CS575: Final Project Report

Project Title: Boggle Game using Tries and Rabin – Karp algrtihm

Team Member(s): Shiva Prasad S – B00768637

I. Problem

To implement an improvised version of the boggle game by using Tries and Rabin – Karp algorithm.

II. Algorithms

In this application, there are two major algorithms used. Tries and Rabin-Karp. The tries algorithm is used to create trees of the string characters and aggregates all possible words in the board. The Rabin – karp algorithm will use this words as references and check whether the word input from the player matches with any of them.

A. Tries

In computer science, a **trie**, also called digital tree, radix tree or prefix tree, is a kind of search tree - an ordered tree data structure used to store a dynamic set or associative array where the keys are usually strings.Unlike a binary search tree, no node in the tree stores the key associated with that node; instead, its position in the tree defines the key with which it is associated. All the descendants of a node have a common prefix of the string associated with that node, and the root is associated with the empty string. Keys tend to be associated with leaves, though some inner nodes may correspond to keys of interest. Hence, keys are not necessarily associated with every node.

B. Rabin – Karp

In computer science, the Rabin–Karp algorithm or Karp–Rabin algorithm is a string-searching algorithm created by Richard M. Karp and Michael O. Rabin (1987) that uses hashing to find any one of a set of pattern strings in a text. For text of length n and p patterns of combined length m, its average and best case running time is O(n+m) in space O(p), but its worst-case time is O(nm). In contrast, the Aho–Corasick string-matching algorithm has asymptotic worst-time complexity O(n+m) in space O(m).A practical application of the algorithm is detecting plagiarism. Given source material, the algorithm can rapidly search through a paper for instances of sentences from the source material, ignoring details such as case and punctuation.

III. Software Design and Implementation

A. Software Design

The application is divided into two components : A user interface and a java program. The user interface will have the game board where there will be jumbled alphabets displayed and a text box to accept the user input. The Java program will then process these inputs, perform algorithmic operations and send back the output to user interface.

B. Implementation and Tools Used

The project is implemented in Java programming language.. The Java program is coded and compiled in Eclipse IDE.

Java is a general-purpose programming language that is class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to "bytecode" that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but it has fewer low-level facilities than either of them.

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins, including Ada, ABAP, C, C++, C#, Clojure, COBOL, D, Erlang, Fortran, Groovy, Haskell, JavaScript, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Rust, Scala, and Scheme. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++, and Eclipse PDT for PHP, among others.

The output of the Java program is passed appropriately and they are displayed in the console of Eclipse IDE.

Source Code

https://github.com/shiva96648/Boggle-Game-using-Trie

References

[1] https://en.wikipedia.org/wiki/Boggle.

[2] https://en.wikipedia.org/wiki/Rabin%E2%80%93Karp_algorithm

[3] https://www.geeksforgeeks.org/boggle-set-2-using-trie/

[4] https://stackoverflow.com/questions/11607270/how-to-check-whether-given-string-is-a-word