
Proyecto juego de posicionamiento de piezas

201930566 - Alexander Manuel de Jesús Tzoc Alvarado

Resumen

El proyecto presentado es producto del aprendizaje obtenido tanto en la clase magistral como en el laboratorio de IPC2. Para su elaboración fue necesaria la implementación de TDAs tales como una matriz dispersa que, básicamente, funciona como una lista de listas. También se implementó una lista simple para guardar datos como figuras necesarias para el funcionamiento del juego, almacenar a los jugadores y además otros elementos importantes. Para la elaboración de la interfaz gráfica se utilizó la librería *PyTQ5*; para el diseño se utilizó el software *QtDesigner*. El juego cuenta simplemente con una ventana muy sencilla, pero bastante completa, en donde están todos los controles necesarios para su óptimo funcionamiento. Ya que la aplicación requería que los juegos puedan ser guardados en un documento *.xml*, se utilizó la librería *xml.etree.cElementTree* para leer y escribir dicho documento. Cuando las partidas finalizan (si encuentra un ganador o se declara empate), los datos de las partidas son guardados en el archivo *index.html* que se encarga de mostrar un resumen de cada partida.

Palabras clave

Matriz, lista, fichero, nodos, puntero

Abstract

This project is the result of the learning obtained both in the master class and in the IPC2 laboratory. For its elaboration it was necessary to implement ADTs such as a sparse matrix that basically works as a list of lists. A simple list was also implemented to save data such as figures, players and also other important elements. For the development of the graphical interface, the PyTQ5 library was used; QtDesigner software was used for the design. The game has a very simple window, but quite complete, where are all the necessary controls for its optimal functioning. Since the application required that the games be saved in an .xml document, the xml.etree.cElementTree library was used to read and write the document. When a game ends (if a winner is found or a tie is declared), the game data is saved in the index.html file that is responsible for showing a summary of each game.

Keywords

Matrix, list, file, node, pointer

Introducción

Para resolver diferentes problemas se pueden utilizar diferentes lenguajes de programación. En este caso, ya que Python es el lenguaje manejado en el curso de IPC2, entonces fue el utilizado. Python a diferencia de otros lenguajes, es muy simple en su escritura, pero también es muy poderoso. Para manejar datos existen muchas formas de hacerlo, pero en este caso se usaron estructuras de datos propiamente programadas. Para entender las estructuras es necesario tener claro el manejo de memoria dinámica y el concepto de punteros, lo demás es pensar y pensar. La programación orientada a objetos también es gran protagonista para la resolución de este problema. Con la implementación de clases el manejo de los datos es mucho más sencillo y ameno. Este es un proyecto que fuerza al estudiante a sacar lo mejor de sí mismo.

Desarrollo del tema

La descripción del general del problema fue el siguiente: “La aplicación consiste en un juego el cual contará con un tablero de $m \times n$ representado a través de una matriz dispersa, dicho juego contará con seis diferentes piezas las cuales pueden ser colocadas a lo largo del tablero de juego, tomando en cuenta las restricciones de movimientos que serán especificados a lo largo del enunciado”.

Para la elaboración primero se planteó la utilización de una matriz ortogonal, pero al final se optó una matriz dispersa. Se diferencian en básicamente en la forma de guardar sus datos. Por un lado la matriz ortogonal es una matriz totalmente cuadrada, con todas sus posiciones llenas, a lo que la matriz dispersa se asemeja más a una escalera, en donde solo se almacenan los nodos usados. De esa manera se aprovecha mejor los recursos

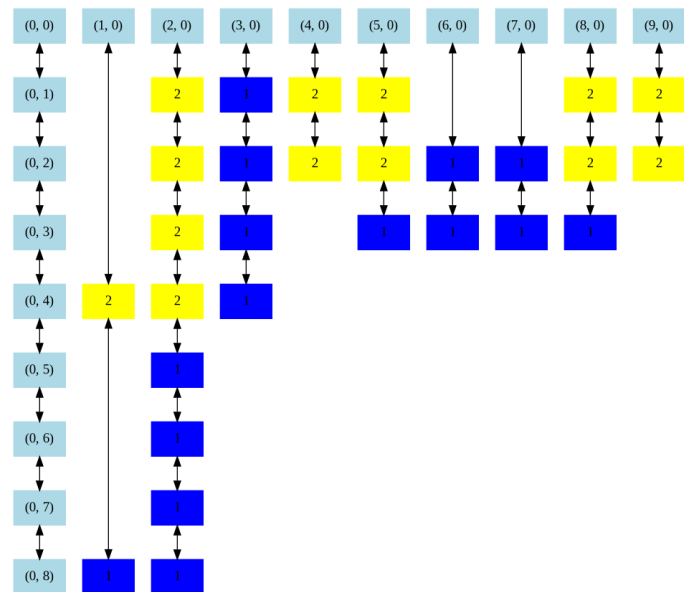


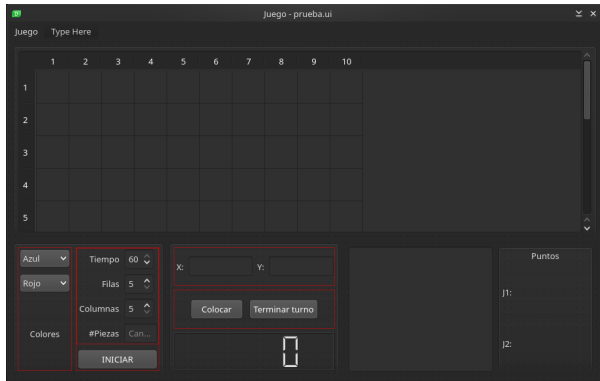
Figura 1. Matriz dispersa.

Fuente: Elaboración propia utilizando la librería graphviz

La matriz consta de un nodo raíz que tiene a su derecha a los nodos columnas y abajo a los nodos filas, de esa manera los nodos logran conectarse y relacionarse con toda la matriz. De esa manera se lograron insertar las piezas al tablero. Utilizando los punteros de “up”, “down”, “right” y “left” se logró validar que, efectivamente, la pieza podía ser insertada en la casilla ubicada. Para el ingreso de las coordenadas se tomó la casilla más arriba y a la izquierda que tuviera la figura.

En la implementación de la interfaz gráfica se utilizó el software QtDesigner que hizo el trabajo mucho más sencillo. La ventana es sencilla pero abarca todo lo necesario. Para pintar el tablero se utilizó un *QTableWidget* que permitía representar las piezas de manera comprensible para el usuario.

Solo se utilizó una ventana porque se consideró que de esa manera era más sencillo el manejo para el usuario, además de que el juego necesita pocos controles para su manejo.



Generar la imágenes de la matriz dispersa fue un trabajo realizado por la librería Graphviz. Se crearon nodos para poder graficarlos, cada nodo tiene el color que el jugador escogió para poder jugar. La representación de los TDAs es muy útil para su comprensión. De esa manera se logra ver que la implementación de la estructura de datos es correcta.

Para la lectura de xml se validaron los datos de entrada y se procedió a cargar las partidas. XML consiste en un lenguaje de marcado creado por el W3C (World Wide Web Consortium), con la finalidad de definir una sintaxis para la codificación de documentos, que tanto los usuarios como las propias máquinas en sí puedan ser capaces de leer.

Para ello, lo hace mediante la utilización de una serie de etiquetas que definen la estructura que posee el documento en cuestión, además de cómo debe ser transportado y almacenado.

Podríamos compararlo con otro lenguaje de marcado con el que posiblemente nos encontremos más familiarizados: el lenguaje de marcado de hipertexto (**HTML**), usado esencialmente para la codificación de las **páginas web**, y que utiliza un conjunto de símbolos de marcado predefinidos que describen el formato que posee el contenido de una página web.

Todas las partidas se guardan en un mismo documento llamado *games.xml*. Este documento cuenta con un formato que permite guardar los elementos más esenciales para las partidas. Su

estructura está compuesta por una etiqueta raíz `<matriz>`, dentro de dicha etiqueta se guardan los juegos que el usuario decida. Para guardar las partidas se utiliza la etiqueta `<matrices>` y dentro de ella usando etiquetas se guardan el nombre de la partida, el color del jugador 1, el color del jugador 2, la cantidad de filas, la cantidad de columnas y la imagen de la partida, que consiste en una sucesión de caracteres; “1” y “2” para las piezas de los jugadores y “-” para casillas que no están ocupadas.

```
<matriz>
  <nombre>hola</nombre>
  <color1>Azul</color1>
  <color2>Rojo</color2>
  <filas>10</filas>
  <columnas>10</columnas>
  <imagen>-22-11---- 222211----
</matriz>
```

Al finalizar una partida era requerido guardar los datos de la partida con el historial de movimientos, ya sean correctos o no, y el resultado de la partida en un documento *HTML*. Según un post realizado en la página de *Código Facilito*:

“HTML es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de la siglas que corresponden a *HyperText Markup Language*, es decir, Lenguaje de Marcas de Hipertexto. Muy parecido a *XML*, pero con etiquetas previamente definidas por el lenguaje”. (Flores Herrera, 2015)

Para una mejor visualización de los reportes se utilizó CSS. “Hojas de Estilo en Cascada (del inglés Cascading Style Sheets) o CSS es el lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML (en-US) (incluyendo varios lenguajes basados en XML como SVG, MathML o XHTML). CSS describe como debe ser renderizado el elemento estructurado en la pantalla, en papel, en el habla o en otros medios.” (CSS | MDN, 2021).

Conclusiones

Las estructuras de datos son útiles para el correcto aprovechamiento de los recursos computacionales. De igual manera la programación orientada a objetos ayuda en gran medida a su implementación, aunque si ella sería igualmente implementable. Conocer sobre el manejo de ficheros es útil para poder crear aplicaciones más complejas

Referencias bibliográficas

CSS | MDN. (2021, 19 de junio). MDN Web Docs.
<https://developer.mozilla.org/es/docs/Web/CSS>

Flores Herrera, J. (2015, 25 de agosto). Qué es HTML. CódigoFacilito.
<https://codigofacilito.com/articulos/que-es-html>

colaboradores de Wikipedia. (2021, 10 junio). Extensible Markup Language. Wikipedia, la enciclopedia libre.
https://es.wikipedia.org/wiki/Extensible_Markup_Language

colaboradores de Wikipedia. (2019, 23 octubre). Matriz dispersa. Wikipedia, la enciclopedia libre.
https://es.wikipedia.org/wiki/Matriz_dispersa

Punteros — Programación. (s. f.).
<http://progra.usm.cl/>. Recuperado 21 de junio de 2021, de
<http://progra.usm.cl/apunte/c/punteros.html#:%7E:txt=Un%20puntero%20es%20un%20tipo,es%20una%20direcci%C3%B3n%20de%20memoria.&text=Cuando%20una%20variable%20de%20tipo,que%20est%C3%A1%20en%20esa%20direcci%C3%B3n>.