

Línea 7: Se incluyen las librerías necesarias para el programa.

Línea 10: Se define la estructura Data que representa una palabra y su contador.

Línea 14: Se define la estructura Nodo que representa un nodo en el árbol binario de búsqueda.

Línea 18: Se declaran los prototipos de las funciones leerTeclado() y leerArchivo().

Línea 20: La función principal main() comienza.

Línea 22: Se declara la variable option para almacenar la opción elegida por el usuario.

Línea 24: Se inicia un bucle while que se repetirá hasta que la opción sea 3 (salir).

Línea 27: Se muestra el menú al usuario utilizando la función cout.

Línea 33: Se lee la opción ingresada por el usuario utilizando la función cin.

Línea 35: Se utiliza una estructura de control switch para ejecutar el código correspondiente a la opción seleccionada.

Línea 39: Si la opción es 1, se llama a la función leerArchivo().

Línea 41: Si la opción es 2, se llama a la función leerTeclado().

Línea 43: Si la opción es 3, se muestra un mensaje de despedida y se sale del bucle.

Línea 45: Si la opción no coincide con ninguna de las anteriores, se muestra un mensaje de opción inválida.

Línea 48: Se cierra el bucle while.

Línea 50: Se retorna 0 para indicar que la ejecución del programa ha sido exitosa.

Línea 54: La función agregar_palabra() comienza.

Línea 56: Si el puntero raíz es nulo, se crea un nuevo nodo y se asigna la palabra y el contador.

Línea 59: Si la palabra es igual a la palabra del nodo actual, se incrementa el contador.

Línea 62: Si la palabra es menor que la palabra del nodo actual, se llama recursivamente a la función con el subárbol izquierdo.

Línea 64: Si la palabra es mayor que la palabra del nodo actual, se llama recursivamente a la función con el subárbol derecho.

Línea 68: La función contar_palabras_diferentes() comienza.

Línea 70: Si el nodo raíz es nulo, se retorna 0.

Línea 72: Se retorna la suma de 1 más el resultado de llamar recursivamente a la función con el subárbol izquierdo y el subárbol derecho.

Línea 76: La función imprimir_arbol() comienza.

Línea 78: Si el nodo raíz no es nulo, se realiza un recorrido en orden del árbol.

Línea 79: Se muestra la palabra y su contador asociado en el nodo actual.

Línea 80: Se llama recursivamente a la función con el subárbol izquierdo.

Línea 81: Se llama recursivamente a la función con el subárbol derecho.

Línea 85: La función liberar_arbol() comienza.

Línea 87: Si el nodo raíz no es nulo, se realiza un recorrido postorden del árbol.

Línea 88: Se llama recursivamente a la función con el subárbol izquierdo.

Línea 89: Se llama recursivamente a la función con el subárbol derecho.

Línea 90: Se libera la memoria asignada para el nodo actual.

Línea 95: La función leerTeclado() comienza.

Línea 97: Se declara el puntero arbol para representar la raíz del árbol binario de búsqueda.

Línea 98: Se declara el arreglo palabra para almacenar las palabras ingresadas por el usuario.

Línea 99: Se declaran variables para almacenar el total de palabras y el total de palabras diferentes.

Línea 101: Se inicia un bucle while que se repetirá hasta que se ingrese el carácter '!'.
 !'.

Línea 102: Se muestra un mensaje para que el usuario ingrese una palabra.

Línea 103: Se lee la palabra ingresada por el usuario.

Línea 105: Si la palabra es igual a '!', se muestra un mensaje de finalización y se realizan las acciones correspondientes.

Línea 109: Se incrementa el contador de total de palabras.

Línea 112: Se convierte la palabra a minúsculas utilizando la función tolower().

Línea 113: Se llama a la función agregar_palabra() para agregar la palabra al árbol.

Línea 118: La función leerArchivo() comienza.

Línea 120: Se declara el puntero arbol para representar la raíz del árbol binario de búsqueda.

Línea 121: Se declara el arreglo palabra para almacenar las palabras leídas desde el archivo.

Línea 122: Se declaran variables para almacenar el total de palabras y el total de palabras diferentes.

Línea 124: Se abre el archivo "data.txt" en modo lectura.

Línea 127: Si el archivo es nulo, se muestra un mensaje de error y se retorna 1.

Línea 130: Se inicia un bucle while que se repetirá hasta que se alcance el final del archivo.

Línea 132: Se incrementa el contador de total de palabras.

Línea 135: Se convierte la palabra a minúsculas utilizando la función tolower().

Línea 136: Se llama a la función agregar_palabra() para agregar la palabra al árbol.

Línea 141: Se cierra el archivo.

Línea 143: Se muestra la frecuencia de las palabras en el árbol utilizando la función imprimir_arbol().

Línea 144: Se calcula el total de palabras diferentes utilizando la función contar_palabras_diferentes().

Línea 146: Se muestra el total de palabras y el total de palabras diferentes.

Línea 147: Se libera la memoria asignada para el árbol utilizando la función liberar_arbol().

Línea 151: Se retorna 0 para indicar que la ejecución de la función ha sido exitosa.