**Método de diseño de ingeniería**

**Definición del problema:**

Una reconocida institución prestadora de salud requiere un sistema que permita administrar el ingreso y egreso de pacientes en un laboratorio clínico; esta debe permitir organizar los pacientes entre acceso general y prioritaria para ser atendidos respectivamente.

**Recopilación de información:**

* Satck:

Es una lista ordenada o estructura de datos que permite almacenar y recuperar datos, su principal característica es su modo de acceso a los elementos, el cual es el tipo L.I.F.O del ingles Last in, First out. (último en entrar, primero en salir).

* Queue:

Es un tipo de dato abstracto, caracterizada por ser una secuencia de elementos en la que la operación de inserción (push) se realiza por un extremo, su característica principal es su modo de acceso a los elementos, el cual es el tipo F.I.F.O. del inglés First in, First Out (primero en entrar, primero en salir).

* Colas de prioridad:

Una cola de prioridad es un tipo de dato abstracto similar a una cola en la que los elementos tienen adicionalmente, una **prioridad** asignada.

Entre sus características están:

Añadir con prioridad: se añade un elemento a la cola, con su correspondiente prioridad.

Eliminar elemento de mayor prioridad: se devuelve y elimina el elemento con mayor prioridad mas antiguo que no haya sido desencolado de la cola.

* Montículo:

En computación, un montículo o heap en ingles es una estructura de datos de tipo árbol que contiene información perteneciente a un conjunto ordenado y estos se denominan en dos categorías por un lado están los montículos máximos cuya característica de cada nodo padre tiene un valor mayor que el de cualquier nodo hijo, y por otro lado están los montículos mínimos, los cuales el valor de cada nodo padre es menor al de sus nodos hijos.

* Tabla hash:

Una tabla hash o también conocida como matriz asociativa, es una estructura de datos que implementa el tipo de dato abstracto llamado Diccionario, las tablas hash asocia llaves o calves con valores. La operación principal que soporta de manera eficiente, es decir, para lo que principalmente se usa es para la búsqueda: permite el acceso a los elementos, un ejemplo de esto sería teléfono y dirección, estos almacenados a partir de una clave generada usando un nombre o un id por ejemplo. Funciona transformando la clave con una función hash en un hash, un numero que identifica la posición donde la tabla hash localiza el valor deseado.

* Linked-List:

En informática, una lista enlazada o linked-list es una colección lineal de elementos de datos cuyo orden no está dado por su ubicación física en la memoria. A diferencia de esto, cada elemento apunta al siguiente. Esta estructura consta de una colección de nodos que al juntarse representan una secuencia.

Las listas enlazadas se encuentran entre las estructuras de datos más simples y comunes. Estas se pueden usar para implementar varios otros tipos de datos abstractos como por ejemplo las tablas hash las pilas y colas etc.

**Búsqueda de soluciones:**

Alternativa 1:

Hacer uso de las colas de prioridad para asignar turno a los pacientes con mayor prioridad, Queues para asignar turno a las personas que no tienen ninguna prioridad, después hacer uso de las tablas hash para agruparlos a todos. También hacer uso de las Stacks o pilas para implementar la opción de deshacer la última acción realizada.

Alternativa 2:

Hacer uso de una sola cola de prioridad para ingresar a todos los pacientes y asignarles su turno, y tabla hash para guardar a todos los pacientes. También hacer uso de las Stacks o pilas para implementar la opción de deshacer la última acción realizada.

Alternativa 3:

Hacer uso de colas de prioridad para asignar a los pacientes con mayor prioridad su turno y así mismo usar colas sencillas para asignar a los demás pacientes en atención general. También hacer uso de las Stacks o pilas para implementar la opción de deshacer la última acción realizada.

**Transición de la formulación de ideas a los diseños preliminares:**

Como primera medida se opta por descartar la Alternativa numero 3 debido a que consideramos la más incompleta para implementar una solución solida y factible haciendo uso de las buenas prácticas para una implementación de software ya que al implementar solo dos colas una de prioridad y una sencilla sin hacer uso de una estructura de datos como la tabla hash hace del algoritmo mucho mas pesado e ineficiente a largo plazo.

Revisión de las otras alternativas:

Alternativa 1:

* Es preciso y puede tener buenas practicas en el proceso de su implementación sin embargo puede llegar a ser extenso y propenso a caer en malas practicas si no se tiene en cuenta como implementar las colas de prioridad de una manera correcta.
* Esta alternativa implica implementar una tabla hash propia, así como también las colas de prioridad y las pilas para su correcto funcionamiento.

Alternativa 2:

* al hacer uso de una sola cola de prioridad estamos delegando muchas funciones a una sola parte del sistema esto haría que nos encontráramos con malas practicas violando los criterios de diseño de software como lo son la partición de obligaciones.

**Evaluación y selección de la mejor solución:**

Criterios de evaluación:

Criterio A.

Cumplimiento de los criterios de diseño.

Criterio B.

Flexibilidad. Solución flexible y escalable a futuro.

Criterio C.

Completitud. Hace uso de todo lo necesario para ser funcional y sostenible a futuro.

Modo de evaluación para cada una:

Numérica, 0 siendo el que menos cumple con el criterio y 5 si cumple completamente.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Criterio A | Criterio B | Criterio C | Total |
| Alternativa 1 | 4 | 4 | 5 | 13 |
| Alternativa 2 | 2 | 3 | 2 | 7 |

Selección:

De acuerdo con la evolución anterior se debe seleccionar la alternativa 1, ya que obtuvo la mayor puntuación de acuerdo con los criterios definidos.