

MEMORIA PROYECTO BUSCADOR
AED1 Nov, 2024
Ñ\201ngel Ruiz Fernandez
Carla Ramos
G2.2 B117

MEMORIA

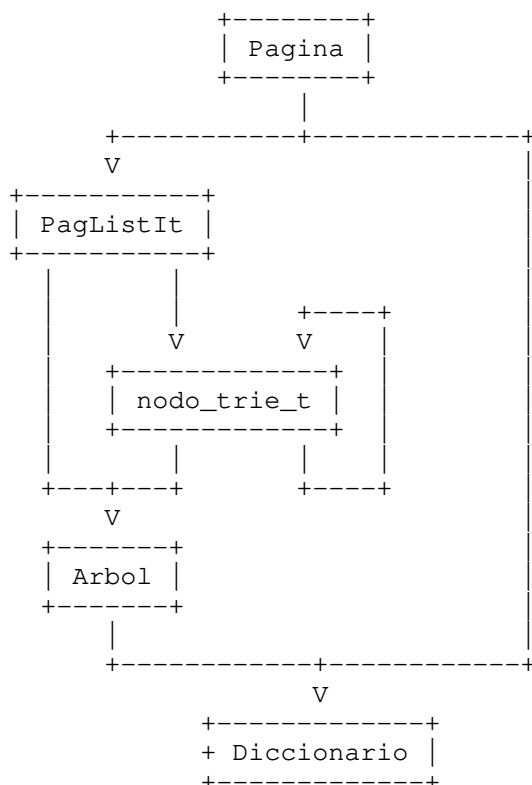
1. Analisis del programa
 - 1.1. Clases
 - 1.2. Modulos
 - 1.3. Makefile
 - 1.4. Normalizaci3n
 - 1.5. Tabla de dispersi3n
 - 1.5.1. Tipo
 - 1.5.2. Funci3n de dispersi3n - comparativa
 - 1.5.3. Liberar (spoiler: est3; en el stack, RAI)
 - 1.6. Arbol
 - 1.6.1. Tipo
 - 1.6.2. Defini3n de arbol y nodo
 - 1.6.3. Referencia a paginas
 - 1.6.3. (AVL - no)
 - 1.6.5. Liberar
 - 1.7. Globales (no)
 - 1.8. ChatGPT (no)
2. Listado del c3digo (adjunto - documento siguiente)
3. Informe de desarrollo
4. Conclusiones y valoraciones personales

1. Analisis del programa

1.1. Clases

- class Pagina
Representa una pagina, almacena su url, titulo, relevancia y contenido.
- class PagListIt : public std::list<Pagina>::iterator
Usa Pagina
Iterador heredado que implementa el operador '<' para poder ser ordenado en un contenedor ordenado.
Representa una referencia a elemento de std::list<Pagina>
- struct nodo_trie_t
Usa PagListIt
Representa un nodo del arbol trie de palabras. De este cuelgan hijos en un diccionario <char, nodo_trie_t>. Relaciona con una lista de referencias a pagina (PagListIt).
- class Arbol
Usa PagListIt y nodo_trie_t
Contiene la estructura arbol oculta, y posibles operaciones sobre el.
- class Diccionario
Usa Pagina, PagListIt y Arbol
Contiene la estructura de tabla de dispersi3n (std::list<Pagina>[N]), su funci3n de hash, y una instancia de Arbol.
Expone las posibles operaciones sobre la tabla, adem3s de pasar las operaciones del arbol.

Diagrama de clases



1.2. Modulos

Cada modulo (menos main) tiene un header asociado

- `diccionario.hpp`
Contiene la declaraci3n de todas las clases y structs, interfaz para ser usada por el interprete.

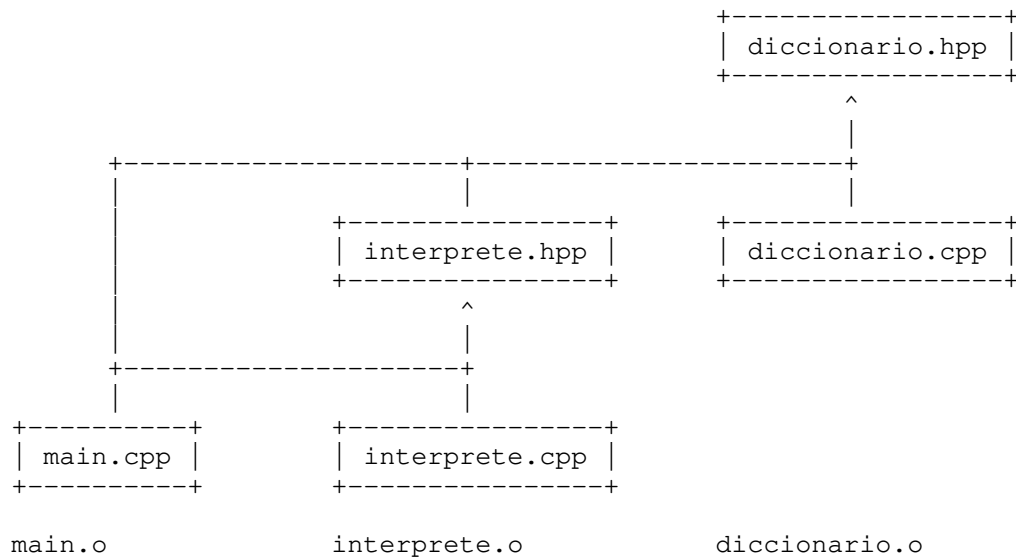
- `diccionario.cpp`
Contiene la definici3n de todos los metodos de las clases, que especifican la estructura de datos y las operaciones asociadas de la base de datos.

- `interprete.hpp`
Contiene la declaraci3n de las funciones que se encargan de interpretar los comandos de la entrada.

- `interprete.cpp`
Contiene la definici3n de las funciones de interpretaci3n de comandos, que llaman a operaciones sobre el diccionario, que se le es pasado por referencia.

- `main.cpp`
Contiene el bucle principal del programa, que lee comandos y llama al interprete. Es propietario de la instanciaci3n del diccionario, donde se almacenan todos los datos de la aplicaci3n.

Diagrama de modulos y unidades de compilaci3n



1.3. Makefile