Universidad de San Carlos de Guatemala
Escuela de Ingeniería en Ciencias y Sistemas, Facultad de Ingeniería
Introducción a la programación y computación 2, 1er. Semestre 2021.

SIMULADOR DE ATENCIÓN

202112145 – Daniel Estuardo Cuque Ruíz

Resumen

Una empresa guatemalteca nos ha contactado para crear un programa que permita manejar el control de los clientes durante el ingreso y atención de los mismos.

Para ello se utilizaron estructura de datos, que permite crear memoria dinámica con los nodos que representan datos dentro las estructuras.

También permite generar archivos de resultados, donde se muestra el resultado final

Palabras clave

Simulación, estructura de datos, reportes, identificación de patrones

Abstract

The Guatemalan epidemiological research laboratory has been investigating how diseases infect the cells of the human body and spread, causing serious illness and even death.

For this, software was developed that allows epidemiologists to perform a simulation with tissue from infected cells.

Thus, facilitating the simulation of several patients, reducing the time in the diagnosis.

It also improves the percentage of correct diagnosis in patients. Reducing the cost of equipment and the monetary costs of running a simulation.

Keywords

Simulation, data structures, report, pattern identification.

Introducción

El programa realiza la simulación de tejido infectado, por medio de la carga de archivos previamente configurados, donde se toma un patrón inicial, y a

partir de ese patrón se generan las simulaciones, hasta acabar con el número de periodos del paciente

Con este programa de simulación, se pueden dictaminar mejores diagnósticos a los pacientes, y la probabilidad de curar su enfermedad en una etapa temprana será más alta.

Reduciendo el costo de tiempo, y recursos ecónomicos, así como la reducción de infraestructura necesaria para realizarlo

Desarrollo del tema

A continuación se presenta la estructura del proyecto, así como los conceptos utilizados para el desarrollo de los algoritmos

a. Arquitectura del proyecto

Se utilizó la arquitectura, Modelo Vista, Controlador, por sus siglas MVC.

El modelo de arquitectura Model-View-Controller (MVC) separa una aplicación en tres componentes principales: el modelo, la vista y el controlador.

MVC es un modelo de diseño estándar con el que están familiarizados muchos desarrolladores.

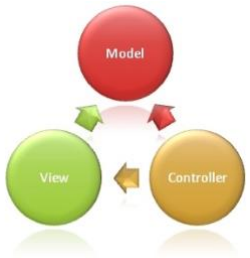

Figura 1. Estructura del proyecto para la simulación de células

Fuente: elaboración propia

- Modelos. Los objetos de modelo son las partes de la aplicación que implementan la lógica para el dominio de datos de la aplicación. A menudo, los objetos de modelo recuperan y almacenan el estado del modelo en una base de datos.

- Vistas. Las vistas son los componentes que muestran la interfaz de usuario (UI) de la aplicación. Normalmente, esta interfaz de usuario se crea a partir de los datos de modelo.

- Los controladores son los componentes que controlan la interacción del usuario, trabajan con el modelo y por último seleccionan una vista para representar la interfaz de usuario. En una aplicación de MVC, la vista solo muestra información; el controlador controla y responde a la interacción y los datos que introducen los usuarios. (Microsoft, 2022)

b. Estructura de datos

Las estructuras de datos se basan generalmente en la capacidad de un ordenador para recuperar y almacenar datos en cualquier lugar de su memoria.

Las operaciones más comunes en estructura de datos son:

- Insertar al final
- Insertar al principio
- Insertar en un índice en específico
- Eliminar el último nodo
- Eliminar por índice

c. Lista doblemente enlazada

Cada nodo contiene dos enlaces, uno a su nodo predecesor y el otro a su sucesor. La lista es

eficiente tanto en recorrido directo (hacia adelante) como recorrido inverso (hacia atrás). (Listas, 2020)

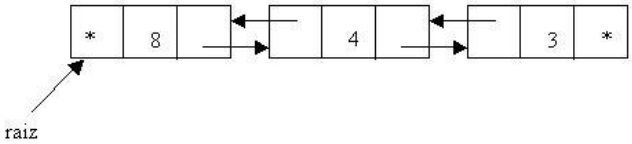

Figura 2. Representación de una lista doblemente enlazada

Fuente: elaboración propia

Otra ventaja de las listas doblemente enlazadas es que podemos usar un puntero a la celda que contiene el i-ésimo elemento de una lista para representar la posición i, mejor que usar el puntero a la celda anterior aunque lógicamente, también es posible la implementación similar a la expuesta en las listas simples haciendo uso de la cabecera.

El único precio que pagamos por estas características es la presencia de un puntero adicional en cada celda y consecuentemente procedimientos algo más largos para algunas de las operaciones básicas de listas.

d. Lista simplemente enlazada

Cada nodo (elemento) contiene un único enlace que conecta ese nodo siguiente o sucesor. La lista es eficiente en recorridos directos (hacia adelante).

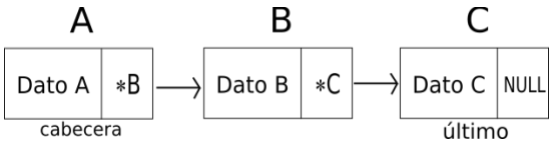

Figura 3. Representación de una lista simplemente enlazada

Fuente: elaboración propia

e. Patrón de infección

- Toda célula contagiada, continúa contagiada si tiene exactamente 2 o 3 células contagiadas en las celdas vecinas, de lo contrario sana para el siguiente periodo.
- Cualquier célula sana que tenga exactamente 3 células contagiadas en las celdas vecinas, se contagia para el siguiente período.

f. Diagnósitco

- Que el patrón inicial se repita siempre después de "N" períodos, en este caso la enfermedad produce un caso grave. Si "N" es igual a 1, entonces, el paciente morirá a causa de la enfermedad, ya que ésta será incurable.
- Que algún patrón, distinto al patrón inicial, se repita luego de "N" períodos cada "N1" períodos, en este caso la enfermedad producirá un caso grave. Si "N1" es igual a 1, entonces, el paciente morirá a causa de la enfermedad, ya que ésta será incurable, en caso de que "N1" es mayor que 1, la enfermedad será grave.

Conclusiones

La utilización de listas doblemente enlazadas, facilitan el recorrido de nodos posteriores y anteriores del nodo actual, ya que no se debe recorrer toda la lista nuevamente.

Utilizar clases base para la creación de distintas listas, facilita la lectura del código al realizar modificaciones, o al agregar funcionalidades.

El rendimiento de la librería tkinter como interfaz gráfica, se nota ineficiente al momento de simular los períodos.

La creación de reportes gráficos facilita la comprensión de la información de la simulación en un paciente en específico.

Referencias bibliográficas

Microsoft. (11 de 07 de 2022). *Documentos de Microsoft*. Obtenido de https://docs.microsoft.com/es-es/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview

Listas, U. I. (2020). *UAEM*. Obtenido de Estructuras de Datos Lineales: http://ri.uaemex.mx/bitstream/handle/20.500.11799/34465/secme-18391.pdf?sequence=1

Apéndices

Operaciones	Matricial	Enlace Simple	Enlace Doble
Crear	O(1)	O(1)	O(1)
Destruir	O(1)	O(n)	O(n)
Primero	O(1)	O(1)	O(1)
Fin	O(1)	O(n)/O(1)	O(1)
Insertar	O(n)	O(1)	O(1)
Borrar	O(1)	O(n)	O(n)
Elemento	O(1)	O(1)	O(1)
Siguiente	O(1)	O(1)	O(1)
Anterior	O(1)	O(n)	O(1)
Posicion	O(n)	O(n)	O(n)
• Memoria	• Tamaño fijo	• Sobrecarga de un puntero por nodo. • Un nodo sin información.	• Sobrecarga de dos punteros por nodo. • Un nodo sin información.

Figura 5. Eficiencia del uso de distintas estructura de datos

Fuente: Universidad de Granada

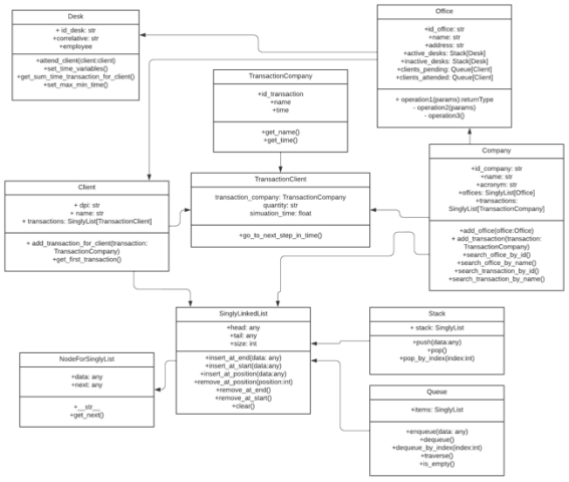

Figura 4. Diagrama de clases utilizadas en el proyecto

Fuente: elaboración propia