Estructuras de Datos I

Examen I

**Nombre: Carlo Marcello Menjivar Montes de Oca**

**Cuenta: 20551123**

**Instrucciones Generales**: escriba todas sus respuestas en el presente documento y súbalo a la plataforma. El examen es a libro abierto, todo lo preguntado se resuelve con análisis del escenario, por lo que la esencia de la respuesta demuestra si tiene los conocimientos teóricos implícitos requeridos en la evaluación. (**Ver rúbrica al final del documento**):

Escenario 1:

(17%) Se necesita una estructura de datos para programar la lista de contactos para un teléfono inteligente de gama alta. ¿Qué estructura de datos recomienda para esta aplicación y por qué?  
  
**Aun tomando en cuenta que es un celular de Alta gama, generalmente vamos a concentrarnos en 2 OSs, Android y iOS. Siendo para alta Gama nos concentraríamos en los que tengan más memoria para afinar detalles en el UX (experiencia) que sea más fluido o mejores animaciones. (Me parece la lista en cuanto a datos correría en cualquier celular reciente sin limitante de memoria)  
  
En cuanto al TDA que usaríamos una lista de Contactos utiliza:**

**Insertar**

**Localizar  
Recuperar  
Suprime  
siguiente**

**anterior  
Imprime Lista**

**\* Al agregar cualquier contacto tiene que ubicarlo en un orden alfabético y mover, en dado caso, al contacto de posición si ingresamos un contacto en medio de la lista debido a su orden alfabético.**

**Con estos comportamientos elegiría un TDA tipo lista doble mente enlazada que le permitirá al usuario recorrer la lista visualmente de arriba abajo, ingresar, buscar y eliminar.**

Escenario 2:

(17%) Debido a la pandemia de COVID19 y posterior cuarentena, un banco de Honduras está implementando un sistema de tickets mediante el cual el usuario hace su cita en línea y el sistema le asigna una fecha y hora para la misma. Le han pedido recomendar una estructura de datos de forma que la asignación de tickets se realice de forma justa. Considere que las personas de la tercera edad no necesitan solicitar cita ya que se les atiende directamente cuando llegan al banco. ¿Qué estructura de datos recomienda para esta aplicación y por qué?

**Estructura de datos que realice la asignación de forma justa tendría un comportamiento tipo FIFO, First in, first out.  
  
Las personas de la tercera edad no necesitan solicitar una cita, podríamos dentro de nuestro objeto tener una variable para identificar las terceras edades interesadas, esto para que el banco tenga un estimado de cuantas personas de tercera edad podrían llegar como máximo en un día.**

**Debido a estos comportamientos realizaría un TDA tipo queue o Cola para llevar las citas de los usuarios normales.**

**La aplicación al identificar al usuario como tercera edad podría darle un anuncio al usuario de que puede ir de inmediato, pero siempre lo guardaría en la cola.**

Escenario 3:

(17%) Le han pedido implementar el software embebido en un sensor. Parte de la funcionalidad consiste en recopilar lecturas de humedad en la cuenca del Lago Yojoa y almacenarlas para posteriormente transmitirlas a través de una red de datos. Las lecturas incluyen los niveles detectados por el sensor y deben ser enviados en el orden en que se van registrando. El envío ocurre cada hora. ¿Qué TDA apoya mejor la implementación de esa funcionalidad y por qué?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 9 | dato 9 | … |  |  |  |  |
| registro | -> | 9 | dato 9 |  |  |  |
| First | In | 8 | dato 8 |  |  |  |
|  |  | 7 | dato 7 |  |  |  |
|  |  | 6 | dato 6 |  |  |  |
|  |  | 5 | dato 5 |  |  |  |
|  |  | 4 | dato 4 |  |  |  |
|  |  | 3 | dato 3 |  |  |  |
|  |  | 2 | dato 2 |  |  |  |
|  |  | 1 | dato 1 |  |  |  |
|  |  | 0 | dato 0 | transmision | - > |  |
|  |  |  |  | 0 | dato 0 | … |
|  |  |  |  | First | Out |  |

**Este es el comportamiento del software.   
  
Se toma el registro de la medición  
se pone en cola FIFO   
luego se envía/transmite el dato   
y se limpia la memoria  
  
Tipo FIFO, first in será el first out al momento de transmitir.  
  
Debido a lo que asumiré es limitante de memoria, utilizaría una linked queue, una cola que me permita limpiar el primer dato después de transmitirlo.**

Escenario 4:

(17%) Un restaurante necesita una aplicación para mostrar su menú a los clientes. ¿Qué TDA apoya mejor la implementación de esa funcionalidad y por qué?  
  
**Considero que una aplicación para mostrar un menú tendría estos comportamientos:  
- Asumiré la impresión de los datos será en base a categorías, así que dentro del objeto de mi TDA   
- tendré una variable categoría para imprimir en zonas específicas del menú.  
- Poder agregar platos en cualquier orden (ejem. si hay platos nuevos me gustaría salgan en primeros lugarares de la lista de categorías y luego cambiar su posición al gusto).  
- Eliminar cualquier plato en cualquier posición.  
  
Considerando estos comportamientos, implementaría una lista siempre enlazada que me permita todos los comportamientos arriba sugeridos.**

Escenario 5:

(17%) Un call center necesita una aplicación para que los clientes esperen mientras son atendidos. ¿Qué TDA apoya mejor la implementación de esa funcionalidad y por qué?  
  
**Similar al ejercicio Escenario dos, un comportamiento tipo FIFO, first in First out, un TDA tipo cola.  
Si quiero mejorar aun más, podría agregar una funcionalidad de poder adelantar o darle prioridad en la cola a algún tipo de caso.**

Escenario 6:

(17%) Para hacer la conversión de decimal a binario, hay que ir dividiendo el número decimal entre dos y anotar en una columna a la derecha el resto (un 0 si el resultado de la división es par y un 1 si es impar).  
La lista de ceros y unos leídos de abajo a arriba es el resultado.

Ejemplo: vamos a pasar a binario 7910

79**1**(impar). Dividimos entre dos:  
39**1** (impar). Dividimos entre dos:  
19**1**(impar). Dividimos entre dos:  
9**1**(impar). Dividimos entre dos:  
4**0** (par). Dividimos entre dos:  
2**0** (par). Dividimos entre dos:  
1     **1** (impar). **<- Last in**

Por tanto, 7910 = 1001111

¿Qué TDA apoya mejor la implementación de este algoritmo? Muestre cómo utilizaría el TDA para implementar la solución

**El orden de registro de los datos sería**

**1111001 <- Last in**

**Si implementamos una pila, queremos implementar un Last In, First out.**

**While X!=null**

**Print X.valor  
 X= X.Siguinte**

**esto imprime**

**1001111**

**Rúbrica**

|  |  |  |
| --- | --- | --- |
| Item | Si (100%) | No  (0%) |
| Identifica las estructuras apropiadas de acuerdo a las características del problema. |  |  |
| Comprende la implementación de las estructuras y selecciona la implementación adecuada de acuerdo a un escenario. |  |  |