Source code: https://github.com/andreibanciu11/University-Projects/tree/main/Year%20III/FLCD/Lab/Lab

The SymbolTable class represents a symbol table that stores characters and their associated index values. It uses a hashtable to efficiently store and retrieve characters, handling collisions using lists.

public class SymbolTable { private Hashtable<Character, List<KeyValuePair» hashtable; // Internal hashtable to store key-value pairs private int tableSize; // Size of the hashtable private int newIndex = 0; // Index for character values

//Constructs a SymbolTable with the specified size and initializes it with characters from the input string.

public SymbolTable(int tableSize, String input)...

//Splits the input string into characters and stores them in the symbol table. //@param input The input string to split and store.

public void splitAndStore(String input)...

//Inserts a character into the symbol table, handling collisions by checking for duplicates. //@param key The character to insert.

public void put(char key)

//Searches for a character in the symbol table and returns its associated index value. //@param key The character to search for. //@return A string representation of the key-value pair if found, or ”Key not found!” if not found.

public String search(char key)

//Prints the contents of the symbol table, displaying buckets with their associated key-value pairs.

public void printAll() }