|  |
| --- |
| **PROYECTO NO. 2** |
| **201807154 – Denny Alexander Chalí Miza** |

**Resumen**

Para la realización del proyecto No. 1 se realizó una aplicación cuyas funciones son las de analizar un archivo de texto plano con formato XML, el cual contiene los datos de distintos pacientes. La función principal del programa es la de crear una malla con la información cargada en el archivo para luego analizarla y obtener una un diagnostico de la gravedad de la enfermedad del paciente.

Para la solución del problema se hizo uso de TDA, usando listas simples, listas doblemente enlazadas y listas ortogonales, siendo la lista ortogonal la que contiene los datos de las mallas de las células de cada paciente.

Para mostrar gráficamente la malla de las células sanas y enfermas se hizo uso del software Graphviz, la cual muestra la matriz que contiene la lista ortogonal y la lista de celdas que componen la malla.

**Palabras clave**

Pila, Lista, Funciones, Métodos y Clases.

***Abstract***

*For the realization of project No. 1, an application was developed whose functions are to analyze a plain text file in XML format, which contains the data of different patients. The main function of the program is to create a mesh with the information loaded in the file and then analyze it to obtain a diagnosis of the severity of the patient's disease.*

*For the solution of the problem we made use of TDA, using simple lists, double linked lists and orthogonal lists, being the orthogonal list the one that contains the data of the meshes of the cells of each patient.*

*To graphically display the mesh of healthy and diseased cells, use was made of Graphviz software, which shows the matrix containing the orthogonal list and the list of cells that make up the mesh.*

***Keywords***

*Stack, List, Functions, Methods and Clases.*

**Introducción**

Se desarrollo una aplicación que simula un sistema de atención de clientes diseñado para cualquier institución o empresas.

La aplicación permite la carga, lectura y análisis de archivos de texto plano en formato XML, las cuales contienen los datos que necesita el programa para inicializar la simulación. También es posible registrar las empresas al sistema de forma manual.

El programa permite de forma sencilla la selección de cada una de las funciones y muestra de forma gráfica y detallada el estado en que se encuentra el punto de atención de cada punto de atención que se selecciona.

**Desarrollo del tema**

1. **Preparando el entorno de trabajo**

Para la resolución del problema se utilizó el lenguaje de programación Python, el cual deberá de descargar e instalar en su ordenador, para el correcto funcionamiento del programa se recomienda instalar una versión superior a la 3.6.

Graphviz

Graphviz es un software de visualización de gráficos de código abierto que nos servirá para graficar el estado del punto de servicio..

Para su instalación debemos de tener previamente instalado Python.

Abrimos el símbolo del sistema (CMD) e ingresamos:

$ pip install graphviz

Para renderizar el código fuente DOT generado se puede usar el software Grahviz descargándola desde su sitio oficial.

1. **TDA implementados**

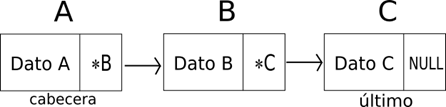
Para almacenar los datos de los archivos y manejar las rutas en cada malla fueron necesarios implementar distintos tipos de TDA.

Listas simples enlazadas

Una lista simple o lista simplemente ligada está constituida por un conjunto de nodos alineados de manera lineal (uno después de otro) y unidos entre sí por una referencia (apuntador).

A diferencia de un arreglo, el cual también es un conjunto de nodos alineados de manera lineal, el orden está determinado por una referencia, no por un índice, y el tamaño no es fijo.

La unidad básica de un alista simple es un elemento nodo, cada elemento de la lista es un objeto que contiene la información que se desea almacenar, así como una referencia (NEXT) al siguiente elemento (sucesor).



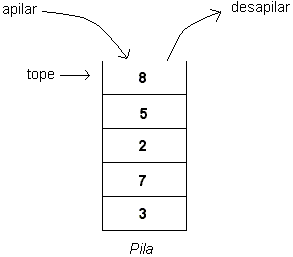
*Figura 1.* Lista enlazada simple

Fuente: Alberto Castillo G, obtenido de:

[www.betoissues.com](http://www.betoissues.com).

Pilas

Una pila (stack) es una lista de elementos de la cual solo se puede extraer el último elemento insertado. La posición en donde se encuentra dicho elemento de denomina tope de la pila. También se conoce a las pilas como listas LIFO (LAS IN – FIRST OUT: el último en entrar es el primero en salir.)



*Figura 2.* Representación gráfica de una pila.

Fuente: Bruno López Takeyas, M.C, obtenido de:

www.itnuevolaredo.edu.mx/Takeyas

1. **Ejecución del programa**

Posicionarse en la carpeta (CMD)

Una vez el entorno de trabajo está listo podemos iniciar el programa, para ello abrimos una pestaña en el símbolo del Sistema (CMD) y navegamos hasta estar en la carpeta donde se ubica el ejecutable de nuestro programa:

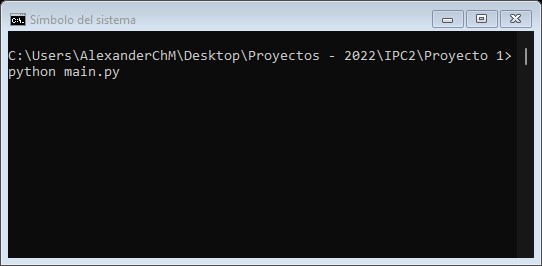
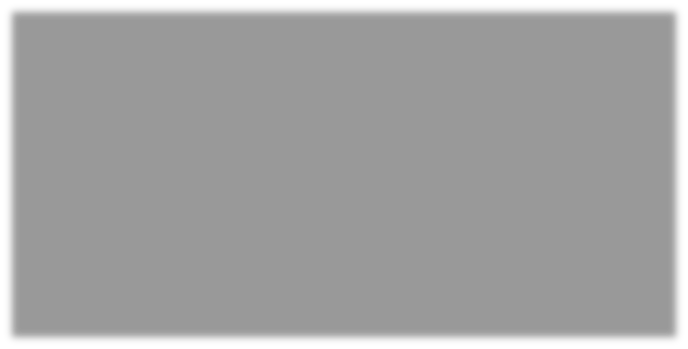
Texto

Descripción generada automáticamente

*Figura 3.* Ejemplo de ubicación en carpeta

Fuente: elaboración propia.

Ingresamos el siguiente comando una vez dentro de la carpeta que contine el ejecutable:



*Figura 4.* Inicializando el programa Fuente: elaboración propia.

Una vez realizado los pasos anteriores se iniciará el programa desplegando el menú de inicio con cada una de las funciones.

1. **Funciones del programa**

Menú inicio

Al iniciar el programa se desplegará el menú inicial con las principales funciones como se muestra en el ejemplo:

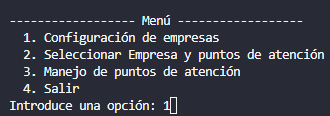
Texto

Descripción generada automáticamente

*Figura 5.* Menú de inicio.

Fuente:elaboración propia.

Para seleccionar una función se deberá de ingresar el número de la opción correspondiente como se muestra a continuación:



*Figura 6.* Ingreso de opción

Fuente:elaboración propia.

Para los submenús se utiliza el mismo método para seleccionar opciones y seleccionar elementos.

Configuración de empresas

Texto

Descripción generada automáticamente

*Figura 7.* Menú configuración de empresas

Fuente:elaboración propia.

1. Limpiar sistema

Inicializa todas las estructuras de datos para iniciar una prueba desde cero

1. Cargar archivo (configuración de sistema)

Esta opción se encarga de desplegar una ventana emergente y seleccionar el archivo de texto plano en formato XML que contine los datos de cada empresa, sus puntos de ventas, escritorios y transacciones disponibles.

Captura de pantalla de computadora

Descripción generada automáticamente

*Figura 8.* Selección de archivo XML

Fuente:elaboración propia.

Una vez se localiza el archivo XML con los datos se selecciona, si los datos son correctos se mostrará un mensaje indicando los resultados satisfactorios de lectura, en caso contrario se mostrará un mensaje indicando que los datos no se cargaron.

1. Crear nueva empresa

Permite crear y agregar al sistema una empresa de forma manual con los datos de puntos de atención, escritorios de servicio y transacciones que maneja la empresa.

1. Cargar Archivo (Configuración Inicial)

Carga el archivo con la configuración inicial de la prueba. Este archivo contiene los datos de los escritorios activos y los clientes que se atenderán.

Esta opción desplegara una ventana emergente similar a la de punto 2, procedimiento para seleccionar el archivo es el mismo que el de la opción para cargar el archivo de las configuraciones del sistema.

1. Regresar

Esta opción nos regresa al menú de inicio

Seleccionar empresa y punto de atención

Esta opción permitirá elegir la empresa y punto de atención cargados con anterioridad al sistema sobre los cuales se realizarán las pruebas.

Se deberá de seleccionar una opción ingresando el número del índice del nombre de la empresa.

Texto

Descripción generada automáticamente

*Figura 9.* Selección de empresa

Fuente:elaboración propia.

Para seleccionar el punto de atención también se deberá de ingresar el índice del nombre del punto de atención.

Texto

Descripción generada automáticamente

*Figura 10.* Selección de punto de atención

Fuente:elaboración propia.

Manejo de Puntos de atención

Esta opción despliega el siguiente menú con las funciones que pueden ser efectuadas sobre el punto de atención que se seleccionó.

Texto

Descripción generada automáticamente

*Figura 11.* Menú configuraciones del sistema.

Fuente:elaboración propia.

1. Ver estado de punto de atención.

Muestra el estado actual del punto de atención de forma gráfica los datos a mostrar son los siguientes:

Punto de atención: Cantidad de escritorios de servicio activos, cantidad de escritorios de servicio inactivos, clientes en espera de atención, tiempo promedio de espera, tiempo máximo de espera, tiempo mínimo de espera, tiempo promedio de atención, tiempo máximo de atención, tiempo mínimo de atención

Por cada escritorio de servicio activo: Tiempo promedio de atención, tiempo máximo de atención, tiempo mínimo de atención. Además, debe mostrar claramente el comportamiento de este listado.

Tabla

Descripción generada automáticamente con confianza media

*Figura 12.* Ejemplo gráfico de la representación del estado actual del punto de servicio selesccionado

Fuente:elaboración propia.

1. Activar escritorio de servicio inactivo

Permite que un escritorio de servicio inactivo cambie a estado activo e inicie la atención de clientes.

1. Desactivar escritorio

Permite que un escritorio de servicio activo cambie a estado inactivo, entonces, al momento que este escritorio se desocupe, no se le asignarán más clientes a atender

1. Atender Cliente

Esta operación concluye la atención del cliente más próximo a ser atendido.

1. Solicitud de atención

Esta operación permite agregar un cliente que solicita atención, debe permitir la selección de las transacciones que realizará y mostrará como respuesta el tiempo promedio de espera para este cliente.

**2.2.3.1 Ejecutar periodos**

Esta opción ejecuta los periodos uno a uno para ver gráficamente la forma en que se desarrolla la enfermedad en el tejido proporcionado en la rejilla inicial, además se identifica el momento en que el patrón inicial se repita o en donde el patrón N1 se repite, indicando el periodo en que apareció el patrón y en cuantos periodos empezó a repetirse.

**Referencias bibliográficas**

Lewis, John; Loftus, William (2009). *Java Software Solutions Foundations of Programming Design* 6th ed. Pearson Education Inc.

Chacón Sartori, Camilo. *Computación y programación funcional: introducción al cálculo lambda y la programación funcional usando Racket y Python*. [Barcelona]: Marcombo.

**Anexos**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

*Figura 13.* Diagrama de clases para la solución del problema

Fuente: elaboración propia