Trabajo Final Algoritmos y Estructura de datos - WASC

Alumno: Josue Cartagena u201415040

Glosario:

1. *Uso del conocimiento:*

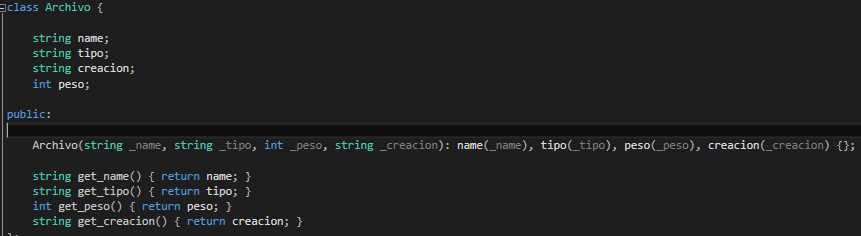
Para la realización del trabajo final se usará de árboles AVL, también de templates, recursividad, lambdas y de POO además de librerías como <filesystem> o <vector> entre otras.

1. *Identifica el problema:*

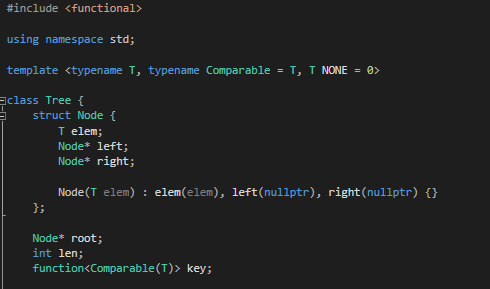
Es problema propuesto consiste en el desarrollo de un explorador de archivos para los sistemas operativos Windows o Linux, el cual debe de estar indexado y organizado, además de poder tener como funciones la de buscar y filtrar de acuerdo a un parámetro ya sea por nombre, por fecha de creación, cantidad de bytes, o su extensión.

1. *Idear:*

Para la búsqueda de archivos se usará la librería <filesystem> la cual nos permitirá obtener lo archivos de un directorio, además de ello se usarán 4 árboles para poder indexarlos de acuerdo a los parámetros brindados y también se usará POO para poder relacionar las clases que se generaran como la clase Archivo.h y la clase Arbol.h. Se usará lambdas para poder comparar los elementos en la búsqueda y template para poder acceder con mas facilidad a la clase Arbol.h y no generar 4 diferentes clases por cada parámetro.

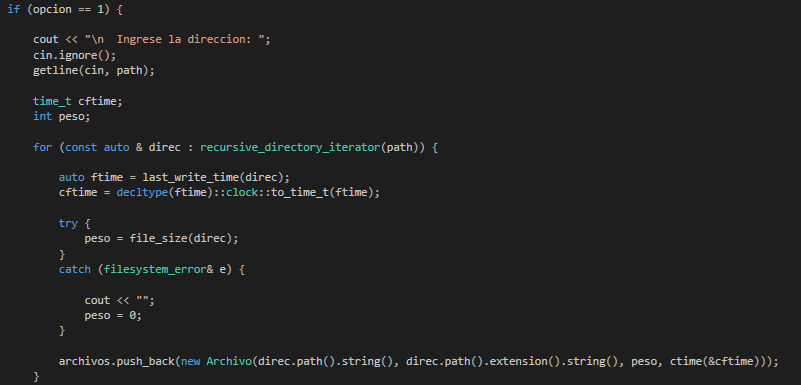
Clase Archivo.h:

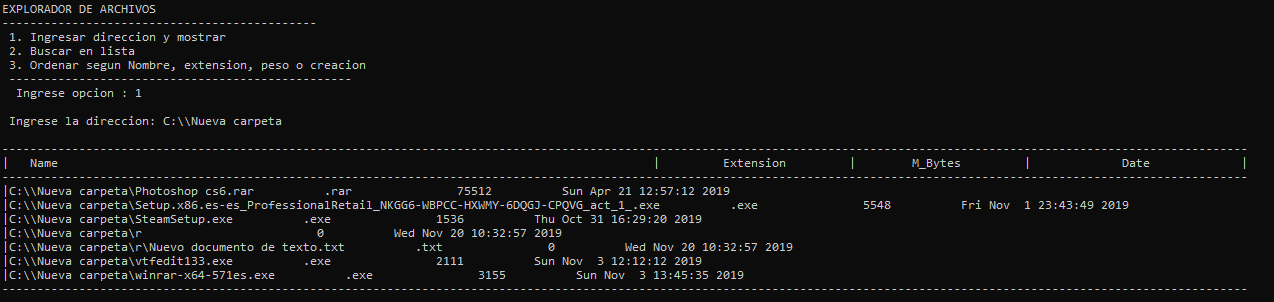
Clase Tree.h

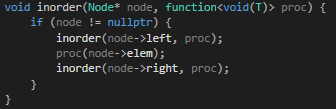


1. *Propuesta:*

Se creará un software tipo consola en el cual al momento de ejecutar el programa nos mostrará un menú el cual nos dará 3 opciones, indicar el directorio, buscar en la colección y filtrar, para ello se generará un vector Archivos de la clase Archivo.h y cuando se selecciona la primera opción y se ingresan valores tipo Archivo.h el cual obtiene los datos del directorio en string para poder manejar de forma fluida los archivos



Una ves que se termina de cargar los valores del vector Archivos, este es enviado a 4 arboles y se ordenara la lista para después mostrar la colección de archivos

Después se usará el método de la clase Tree.h para poder buscar/comparar el elemento que se desee buscar usando lambdas y con el método inorder se filtrará de acuerdo a los parámetros brindados haciendo uso de la recursividad.

