

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA

3ra. práctica (tipo B) (Primer Semestre 2019)

Indicaciones Generales:

- Duración: 2h 50 min.
- Al inicio de cada programa, el alumno deberá incluir, a modo de comentario, la estrategia que utilizará para resolver el problema. De no incluirse dicho comentario, el alumno perderá el derecho a reclamo en esa pregunta.
- Si la implementación es significativamente diferente a la estrategia indicada o no la incluye, la pregunta no será corregida.
- Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 50% del puntaje asignado a dicha pregunta.
- Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado.
- El orden será parte de la evaluación.
- Su trabajo deberá ser subido a PAIDEIA en el espacio indicado por los jefes de práctica.

Pregunta 1 (10 puntos)

El almacén de ventas de una gran empresa, ordena sus productos apilándolos uno sobre otro, debido al poco espacio y gran volumen de cada uno de ellos. Para agilizar su trabajo, ubica los productos que tienen una fecha reciente abajo y los más antiguos encima, con el fin de despachar siempre la mercadería más antigua y evitar su vencimiento. Se sabe que cada pila tiene un tamaño o alto N de acuerdo a los productos que ingresen al almacén para la venta. Esta estructura se puede representar de la siguiente forma:

N = 6

2016.08.20
2017.02.22
2017.10.05
2018.03.30
2018.11.15
2019.05.02

- a) Defina la estructura que debe usar cada nodo de la pila, para almacenar lo indicado (1 punto).
- b) Desarrolle un programa que permita el ingreso del tamaño N de la pila y llene dicha estructura según el formato mostrado en la imagen anterior (1 punto).
- c) Por error un nuevo operador apilo de forma incorrecta los productos, olvidando que deben estar ordenados por antigüedad, por tal motivo se necesita que se desarrolle una función que ordene de forma correcta una pila (la pila debe quedar luego del proceso en su ubicación original), mostrando en pantalla cada paso del proceso de ordenamiento, para que ayude al operador a corregir su error. Al realizar el ordenamiento se debe recordar que dentro del almacén solo se pueden utilizar espacios para un producto de forma temporal (variables) y el espacio para una pila auxiliar (8 puntos).

Ejemplo de salida:

Desapila Pila Principal: 2018.06.02

Espacio Temporal: 2018.06.02

Apila Pila Auxiliar: 2018.06.02

Desapila Pila Principal: 2017.02.22

Pregunta 2 (10 puntos)

ISO 8583 es un estándar para transacciones financieras y define un formato de mensaje y un flujo de comunicación para que diferentes sistemas puedan intercambiar estas transacciones. Las transacciones incluyen compras, extracciones, depósitos, reintegros, reversos, consultas de saldo, pagos, transferencias entre cuentas y bloqueos de tarjetas o cuentas. Por otro lado, ALG 6969 es un estándar simplificado de la ISO 8583 desarrollado en el curso de Algoritmia de la PUCP. A continuación se describe el estándar ALG 6969:

Message Type Indicator (MTI) - Indicador de Tipo de Mensaje	Bitmap - Mapa de Bits	Data Elements - Campos de datos
--	-----------------------	---------------------------------

MTI: Este es un campo numérico de 4 dígitos que clasifica la función de alto nivel del mensaje. Un MTI incluye la versión ALG 6969, la clase (Message Class), la función (Message Function) y el origen del mensaje (Message Origin). A continuación se presentan un par de ejemplos:

Ejemplo 1 de MTI	Ejemplo 2 de MTI
2xxx ----> versión de ALG 6969 (Versión 2019) x2xx ----> clase de mensaje (Financiero) xx0x ----> función del mensaje (Petición) xxx0 ----> quien empezó la comunicación (usuario tarjeta habiente)	2xxx ----> versión de ALG 6969 (Versión 2019) X9xx ----> clase de mensaje (Bloqueo) xx0x ----> función del mensaje (Petición) xxx2 ----> quien empezó la comunicación (usuario empleado de entidad bancaria)

Bitmap: Dentro de ALG 6969, un mapa de bits es un campo o subcampo dentro de un mensaje que indica qué otros elementos (campos o subcampos) se encuentran en el mensaje. Este *Mapa de Bits* indica qué campos (Data Elements) del 1 al 8 están presentes. El mapa de bits se puede transmitir como un campo de 2 caracteres hexadecimales 0-9, A-F. Por ejemplo, el byte 42 en binario es '0100 0010' lo que significa que los campos 2 y 7 están presentes en este mensaje.

Data Elements: Son los campos individuales que llevan la información sustancial acerca de la transacción. Hay 8 campos definidos en ALG 6969:2019.

Data element	Tipo	Uso
1	n 2 (numérico de 2 dígitos)	Moneda. 01 es dólares, 02 es soles.
2	n 20	Número de cuenta origen o número de tarjeta. Si se trata de bloqueo, será la cuenta o tarjeta a bloquear.
3	n 6	Reservado, completados con ceros.
4	n 12	Monto de la transacción, siendo los dos últimos dígitos la parte decimal.
5	n 20	Número de cuenta destino.
6	n 12	Reservado, completados con ceros.
7	n 10	Fecha y hora de la transmisión en MMDDhhmmss
8	n 8	Reservado, completados con ceros.

Como ejemplo, analizaremos la siguiente trama en formato ALG 6969:

[illegible]

MTI (Tamaño: 4 dígitos)	2902	Versión ALG 2019 Bloqueo Petición Usuario empleado de la entidad bancaria
Bitmap (Tamaño: 2 caracteres)	42	Significa que los campos 2 y 7 están presentes en este mensaje.
Data Elements (Tamaño 90 dígitos)	00 (Campo 1) 12345123451234512345 (Campo 2) 000000 (Campo 3) 00000000000000 (Campo 4) 00000000000000000000 (Campo 5)	Moneda: ninguna Número de cuenta de origen Reservado Monto de la transacción: S/. 0.0 Número de cuenta destino

	000000000000 (Campo 6) 0423143052 (Campo 7) 00000000 (Campo 8)	Reservado Fecha y hora de la transmisión: 23 de abril a las 14:30:52 Reservado
--	--	--

El Banco Decrépito ha decidido utilizar el estándar simplificado ALG 6969 para sus transacciones financieras. Es por ello que ha implementado una cola en el servidor central del banco para recibir todas las tramas de peticiones de operaciones financieras enviadas desde los diferentes terminales desde cualquier parte del mundo. Y es que, debido a la gran cantidad de peticiones que se realizan en cada instante de tiempo, ha sido necesario implementar una cola de mensajes o tramas ALG 6969, de la que el servidor del banco extraerá el mensaje que está primero en la cola para procesarla. Debido a que hay peticiones que pueden considerarse como prioritarias, como por ejemplo las instrucciones de bloqueo de tarjetas o cuentas, si una petición del tipo bloqueo llega a la cola, esta deberá colocarse inmediatamente después de la última petición del tipo bloqueo si es que hubiese, sino iría como primer elemento en la cola. Como parte de la auditoría de procesos del banco, el responsable de seguridad informática del banco desea saber en un determinado momento cuáles son las peticiones que están en la cola de mensajes o tramas ALG 6969.

Elabore un programa en C que haga lo siguiente:

- Defina la estructura tipo cola de mensajes o tramas AL6969 del banco (1 punto).
- Defina la función que permita el ingreso de una petición en formato ALG 6969 a la cola de mensajes del banco, teniendo en cuenta que una petición prioritaria como la del bloqueo de tarjeta o cuenta debe colocarse inmediatamente después de la última petición del tipo bloqueo si es que hubiese, sino iría como primer elemento de la cola (5 puntos).
- Defina la función que permita la obtención de la primera petición en formato ALG 6969 de la cola de mensajes del banco (1 punto).
- Defina la función que permita mostrar en pantalla las peticiones en formato ALG6969 de la cola (1.5 puntos).
- Defina la función main que simule el envío de peticiones ALG 6969 a la cola de mensajes del banco a partir de un arreglo de tramas ALG 6969. Muestre en pantalla las peticiones ingresadas a la cola. Luego, extraiga la primera petición de la cola para su simular su procesamiento (no debe hacer nada). Finalmente, vuelva a mostrar en pantalla las peticiones restantes de la cola (1.5 puntos).

Profesores del curso: Rony Cueva
Johan Baldeón

Pando, 07 de mayo del 2019