que reciba como parámetros un arreglo dinámico de transacciones, el tamaño del arreglo, y un tipo de transacción (C o D). La función debe retornar la sumatoria de los montos de ese tipo de transacción.
5. (5 puntos) Desarrolle una función que reciba como parámetros un arreglo dinámico de transacciones y el tamaño del arreglo, y retorne el saldo de la cuenta. El saldo de la cuenta se calcula a partir de los créditos y débitos sobre la cuenta asumiendo un saldo inicial de 0 centavos.
6. (10 puntos) Desarrolle una función que reciba como parámetro una estructura como la definida en el punto 2 y la ruta a un archivo de texto. Esta función debe escribir la información de la cuenta en el archivo, como se muestra en el siguiente ejemplo:

Cliente: Juan Arroyo  
Cuenta: 1234567890  
Tipo de Cuenta: Ahorros  
Total en Cuenta: X pesos  
Total Créditos: Y pesos  
Total Débitos: Z pesos  
Número de Transacciones en Archivo: 10

¡Acepta el reto de HellBank! Tu habilidad para descifrar y gestionar estas transacciones binarias es crucial para mantener el flujo de las riquezas en el inframundo. ¡Buena suerte!

(Restricciones: No se puede usar el tipo de datos `string`, los arreglos deben ser dinámicos, y recorrerse con aritmética de apuntadores.)