
Proyecto 2. Máquina de Ensamblaje

201905515 – CARLOS AUGUSTO CALDERON ESTRADA

Resumen

Se desarrollo un software capaz de guardar en memoria archivos XML, para esto se usaron distintos tipos de TDA, lista simple enlazada y Lista doblemente enlazada. Se recorrió los distintos TDA para obtener los datos de ensamble y se gestionaron dentro de un algoritmo que nos retorna el tiempo óptimo para ensamblar el producto. Al igual que graficar la cola de prioridad de cada producto por medio de un software de terceros llamado Graphviz y por último sus reportes en archivos XML y HTML.

Palabras clave

XML, TDA, LISTA DOBLE, LISTA, HTML

Abstract

A software capable of storing XML files in memory was developed, using different types of TDA, simple list and double-linked simple list. The different TDAs were covered to obtain the assembly data and they were managed within an algorithm that returns us the optimal time to assemble the product. As well as the priority queue of each product by means of a third-party software called Graphviz and finally its reports in XML and HTML files.

Keywords

XML, TDA, DOUBLE LIST, LIST, HTML

Introducción

El uso de archivos XML es cada vez más usual entre los distintos tipos de software que existen, por lo mismo, la lectura de estos fue implementada en el proyecto, con la ayuda de la librería Element Tree, misma que funciona para guardar archivos de salida con nueva información luego de la manipulación de las listas de listas y las listas dobles enlazadas. Los Tipos de Datos Abstractos son una herramienta muy eficaz, permitiendo el fácil manejo de la información de cada lista. Por último se gestionaron los datos para realizar al algoritmo de la simulación de ensamblaje con la lista doble enlazada y generamos los reportes en los distintos tipos de archivos HTML y XML.

Desarrollo del tema

El proyecto consta de varias partes, la primera y una de las más importante, es las lecturas de los archivos XML, el xml maquina y el xml simulación, estos contienen todos los datos de la máquina, número de líneas, listado de líneas con su tiempo de ensamblaje y cantidad de componentes, los productos contienen sus líneas y componentes para gestionar la simulación, mientras que el archivo de simulación contiene los productos a simular de una vez. Una vez que se lee el archivo máquina, se presentaran todos los productos en un combo box al seleccionar la lista de componentes muestra cada componente que se necesita durante el ensamblaje, se hace uso de una lista doble enlazada para trabajar la simulación del producto y se muestra en un componente lcd el tiempo óptimo de ensamblaje, en este caso, los reportes posibles a generar son en HTML y XML por medio de un botón para cada reporte, estos se ubican en sus respectivas carpetas, para el tema de la simulación masiva, los datos del archivo son guardados de igual manera en una lista simple y estos

se podrán ver enlistados en un combo box, la tabla automáticamente mostrara los datos del producto seleccionado y los reportes se generan por separado e igual en sus respectivas carpetas.

XML, siglas en inglés de eXtensible Markup Language, traducido como "Lenguaje de Marcado Extensible" o "Lenguaje de Marcas Extensible", es un metalenguaje que permite definir lenguajes de marcas desarrollado por el World Wide Web Consortium utilizado para almacenar datos en forma legible.

- a. TDA
En ciencias de la computación un tipo de dato abstracto o tipos abstractos de datos, es un modelo matemático compuesto por una colección de operaciones definidas sobre un conjunto de datos para el modelo.
- b. Lista simplemente enlazada
Es una lista enlazada de nodos, donde cada nodo tiene un único campo de enlace. Una variable de referencia contiene una referencia al primer nodo, cada nodo (excepto el último) enlaza con el nodo siguiente, y el enlace del último nodo contiene NULL para indicar el final de la lista.
- c. Lista doblemente enlazada
En una lista vinculada única, cada node de la lista tiene dos componentes, el valor real del node y la referencia al siguiente node de la lista vinculada. En la lista doblemente enlazada, cada node tiene tres componentes: el valor del node, la referencia al node anterior y la referencia al siguiente node.
- d. HTML
es el componente más básico de la Web. Define el significado y la estructura del contenido web.



Figura 1. Interfaz de inicio individual.

Fuente: elaboración propia.

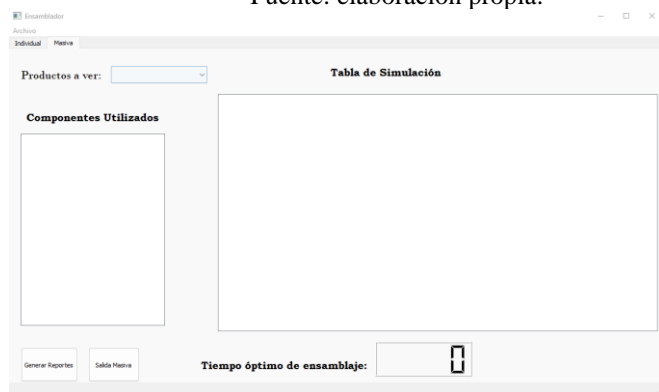


Figura 2. Interfaz inicio masiva

Fuente: elaboración propia.

maquina.xml

```
<Maquina>
  <CantidadLineasProduccion>
    <!--[0<NumeroEntero<1000]-->
  </CantidadLineasProduccion>
  <ListadoLineasProduccion>
    <LineaProduccion>
      <Numero>
        <!--[NumeroEntero>0]-->
      </Numero>
      <CantidadComponentes>
        <!--[0<NumeroEntero<1000]--> *Componentes de la línea
      </CantidadComponentes>
      <TiempoEnsamblaje>
        <!--[NumeroEntero>0]--> *Representa el tiempo en segundos
        que toma ensamblar en la línea
      </TiempoEnsamblaje>
    </LineaProduccion>
  </ListadoLineasProduccion>
  <ListadoProductos>
    <Producto>
      <nombre>
        <!--[Texto]--> *Nombre del Producto
      </nombre>
      <elaboracion> *Corresponde a instrucciones para ensamblar el producto
        <!--[Texto con formato L1pC1p L2pC2p L3pC3p ... LnpCnp]-->
      </elaboracion>
    </Producto>
  </ListadoProductos>
</Maquina>
```

Figura 3. Archivo de entrada maquina XML

Fuente: Instrucciones del proyecto

simulación_N.xml

```
<Simulacion>
  <Nombre>
    <!--[Texto]-->
  </Nombre>
  <ListadoProductos>
    <Producto>
      <!--[Texto]--> *Nombre de producto configurado en la máquina
    </Producto>
    <Producto>
      <!--[Texto]2-->
    </Producto>
  </ListadoProductos>
</Simulacion>
```

Figura 4. Archivo de entrada simulación XML

Fuente: Instrucciones del proyecto

```
<SalidaSimulacion>
  <Nombre>
    <!--[Texto]--> *Nombre de la simulación que generó esta salida.
  </Nombre>
  <ListadoProductos>
    <Producto>
      <Nombre>
        <!--[Texto]-->
      </Nombre>
      <TiempoTotal><!--[NumeroEntero>0]--></TiempoTotal>
      <ElaboracionOptima>
        <Tiempo NoSegundo="[NumeroEntero]">
          <LineaEnsamblaje NoLinea="[NumeroEntero]">
            <!--[Texto]--> *Representa la acción en la línea (movimiento,
            no hacer nada ó ensamblar)
          </LineaEnsamblaje>
        </Tiempo>
      </ElaboracionOptima>
    </Producto>
  </ListadoProductos>
</SalidaSimulacion>
```

Figura 5. Archivo de salida maquina/simulación XML

Fuente: Instrucciones del proyecto

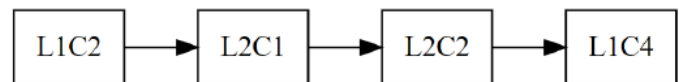


Figura 6. Cola de secuencia

Fuente: Instrucciones del proyecto

Tabla de Ensamble deiWatch													
Tiempo Optimo del Ensamble de iWatch: 98													
Tiempo en segundos	Línea 1	Línea 2	Línea 3	Línea 4	Línea 5	Línea 6	Línea 7	Línea 8	Línea 9	Línea 10	Línea 11	Línea 12	Línea 13
1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1	Se mueve a componente 1
2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2	Se mueve a componente 2
3	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3	No hace nada	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3	Se mueve a componente 3
4	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4	No hace nada	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4	Se mueve a componente 4
5	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5	No hace nada	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5	Se mueve a componente 5
6	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6	No hace nada	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6	Se mueve a componente 6

Figura 7. Reporte HTML del producto

Fuente: elaboración propia

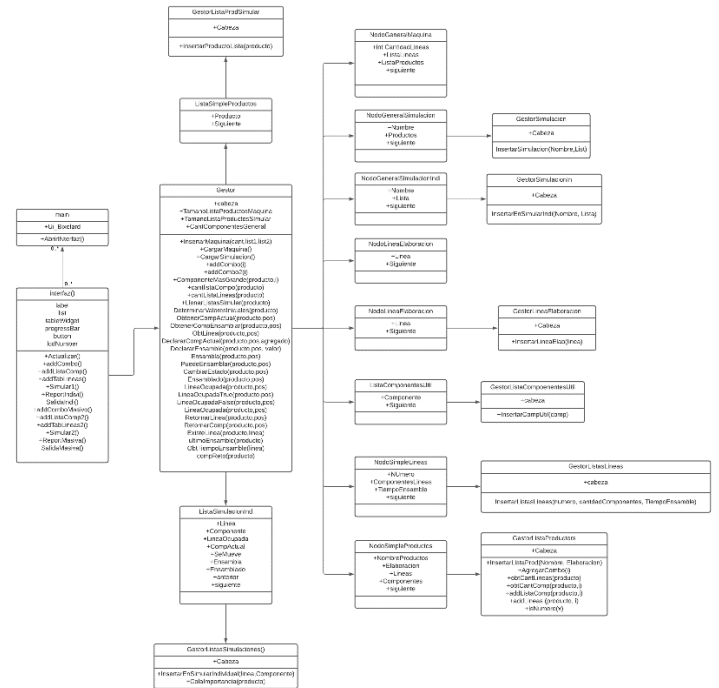


Figura 9. Diagrama de clases (se encuentra en carpeta documentación)

Fuente: elaboración propia

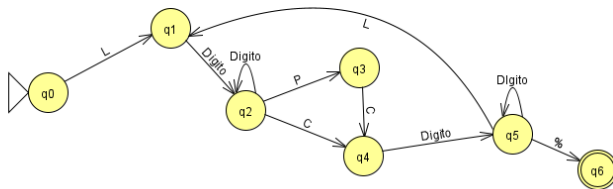


Figura 8. Autómata Cola Ensamble

Fuente: elaboración propia

Conclusiones

Los archivos XML, son una herramienta muy útil para guardar diferentes tipos de datos, desde juegos hasta información de alguna empresa. La correcta manipulación de estos puede tener muchos beneficios.

Los Tipos de Datos Abstractos permiten manipular y almacenar n cantidad de datos, ofreciendo una fácil administración como modificación de estos mismos.

La lista doblemente enlazada es una herramienta muy útil cuando un dato depende de un dato de un nodo anterior.

Referencias bibliográficas

- Anonimo, (2021). *Extensible Markup Language. Disponible en:* https://es.wikipedia.org/wiki/Extensible_Markup_Language
- Anonimo, (2021). *Tipo de dato abstracto. Disponible en:* https://es.wikipedia.org/wiki/Tipo_de_dato_abstracto
- Anonimo, (2021). *Tipo de dato abstracto. Disponible en:* [https://es.wikipedia.org/wiki/Matriz_\(matem%C3%A1ticas\)](https://es.wikipedia.org/wiki/Matriz_(matem%C3%A1ticas))