CHAPTER V
IMPLEMENTATION AND TESTING

5.1 Implementation

5.1.1 GADDAG Pattern

On this project using the Java programming language. When the program start, first time program will read dictionary to store on the data structure. While program read each word on dictionary, each word will form into GADDAG pattern. On the GADDAG algorithm, a word will be formed into some form that has a pattern (Pattern GADDAG can be seen in Figure 3). Here is the code to generate the pattern.

```java
public String[] generate(String str2){
    String str = str2.toUpperCase();
    String[] result = new String[str.length()];
    String depan = "\",belakang = "\";
    int prefix = 1;

    for(int i=0;i<str.length();i++){
        belakang = str.substring(prefix);
        depan = str.charAt(prefix-1) + depan;
        result[i] = depan + belakang;
        System.out.println(result[i]);
        trie.addWord(result[i]);
        System.out.println("====");

        prefix++;
    }
    return result;
}
```

*Figure 12: Code to establish a pattern of the GADDAG*

5.1.2 Insert Trie

After find the GADDAG pattern of a word then each pattern will be insert into the data trie structure. Look figure below for the insert word to trie data structure.
The example is if program want insert “WORD” to the data structure look on figure 14. For easily imagine the trie structure look at the memory adders from log. Program will search child of root where valued “W”. If found the root change into child which valued “W”. After that “O” if not found program will create new node valued “O” as a child then root become that child and soon.
5.1. Search Word

There are 2 main procedures in the GADDAG algorithm, namely Gen and Goon. The function procedure Gen is put up each letter on the rack to check on the procedure Goon. Goon is used to check the letter to the data structure is a letter that can be formed a word. These 2 procedures are run recursively. Below are the logs of searching words where “OR” on the board and “WD” rack from the player.

```plaintext
ANKOR SQUAR 0
framebox: gendag,Gen(0,'1',W,[100E600,H,false,1]),[x:y,7,yParent:7,xParent:7,H:1true]);
lock b 0 level 0 paper 7,0 D ischemic dgo d W POS:0
1 hof [700006,F,false,2] Data dishang at 7,0 karena ada dl [700006,0,False,1]
checkprocesstingword = false
lock b 0 level 2 paper 7,1 dishang dgo d W POS:1
1 hof 0 '1'okets dgo d dl 7,1 karena mul dl [2a7c32,F,false,2]
2 RECORD
Cek harf d dl dari parent
3 hof [720090,F,false,3] Data dishang at 7,1 karena ada dl [2a7c32,Ffalse,2]
Posit [7,0]
checkprocesstingword = false
Memori 0 dl [720090,Ffalse,2]
3 hof [789009,Ffalse,2] Data dishang at 7,0 karena ada dl [720090,Ffalse,2]
checkprocesstingword = false
Lock b 0 level 0 paper 7,1 dishang dgo d W POS:0
1 hof [700006,F,false,2] Data dishang at 7,1 karena mul dl [700006,Ffalse,2]
lock harf [789009,Ffalse,2] Data dishang at 7,1 karena ada dl [789009,Ffalse,2]
cek harf.Promer of word: WD
Anchor square: [x:y,7+yParent:7,xParent:7,H:1true]
Paper 0 0 dan 6 kosong naka hasil true
Y: 714: 0
Paper 0 0 dan 6,8 kosong naka hasil true
Y: 714: 9
Paper 0 0 dan 0,9 kosong naka hasil true
Lendet builder Prober of true
Cek JALUR: 7,0,4,[x:y,7+yParent:7,xParent:7,H:1true],Paper 7,3 Kosong
getbonusvalue=9800
PAPAN 0 7,8:0:0:0 BONUS
PAPAN 0 7,8:0:0:0 BONUS
TEND: getbonusvalue=
V2 = 0 7,8:0:0:0 BONUS
Hakti Cek Jalur: true
checkprocesstingword = false
Lock b 0 level 6 paper 7,10 dishang dgo d W POS:1
```

5.2 Testing

Testing will be done in several conditions, there are add behind the word, add in front of the word, and both. The first experiment is adding a letter behind the word. Examples on the board are the letters G and O, and the letter in hand are N and E. The following are examples of conditions on the board on figure 16.
The search process will begin with a word search for the coordinates of the board to be mounted letter (Anchor Square). In each of these coordinates the program will find all the possible words that can be formed. Here is the result of conditions the board.

Figure 16: Gone Example

Figure 17: Gone Example 2
After the program is finished searching for all the words, the program will display all words are formed from each coordinate. In the above example the coordinates of the line 7 columns 6 has the highest value. After that program will update the condition of board, look at figure 18.

The second experiment are when put behind of a word on the board. In this example, the board has the letters O and R. Then the letter in hand O and P. Here is a picture of the condition of the board.

Figure 18: Result Gone

Figure 19: Poor Example
Just like the previous process, the program will first find the coordinates that might be to set a letter. On the boards above conditions produce results as shown below.

On the third experiment, GADDAG algorithm can also pair the letter in front of and behind the words on the board.

Below is the result for figure 22 condition
The last experiment is cross the word with other word. On the condition of the board over the letter which is owned by the I and L look at figure 23.

Figure 22: Result Rise Example

Figure 23: Cross Example
In the above conditions the program generates word "FILL" which is in line with the word "RISE". This result is correct because the letter I beside with letter S and form a word of "IS".

*Figure 24: Result Cross Example*