

LAPORAN TUGAS KECIL 1

IF2211 STRATEGI ALGORITMA



Disusun oleh:

Hansel Valentino Tanoto

13520046

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

2022

DAFTAR ISI

| | |
|---------------------------------------|----|
| DAFTAR ISI..... | 2 |
| A. ALGORITMA <i>BRUTE FORCE</i> | 3 |
| B. <i>SOURCE CODE</i> | 6 |
| C. <i>SCREENSHOTS</i> | 14 |
| D. TAUTAN KODE PROGRAM..... | 29 |
| E. <i>CHECKBOX</i> | 29 |

A. ALGORITMA *BRUTE FORCE*

Algoritma *Brute Force* merupakan suatu pendekatan yang *straightforward* dalam penyelesaian masalah sehingga algoritma cenderung bersifat sederhana dan jelas (*obvious*). Hampir seluruh permasalahan pasti memiliki solusi dalam bentuk algoritma *Brute Force*, termasuk permasalahan *word search puzzle*. *Word search puzzle* adalah suatu teka teki permainan kata dengan tujuan mencari kata-kata yang tersembunyi dalam kumpulan huruf acak pada suatu papan berbentuk segi empat. Kata-kata tersebut dapat ditemukan dalam 8 arah pembacaan yaitu ke kanan, ke kiri, ke atas, ke bawah, diagonal ke kanan atas, diagonal ke kanan bawah, diagonal ke kiri atas, dan diagonal ke kiri bawah. Dalam laporan ini akan dibahas algoritma *brute force* yang dibuat dalam bahasa pemrograman Java untuk menyelesaikan *puzzle* tersebut.

Secara sederhana, algoritma program ini dimulai dengan pembacaan *input file* dalam format .txt yang berisi konfigurasi papan *puzzle* dan *list* kata-kata yang harus dicari. Hasil dari pembacaan file tersebut adalah dimensi papan *puzzle*, jumlah kata, matriks huruf (representasi papan *puzzle*), dan *list* kata. Lalu papan *puzzle* dan *list* kata akan ditampilkan ke layar. Kemudian, akan dilakukan traversal pada *list* kata dan untuk setiap kata akan dilakukan traversal lagi terhadap setiap elemen matriks huruf. Dalam proses traversal tersebut, akan dilakukan *string matching* pada setiap titik / elemen matriks yang dikunjungi dalam 8 arah seperti yang telah disebutkan sebelumnya. Terdapat sedikit optimasi (metode heuristik) yang digunakan dalam algoritma *string matching* tersebut yaitu apabila panjang teks pada papan *puzzle* dalam suatu arah lebih kecil dari panjang kata yang dicari, maka *string matching* akan langsung dihentikan atau dilewati (*skip*).

Cara kerja algoritma *string matching* (sama untuk setiap arah) pada program ini yaitu pertama-tama akan dilakukan pengecekan terhadap panjang teks untuk optimasi seperti yang sudah dijelaskan sebelumnya. Yang dimaksud dengan teks di sini adalah kumpulan huruf yang dibaca dimulai dari titik saat itu ke suatu arah tertentu pada papan *puzzle*. Setelah itu akan dijalankan *loop* untuk mencocokkan satu per satu huruf-huruf pada kata yang dicari dengan huruf-huruf pada papan *puzzle*. Selama *loop* ini, akan dilakukan pencacahan untuk menghitung berapa huruf yang sudah cocok (*matched*). *Loop* ini akan berhenti apabila ditemukan ketidakcocokan huruf. Apabila pada akhir *loop*, nilai pencacah sama dengan panjang kata yang dicari, maka disimpulkan kata tersebut sudah ditemukan. Terdapat suatu matriks yang digunakan untuk menandai pada posisi mana di papan *puzzle*, sebuah kata ditemukan, yang elemennya akan diisi dengan indeks dari *array/list* warna (kode ANSI) apabila kata ditemukan sesuai kondisi yang dijelaskan pada kalimat sebelumnya. Apabila tidak, artinya kata belum ditemukan dan proses *string matching* akan dilanjutkan pada elemen matriks huruf berikutnya. Proses *string matching* untuk setiap kata yang dicari akan otomatis berhenti apabila kata sudah ditemukan sehingga apabila kata sudah ditemukan pada suatu arah, tidak perlu dilakukan pengecekan lagi pada arah yang lain. Setelah semua kata ditemukan, akan ditampilkan solusinya berupa papan *puzzle* yang sudah diwarnai pada bagian kata-katanya. Selain itu, ditampilkan pula statistik dari proses pencarian tersebut berupa waktu proses dan jumlah perbandingan (komparasi) baik untuk setiap kata maupun secara keseluruhan (total). Waktu eksekusi untuk setiap kata dihitung sejak dilakukan komparasi huruf pertama untuk dan berakhir ketika kata sudah ditemukan atau semua elemen matriks huruf sudah dilakukan pengecekan. Sedangkan jumlah komparasi dicacah setiap ada proses perbandingan huruf.

Dalam program ini terdapat 15 buah *method* yang dibuat untuk menjalankan algoritma di atas yaitu:

1. `public static void main (String[] args) throws Exception`

Method ini merupakan algoritma program utama yang akan dijalankan saat *file* program dieksekusi. Semua *method* di bawah akan dipanggil dan digunakan di sini. Variabel-variabel penting yang digunakan dalam *method* ini adalah *dim* (*array* untuk menyimpan nilai *row*, *col*, dan *wCount*), *row* (baris pada papan *puzzle*), *col* (kolom pada papan *puzzle*), *wCount* (jumlah kata pada *list* kata), *puzzle* (matriks huruf sebagai representasi papan *puzzle*), *words* (*array* dengan elemen berupa semua kata yang dicari), *solved* (matriks yang berisi kode warna untuk menentukan lokasi kata-kata yang ditemukan), *solvingTime* (*array* yang menyimpan data waktu eksekusi setiap kata), dan *comparison* (*array* yang menyimpan data jumlah perbandingan huruf untuk setiap kata).

2. `public static int[] dimension (String path) throws Exception`
Pembacaan *file* konfigurasi (.txt) untuk mengetahui dimensi papan *puzzle* (baris dan kolom) dan jumlah huruf yang ingin dicari.
3. `public static void readFile(String path, char[][] puzzle, String[] words) throws Exception`
Pembacaan *file* konfigurasi (.txt) untuk menyimpan data *puzzle* sebagai matriks huruf dan *list* kata.
4. `public static void writePuzzle (char[][] puzzle)`
Menampilkan *puzzle* yang akan dipecahkan dengan program ini.
5. `public static void writeWords (String[] words)`
Menampilkan *list* kata yang akan dicari pada papan *puzzle*.
6. `public static void writeSolved (char[][] puzzle, int[][] solved)`
Menampilkan solusi teka-teki berupa papan *puzzle* yang sudah diberi warna pada lokasi-lokasi kata ditemukan.
7. `public static void writeStat (String[] words, long[] solvingTime, int[] comparison)`
Menampilkan statistik algoritma berupa waktu proses dan jumlah perbandingan (komparasi).
8. `public static boolean down (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`
Melakukan *string matching* ke arah bawah.
9. `public static boolean up (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`
Melakukan *string matching* ke arah atas.
10. `public static boolean right (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`
Melakukan *string matching* ke arah kanan.
11. `public static boolean left (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`
Melakukan *string matching* ke arah kiri.
12. `public static boolean upright (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`
Melakukan *string matching* ke arah diagonal kanan atas.
13. `public static boolean upleft (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`
Melakukan *string matching* ke arah diagonal kiri atas.
14. `public static boolean downright (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`
Melakukan *string matching* ke arah diagonal kanan bawah.
15. `public static boolean downleft (char[][] puzzle, String word, int x, int y, int[][] solved, int colorId, int[] count)`

Melakukan *string matching* ke arah diagonal kiri bawah.

Pada *method* ke-8 (*down*) sampai ke-15 (*downleft*) variable yang menjadi parameternya memiliki kesamaan, variabel `colorId` menyimpan indeks dari *array/list* warna yang digunakan untuk menandakan bahwa kata tersebut, saat ditemukan, akan diwarnai dengan warna pada indeks tersebut. Variabel `count` menyimpan jumlah perbandingan yang dilakukan. Sedangkan variable `x` dan `y` menandakan posisi dimulainya awal pengecekan kata pada saat itu dengan `x` sebagai baris dan `y` sebagai kolom.

B. SOURCE CODE

Program *word search puzzle* ini dibuat menggunakan bahasa pemrograman Java. Berikut adalah kode program lengkapnya:

```
import java.io.*;
import java.util.Scanner;
import java.util.Arrays;

public class WordSearchPuzzle_nonheuristic {

    // MAIN PROGRAM
    public static void main (String[] args) throws Exception{

        Scanner sc = new Scanner(System.in);
        System.out.printf("Enter file path: ");
        String path = sc.nextLine();

        // Dimensi "puzzle board" (row & column) dan jumlah kata yang dicari
        int[] dim = dimension(path);
        int row = dim[0];
        int col = dim[1];
        int wCount = dim[2];

        char[][] puzzle = new char[row][col];    // papan puzzle
        String[] words = new String[wCount];    // himpunan kata yang dicari
        int[][] solved = new int[row][col];    // hasil pemecahan puzzle
        long[] solvingTime = new long[wCount]; // waktu pemecahan puzzle
        int[] comparison = new int[wCount];    // jumlah perbandingan

        // Pembacaan file puzzle (.txt) dan menampilkannya ke layar
        readFile(path, puzzle, words);
        writePuzzle(puzzle);
        writeWords(words);

        // Pemecahan puzzle (string matching)
        for (int i = 0; i < wCount; i++) {
            int colorId = i%11 + 1;
            int[] compareCounter = {0};
            long startTime = System.nanoTime();
            letterTransversal:
            for (int j = 0; j < row; j++) {
                for (int k = 0; k < col; k++) {
                    if
(down(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
                        break letterTransversal;
                    }
                }
            }
        }
    }
}
```

```

        else if
(right(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
            break letterTransversal;
        }
        else if
(up(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
            break letterTransversal;
        }
        else if
(left(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
            break letterTransversal;
        }
        else if
(downright(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
            break letterTransversal;
        }
        else if
(downleft(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
            break letterTransversal;
        }
        else if
(upleft(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
            break letterTransversal;
        }
        else if
(upright(puzzle,words[i],j,k,solved,colorId,compareCounter)) {
            break letterTransversal;
        }
    }
    }
    solvingTime[i] = System.nanoTime() - startTime;
    comparison[i] = compareCounter[0];
}
// Menuliskan hasilnya
writeSolved(puzzle, solved);
writeStat(words, solvingTime, comparison);
System.out.println();

sc.close();
return;
}

// DIMENSION
public static int[] dimension (String path) throws Exception {
    // Menghitung dimensi papan puzzle dan jumlah kata yang dicari
    // Keterangan: result[0] = baris, result[1] = kolom, result[2] =
jumlah kata
    int[] result = new int[3];

```

```

        int separator = 0; // baris yang memisahkan papan puzzle dan list
kata pada file .txt
        Scanner sc = new Scanner(new BufferedReader(new FileReader(path)));

        result[1] = sc.nextLine().length()/2 + 1;
        int lineCount = 1;
        while (sc.hasNextLine()) {
            lineCount += 1;
            String line = sc.nextLine();
            if (line.equals("")) {
                separator = lineCount;
            }
        }
        result[0] = separator - 1;
        result[2] = lineCount - separator;

        sc.close();
        return result;
    }

    // COLOR
    public static final String ANSI_RESET    = "\u001B[0m";
    public static final String ITALIC        = "\u001B[3m";
    public static final String UNDERLINE     = "\u001B[4m";
    public static final String ANSI_COLOR[] = {"\u001B[37m", "\u001B[32m",
"\u001B[33m", "\u001B[34m", "\u001B[35m", "\u001B[36m", "\u001B[91m",
"\u001B[92m", "\u001B[93m", "\u001B[94m", "\u001B[95m", "\u001B[96m"};

    // READ AND WRITE
    public static void readFile(String path, char[][] puzzle, String[] words)
throws Exception {
        // Membaca file puzzle (.txt) dan mengkonversinya ke matriks atau
array
        String line;
        Scanner sc = new Scanner(new BufferedReader(new FileReader(path)));
        for (int i = 0; i < puzzle.length; i++) {
            line = sc.nextLine();
            for (int j = 0; j < line.length(); j++) {
                if (line.toCharArray()[j] != ' ') {
                    puzzle[i][j/2] = line.toCharArray()[j];
                }
            }
        }
        line = sc.nextLine();
        for (int k = 0; k < words.length; k++) {
            line = sc.nextLine();
            words[k] = line;
        }
    }

```



```

}

public static void writePuzzle (char[][] puzzle) {
    // Menuliskan papan puzzle ke layar
    System.out.println("\n" + "=".repeat(puzzle[0].length-13) + " WORD
SEARCH PUZZLE (" + puzzle.length + "x" + puzzle[0].length +") " +
"=".repeat(puzzle[0].length-12));
    for (int i = 0; i < puzzle.length; i++) {
        System.out.printf("| ");
        for (int j = 0; j < puzzle[0].length; j++) {
            System.out.printf("%c ", puzzle[i][j]);
        }
        System.out.printf("| \n");
    }
    System.out.println("=".repeat(puzzle[0].length*2+3));
}

public static void writeWords (String[] words) {
    // Menuliskan list kata yang dicari ke layar
    System.out.println("\nWORDS LIST (" + words.length + " words):");
    for (int i = 0; i < words.length; i++) {
        System.out.println("- " + words[i]);
    }
}

public static void writeSolved (char[][] puzzle, int[][] solved) {
    // Menuliskan papan puzzle yang sudah berhasil diselesaikan
    System.out.println("\n" + "=".repeat(puzzle[0].length-4) + " SOLUTION
" + "=".repeat(puzzle[0].length-3));
    for (int i = 0; i < puzzle.length; i++) {
        System.out.printf("| ");
        for (int j = 0; j < puzzle[0].length; j++) {
            System.out.printf(ANSI_COLOR[solved[i][j]]);
            if (solved[i][j] != 0) {
                System.out.printf(ITALIC);
                System.out.printf(UNDERLINE);
            }
            System.out.printf("%c ", puzzle[i][j]);
            System.out.printf(ANSI_RESET);
        }
        System.out.printf("| \n");
    }
    System.out.println("=".repeat(puzzle[0].length*2+3));
}

public static void writeStat (String[] words, long[] solvingTime, int[]
comparison) {

```

```

        // Menuliskan statistik berupa waktu eksekusi dan jumlah komparasi
yang terjadi
        System.out.println("\n" + "=".repeat(31) + " STATISTICS " +
"=".repeat(31));
        for (int i = 0; i < words.length; i++) {
            System.out.printf("- " + words[i] + " ".repeat(15-
words[i].length()) + "||  ");
            String timeStat = "Time taken: " + solvingTime[i] + " ns";
            System.out.println(timeStat + " ".repeat(25-timeStat.length()) +
"- Comparison count: " + comparison[i]);
        }
        long totalTime = Arrays.stream(solvingTime).sum();
        int totalComparison = Arrays.stream(comparison).sum();
        System.out.println("=".repeat(74));
        System.out.println("Total Solving Time          = "+ totalTime + "
ns");
        System.out.println("Total Comparison          = "+ totalComparison);
        System.out.println("Average Solving Time      = "+
totalTime/words.length + " ns");
        System.out.println("Average Comparison Count  = "+
totalComparison/words.length);
    }

    // PROSEDUR STRING MATCHING
    public static boolean down (char[][] puzzle, String word, int x, int y,
int[][] solved, int colorId, int[] count) {
        // Mencari kata yang arahnya vertikal ke bawah
        int wordLen = word.length();
        if (puzzle.length-x >= wordLen) {
            int j = 0;
            count[0]++;
            while ((j < wordLen) && (puzzle[x+j][y] == word.charAt(j))) {
                j++;
            }
            count[0] += j;
            if (j == wordLen) {
                for (int k = x; k < x+wordLen; k++) {
                    solved[k][y] = colorId;
                }
                return true;
            }
        }
        return false;
    }

    public static boolean up (char[][] puzzle, String word, int x, int y,
int[][] solved, int colorId, int[] count) {
        // Mencari kata yang arahnya vertikal ke atas

```

```

        int wordLen = word.length();
        if (x+1 >= wordLen) {
            int j = 0;
            count[0]++;
            while ((j < wordLen) && (puzzle[x-j][y] == word.charAt(j))) {
                j++;
            }
            count[0] += j;
            if (j == wordLen) {
                for (int k = x; k > x-wordLen; k--) {
                    solved[k][y] = colorId;
                }
                return true;
            }
        }
        return false;
    }

    public static boolean right (char[][] puzzle, String word, int x, int y,
int[][] solved, int colorId, int[] count) {
        // Mencari kata yang arahnya horizontal ke kanan
        int wordLen = word.length();
        if (puzzle[0].length-y >= wordLen) {
            int j = 0;
            count[0]++;
            while ((j < wordLen) && (puzzle[x][y+j] == word.charAt(j))) {
                j++;
            }
            count[0] += j;
            if (j == wordLen) {
                for (int k = y; k < y+wordLen; k++) {
                    solved[x][k] = colorId;
                }
                return true;
            }
        }
        return false;
    }

    public static boolean left (char[][] puzzle, String word, int x, int y,
int[][] solved, int colorId, int[] count) {
        // Mencari kata yang arahnya horizontal ke kiri
        int wordLen = word.length();
        if (y+1 >= wordLen) {
            int j = 0;
            count[0]++;
            while ((j < wordLen) && (puzzle[x][y-j] == word.charAt(j))) {
                j++;
            }

```

```

    }
    count[0] += j;
    if (j == wordLen) {
        for (int k = y; k > y-wordLen; k--) {
            solved[x][k] = colorId;
        }
        return true;
    }
}
return false;
}

public static boolean downright (char[][] puzzle, String word, int x, int
y, int[][] solved, int colorId, int[] count) {
    // Mencari kata yang arahnya diagonal ke bawah kanan
    int wordLen = word.length();
    if (Math.min(puzzle.length-x,puzzle[0].length-y) >= wordLen) {
        int j = 0;
        count[0]++;
        while ((j < wordLen) && (puzzle[x+j][y+j] == word.charAt(j))) {
            j++;
        }
        count[0] += j;
        if (j == wordLen) {
            for (int k = y; k < y+wordLen; k++) {
                solved[x+k-y][k] = colorId;
            }
            return true;
        }
    }
    return false;
}

public static boolean upleft (char[][] puzzle, String word, int x, int y,
int[][] solved, int colorId, int[] count) {
    // Mencari kata yang arahnya diagonal ke kiri atas
    int wordLen = word.length();
    if (Math.min(x+1,y+1) >= wordLen) {
        int j = 0;
        count[0]++;
        while ((j < wordLen) && (puzzle[x-j][y-j] == word.charAt(j))) {
            j++;
        }
        count[0] += j;
        if (j == wordLen) {
            for (int k = y; k > y-wordLen; k--) {
                solved[x+k-y][k] = colorId;
            }
        }
    }
}

```

```

        return true;
    }
}
return false;
}

public static boolean upright (char[][] puzzle, String word, int x, int y,
int[][] solved, int colorId, int[] count) {
    // Mencari kata yang arahnya diagonal ke kanan atas
    int wordLen = word.length();
    if (Math.min(x+1,puzzle[0].length-y) >= wordLen) {
        int j = 0;
        count[0]++;
        while ((j < wordLen) && (puzzle[x-j][y+j] == word.charAt(j))) {
            j++;
        }
        count[0] += j;
        if (j == wordLen) {
            for (int k = y; k < y+wordLen; k++) {
                solved[x-k+y][k] = colorId;
            }
            return true;
        }
    }
    return false;
}

public static boolean downleft (char[][] puzzle, String word, int x, int
y, int[][] solved, int colorId, int[] count) {
    // Mencari kata yang arahnya diagonal ke kiri bawah
    int wordLen = word.length();
    if (Math.min(puzzle.length-x,y+1) >= wordLen) {
        int j = 0;
        count[0]++;
        while ((j < wordLen) && (puzzle[x+j][y-j] == word.charAt(j))) {
            j++;
        }
        count[0] += j;
        if (j == wordLen) {
            for (int k = y; k > y-wordLen; k--) {
                solved[x-k+y][k] = colorId;
            }
            return true;
        }
    }
    return false;
}
}

```

C. SCREENSHOTS

1. Test case puzzle ukuran kecil ke-1 (12x14)

```
Enter file path: ../test/test1-small.txt

= WORD SEARCH PUZZLE (12x14) ==
| Y C R B O I A X T M S Q R N |
| S W E H B X X M A F R J E O |
| U Y B J C T W Y S E R O Y V |
| I R M C K R N I B D S X R E |
| S A E W S K A O V A I E A M |
| F U C B U R T M U Y K H U B |
| H N E I M C F G C X M A R E |
| C A D R O E U J U N E H B R |
| J J Y T C S T L L Q Z R E J |
| H U W S T R N P N X T V F Z |
| W F L I R P A X E M K Y G B |
| T O A Y J H O J I S V H C O |
=====

WORDS LIST (12 words):
- FEBRUARY
- SEPTEMBER
- AUGUST
- JULY
- NOVEMBER
- OCTOBER
- DECEMBER
- APRIL
- MARCH
- JUNE
- MAY
- JANUARY

===== SOLUTION =====
| Y C R B O I A X T M S Q R N |
| S W E H B X X M A F R J E O |
| U Y B J C T W Y S E R O Y V |
| I R M C K R N I B D S X R E |
| S A E W S K A O V A I E A M |
| F U C B U R T M U Y K H U B |
| H N E I M C F G C X M A R E |
| C A D R O E U J U N E H B R |
| J J Y T C S T L L Q Z R E J |
| H U W S T R N P N X T V F Z |
| W F L I R P A X E M K Y G B |
| T O A Y J H O J I S V H C O |
=====

===== STATISTICS =====
- FEBRUARY      || Time taken: 240100 ns - Comparison count: 373
- SEPTEMBER     || Time taken: 105700 ns - Comparison count: 372
- AUGUST        || Time taken: 52600 ns  - Comparison count: 265
- JULY          || Time taken: 114300 ns - Comparison count: 673
- NOVEMBER      || Time taken: 10700 ns  - Comparison count: 48
- OCTOBER       || Time taken: 120100 ns - Comparison count: 360
- DECEMBER      || Time taken: 123000 ns - Comparison count: 258
- APRIL         || Time taken: 121900 ns - Comparison count: 754
- MARCH         || Time taken: 67000 ns  - Comparison count: 387
- JUNE          || Time taken: 94400 ns  - Comparison count: 612
- MAY           || Time taken: 9400 ns   - Comparison count: 53
- JANUARY       || Time taken: 89800 ns  - Comparison count: 396

=====
Total Solving Time      = 1149000 ns
Total Comparison        = 4551
Average Solving Time    = 95750 ns
Average Comparison Count = 379
```

2. Test case puzzle ukuran kecil ke-2 (14x16)

Enter file path: ../test/test2-small.txt

=== WORD SEARCH PUZZLE (14x16) ===

```
| U L P O F E X N I K M I S Q G V |
| O U Z H E D Z J J J A E V E R Y |
| T Q V T U F A B A T V Y G D E K |
| W O D F S K C G T E X A H D A A |
| E N R M R U I V A W T H I N K H |
| D W B Z S O J P C I N D R U V F |
| A R E S T D M R B F N W D O X J |
| L Z E T O Z G N I V I G Q R E R |
| I W L B P M I F F I O W S Z G N |
| R Y L B S O E B K U E Y E B Z Y |
| U J S I K R W S H T G V G D A T |
| Z A W G D W B U C F A W I W O O |
| U V N Q T V O D P Z X P G L F J |
| P E E V I G F H R K F O V P B Y |
```

WORDS LIST (14 words):

- LIVE
- AGAIN
- THINK
- GIVING
- HOW
- GIVE
- JUST
- SOME
- PUT
- ROUND
- EVERY
- STOP
- AS
- FROM

===== SOLUTION =====

```
| U L P O F E X N I K M I S Q G V |
| O U Z H E D Z J J J A E V E R Y |
| T Q V T U F A B A T V Y G D E K |
| W O D F S K C G T E X A H D A A |
| E N R M R U I V A W T H I N K H |
| D W B Z S O J P C I N D R U V F |
| A R E S T D M R B F N W D O X J |
| L Z E T O Z G N I V I G Q R E R |
| I W L B P M I F F I O W S Z G N |
| R Y L B S O E B K U E Y E B Z Y |
| U J S I K R W S H T G V G D A T |
| Z A W G D W B U C F A W I W O O |
| U V N Q T V O D P Z X P G L F J |
| P E E V I G F H R K F O V P B Y |
```

===== STATISTICS =====

| | | | | |
|----------|--|-----------------------|---|------------------------|
| - LIVE | | Time taken: 369200 ns | - | Comparison count: 1240 |
| - AGAIN | | Time taken: 30500 ns | - | Comparison count: 167 |
| - THINK | | Time taken: 90000 ns | - | Comparison count: 348 |
| - GIVING | | Time taken: 107900 ns | - | Comparison count: 601 |
| - HOW | | Time taken: 390500 ns | - | Comparison count: 1451 |
| - GIVE | | Time taken: 298200 ns | - | Comparison count: 1318 |
| - JUST | | Time taken: 94900 ns | - | Comparison count: 495 |
| - SOME | | Time taken: 150000 ns | - | Comparison count: 584 |
| - PUT | | Time taken: 3800 ns | - | Comparison count: 18 |
| - ROUND | | Time taken: 156600 ns | - | Comparison count: 700 |
| - EVERY | | Time taken: 28700 ns | - | Comparison count: 129 |
| - STOP | | Time taken: 81200 ns | - | Comparison count: 466 |
| - AS | | Time taken: 188700 ns | - | Comparison count: 1384 |
| - FROM | | Time taken: 46300 ns | - | Comparison count: 242 |

Total Solving Time = 2036500 ns
 Total Comparison = 9143
 Average Solving Time = 145464 ns
 Average Comparison Count = 653

3. Test case puzzle ukuran kecil ke-3 (16x18)

```

Enter file path: ../test/test3-small.txt

===== WORD SEARCH PUZZLE (16x18) =====
| L K E D X K K C O R E L T T I L Z M |
| V F F F H E F Z G M N E X X R S O L |
| W W H J R O P T A P O M S A L E M I |
| E Y F C E C N D Z X D W M H S K C V |
| R L K Y Y F I I R A N A A C E X R N |
| R S L U O S F I R M T R M G U N S B |
| X I N I O I A E T E R S U T J K J N |
| I L O N V T S H R I V O U W O Q T O |
| N O S X K H R L S S R N P G N W S G |
| E P K Y D Z S B H N O H E Y U L J Y |
| O A C A G W U A O P U N S D W A I Y |
| H N A L S R S T N U I C C W T T B A |
| P N J B G K A R S A D G S I H O G I |
| L A P A N B W L A N S I N G T I B S |
| J J Y N O Y C A R S O N C I T Y I N |
| O L B Y B V Q G C C O L U M B I A M |
=====

WORDS LIST (16 words):
- LITTLE ROCK
- DENVER
- SALEM
- JEFFERSON CITY
- LANSING
- AUGUSTA
- NASHVILLE
- ALBANY
- CARSON CITY
- JACKSON
- HARRISBURG
- PHOENIX
- BATON ROUGE
- ANNAPOLIS
- MADISON
- COLUMBIA

===== SOLUTION =====
| L K E D X K K C O R E L T T I L Z M |
| V F F F H E F Z G M N E X X R S O L |
| W W H J R O P T A P O M S A L E M I |
| E Y F C E C N D Z X D W M H S K C V |
| R L K Y Y F I I R A N A A C E X R N |
| R S L U O S F I R M T R M G U N S B |
| X I N I O I A E T E R S U T J K J N |
| I L O N V T S H R I V O U W O Q T O |
| N O S X K H R L S S R N P G N W S G |
| E P K Y D Z S B H N O H E Y U L J Y |
| O A C A G W U A O P U N S D W A I Y |
| H N A L S R S T N U I C C W T T B A |
| P N J B G K A R S A D G S I H O G I |
| L A P A N B W L A N S I N G T I B S |
| J J Y N O Y C A R S O N C I T Y I N |
| O L B Y B V Q G C C O L U M B I A M |
=====

===== STATISTICS =====
- LITTLE ROCK || Time taken: 119300 ns - Comparison count: 64
- DENVER || Time taken: 192700 ns - Comparison count: 1042
- SALEM || Time taken: 53000 ns - Comparison count: 211
- JEFFERSON CITY || Time taken: 34900 ns - Comparison count: 111
- LANSING || Time taken: 308300 ns - Comparison count: 1108
- AUGUSTA || Time taken: 223300 ns - Comparison count: 938
- NASHVILLE || Time taken: 218600 ns - Comparison count: 725
- ALBANY || Time taken: 207300 ns - Comparison count: 978
- CARSON CITY || Time taken: 181100 ns - Comparison count: 739
- JACKSON || Time taken: 167400 ns - Comparison count: 995
- HARRISBURG || Time taken: 46600 ns - Comparison count: 222
- PHOENIX || Time taken: 159000 ns - Comparison count: 997
- BATON ROUGE || Time taken: 177300 ns - Comparison count: 673
- ANNAPOLIS || Time taken: 204800 ns - Comparison count: 820
- MADISON || Time taken: 21500 ns - Comparison count: 118
- COLUMBIA || Time taken: 216100 ns - Comparison count: 1088
=====

Total Solving Time = 2531200 ns
Total Comparison = 10829
Average Solving Time = 158200 ns
Average Comparison Count = 676

```


4. Test case puzzle ukuran sedang ke-1 (20x22)

```
Enter file path: ../test/test1-medium.txt

===== WORD SEARCH PUZZLE (20x22) =====
| C Q D A M P L I F I E D Y X Y D K O X O X T |
| G I Z R R D N O I S I E R P M F Q V E E S I |
| G D J B S N U Y B T Y H U Y Y D M G N A Q B |
| E N L F K S N V D D F M Q F J M B H E Z W D |
| F Z W C A L Y F U S N W Z I I A S B R O S Y |
| D A I N X J C T O L G S C T F T C L H D D T |
| B S G R E A S S S C G P A N C W U J M R A G |
| N N N Q A R U A C U U F I E H O L N Z A O T |
| E N X N E I L Z R Y A S Y D P Z P W Y Z L T |
| I R K D L H G G U T C Q S I K P T V W I Y N |
| E W N V C U Q A P E O E Q E D S E V X W A E |
| S U H N D X F M L P B P W O D W D J Y S P L |
| D K G Y I N W Z E P Z K I O Y E G Y O D R L |
| K X S F I V A Q B E A L G U R J U U J M B E |
| U O W G C E A S T D H X F K M R G U S T O C |
| Y S G F A X E B U I R R E P O L R E T N I X |
| J T D G U X T F R O G J T O P Z P J P C E E |
| K L L A G S E D T F J P H R F Q Z Z N P Z U |
| F E M C H K L K V M Q T P U Q A H M M G R X |
| T Z Z T T I J M S O B Z Q E Y R S I R V S I |
=====
```

WORDS LIST (20 words):

- AMPLIFIED
- BEAST
- CAUGHT
- EAGLET
- EURO
- EXCELLENT
- FOCUSED
- GUSTO
- IDENTIFY
- INTERLOPER
- JEWS
- NOISIER
- OPIUM
- PAYLOADS
- PLAGIARIZE
- SCRUPLE
- SCULPTED
- THOUSAND
- UNDERSTUDY
- WIZARD

```
===== SOLUTION =====
| C Q D A M P L I F I E D Y X Y D K O X O X T |
| G I Z R R D N O I S I E R P M F Q V E E S I |
| G D J B S N U Y B T Y H U Y Y D M G N A Q B |
| E N L F K S N V D D F M Q F J M B H E Z W D |
| F Z W C A L Y F U S N W Z I I A S B R O S Y |
| D A I N X J C T O L G S C T F T C L H D D T |
| B S G R E A S S S C G P A N C W U J M R A G |
| N N N Q A R U A C U U F I E H O L N Z A O T |
| E N X N E I L Z R Y A S Y D P Z P W Y Z L T |
| I R K D L H G G U T C Q S I K P T V W I Y N |
| E W N V C U Q A P E O E Q E D S E V X W A E |
| S U H N D X F M L P B P W O D W D J Y S P L |
| D K G Y I N W Z E P Z K I O Y E G Y O D R L |
| K X S F I V A Q B E A L G U R J U U J M B E |
| U O W G C E A S T D H X F K M R G U S T O C |
| Y S G F A X E B U I R R E P O L R E T N I X |
| J T D G U X T F R O G J T O P Z P J P C E E |
| K L L A G S E D T F J P H R F Q Z Z N P Z U |
| F E M C H K L K V M Q T P U Q A H M M G R X |
| T Z Z T T I J M S O B Z Q E Y R S I R V S I |
=====
```

```

===== STATISTICS =====
- AMPLIFIED || Time taken: 124600 ns - Comparison count: 21
- BEAST || Time taken: 143900 ns - Comparison count: 536
- CAUGHT || Time taken: 400500 ns - Comparison count: 1867
- EAGLET || Time taken: 438900 ns - Comparison count: 1930
- EURO || Time taken: 539600 ns - Comparison count: 2955
- EXCELLENT || Time taken: 407900 ns - Comparison count: 1646
- FOCUSED || Time taken: 91000 ns - Comparison count: 389
- GUSTO || Time taken: 394500 ns - Comparison count: 2107
- IDENTIFY || Time taken: 325800 ns - Comparison count: 997
- INTERLOPER || Time taken: 319300 ns - Comparison count: 1361
- JEWS || Time taken: 367500 ns - Comparison count: 2031
- NOISIER || Time taken: 21700 ns - Comparison count: 114
- OPIUM || Time taken: 247000 ns - Comparison count: 1412
- PAYLOADS || Time taken: 241300 ns - Comparison count: 1302
- PLAGIARIZE || Time taken: 233800 ns - Comparison count: 1071
- SCRUPLE || Time taken: 150200 ns - Comparison count: 626
- SCULPTED || Time taken: 113000 ns - Comparison count: 420
- THOUSAND || Time taken: 317600 ns - Comparison count: 2044
- UNDERSTUDY || Time taken: 299100 ns - Comparison count: 966
- WIZARD || Time taken: 183300 ns - Comparison count: 1355
=====
Total Solving Time = 5360500 ns
Total Comparison = 25150
Average Solving Time = 268025 ns
Average Comparison Count = 1257

```

5. Test case puzzle ukuran sedang ke-2 (20x22)

```

Enter file path: ../test/test2-medium.txt

===== WORD SEARCH PUZZLE (20x22) =====
| K H D R O O T L E S S Z A J H U L H S E P Z |
| Q S P M I Q E H U K U U R A T R Y X C Y Z G |
| X Q W B N Q H L G O Q D B R E G N A H E E C |
| W Z Y C Z U H T D A B A J R E V V O K W C C |
| Y L T N A R B I V H E R Y F A S H Q M I E G |
| N X D X F E Z H Q K W L O B F V J U B T F C |
| I N T R U D E H V R E N M W P Y E V J N F M |
| M R I N X Z H C N U R C U X S R Q L S E A V |
| F K I N D N E S S V R O T K L I X V Y S T U |
| Y J H O R J Y G G W K O C Y Q U N J W S I Z |
| W M G T E Z S I V J D B N F V P I G D U J I |
| R C B B S U O A D F W A A W G S Y Q D R E E |
| O E Q V I O J M C O N T S Q K S B B I V C F |
| U R Y X D K F V R T Q E V I S N E P X E N F |
| G H J S I T V R S B U S O B E S U R B V E W |
| H B H R N R A P H C N A X J N L C B R C I S |
| T Y H R G P C X I E S E L W U A B V R D T K |
| Y P F A S L O V E W N O O L V C E B B S A G |
| Q Y L N Q C I Z B R V J M X Y W A L U M P N |
| Q N T B F R E I K U O K W O B P J E O B W J |
=====

```

WORDS LIST (20 words):

- ACTUALLY
- BRAVELY
- BROWSING
- BUBBLE
- CRUNCH
- EXPENSIVE
- EYEWITNESS
- HANGER
- INTRUDE
- KINDNESS
- OUGHT
- OWNS
- PATIENCE
- RESIDING
- ROOTLESS
- SANCTUM
- SNEEZE
- SPARROW
- TABOO
- VIBRANTLY

SOLUTION

```

=====
| K H D R O O T L E S S Z A J H U L H S E P Z |
| Q S P M I Q E H U K U U R A T R Y X C Y Z G |
| X Q W B N Q H L G O Q D B R E G N A H E C |
| W Z Y C Z U H T D A B A J R E V V O K W C C |
| Y L T N A R B I V H E R Y F A S H Q M I E G |
| N X D X F E Z H Q K W L O B F V J U B T F C |
| I N T R U D E H V R E N M W P Y E V J N F M |
| M R I N X Z H C N U R C U X S R Q L S E A V |
| F K I N D N E S S V R O T K L I X V Y S T U |
| Y J H O R J Y G G W K O C Y Q U N J W S I Z |
| W M G T E Z S I V J D B N F V P I G D U J I |
| R C B B S U O A D F W A A W G S Y Q D R E E |
| O E Q V I O J M C O N T S Q K S B B I V C F |
| U R Y X D K F V R T Q E V I S N E P X E N F |
| G H J S I T V R S B U S O B E S U R B V E W |
| H B H R N R A P H C N A X J N L C B R C I S |
| T Y H R G P C X I E S E L W U A B V R D T K |
| Y P F A S L O V E W N O O L V C E B B S A G |
| Q Y L N Q C I Z B R V J M X Y W A L U M P N |
| Q N T B F R E I K U O K W O B P J E O B W J |
=====

```

STATISTICS

| | | | | |
|--------------|--|-----------------------|---|------------------------|
| - ACTUALLY | | Time taken: 496200 ns | - | Comparison count: 1198 |
| - BRAVELY | | Time taken: 114500 ns | - | Comparison count: 238 |
| - BROWSING | | Time taken: 121400 ns | - | Comparison count: 307 |
| - BUBBLE | | Time taken: 708800 ns | - | Comparison count: 2478 |
| - CRUNCH | | Time taken: 234500 ns | - | Comparison count: 856 |
| - EXPENSIVE | | Time taken: 341000 ns | - | Comparison count: 1374 |
| - EYEWITNESS | | Time taken: 21200 ns | - | Comparison count: 79 |
| - HANGER | | Time taken: 74200 ns | - | Comparison count: 289 |
| - INTRUDE | | Time taken: 144700 ns | - | Comparison count: 538 |
| - KINDNESS | | Time taken: 202500 ns | - | Comparison count: 740 |
| - OUGHT | | Time taken: 374500 ns | - | Comparison count: 1657 |
| - OWNS | | Time taken: 521400 ns | - | Comparison count: 2677 |
| - PATIENCE | | Time taken: 549600 ns | - | Comparison count: 1905 |
| - RESIDING | | Time taken: 250000 ns | - | Comparison count: 943 |
| - ROOTLESS | | Time taken: 4600 ns | - | Comparison count: 20 |
| - SANCTUM | | Time taken: 582800 ns | - | Comparison count: 1529 |
| - SNEEZE | | Time taken: 469200 ns | - | Comparison count: 1978 |
| - SPARROW | | Time taken: 696400 ns | - | Comparison count: 2052 |
| - TABOO | | Time taken: 326600 ns | - | Comparison count: 1728 |
| - VIBRANTLY | | Time taken: 100100 ns | - | Comparison count: 356 |

```

=====
Total Solving Time      = 6334200 ns
Total Comparison        = 22942
Average Solving Time    = 316710 ns
Average Comparison Count = 1147
=====

```

6. Test case puzzle ukuran sedang ke-3 (20x22)

```
Enter file path: ../test/test3-medium.txt

===== WORD SEARCH PUZZLE (20x22) =====
| S P P J Q F I Q R M H J W S W B R E B Z S U |
| S O T Z Z T J M A P R W H X E S K L P S C R |
| P W D G O S B R G B T U E R C J P G C M E S |
| A A N X X B C O M W T B U P G B E N X E R D |
| G V P C L S M W T T O R E R W L A A Y W T Y |
| V W H E B F S B L F N J G X O A H T W B I W |
| K B A A R S I E B A L J B H R P W C S R F P |
| K B R J G B R P N Y I U P I R Y S E X E Y B |
| Z Q V X X X A U F O M I Z P I T U R Q L I H |
| H I T I H A T C B T B C M G S E R K Z A N Y |
| J E E V L K E V K E T Y N T O H G R O X G A |
| X E D I S G N I R E C M K J M C G Y R D F R |
| B M X N H E O Q E O D B P L E R W R H U E W |
| Z Z R B W L M F M B D V L I B A Y X Q Q L C |
| M P F X X E O M Y G I L Q Y H V M S Z J O S |
| V K L B L N E N A Y L S S E R P X E N Q D K |
| T X F L S N Z L G G Y W U B K D O U B T O G |
| P R E Z T Y A U F I H U O B M M R T Q Q D K |
| L Z M S B U S K H P N I S R P I Z F D V W Z |
| T A C L U M I N A R Y G N C B V H N N A L E |
=====
```

WORDS LIST (20 words):

- ARCHETYPAL
- BUSK
- CERTIFYING
- COMMENTS
- DOUBT
- EXPRESSLY
- FEET
- GALA
- JUICY
- LONGING
- LUMINARY
- MONETARISM
- PAPERBACK
- RECTANGLE
- RELAX
- RINGSIDE
- SCRAM
- SHUTTLE
- TAHITI
- WORRISOME

```
===== SOLUTION =====
| S P P J Q F I Q R M H J W S W B R E B Z S U |
| S O T Z Z T J M A P R W H X E S K L P S C R |
| P W D G O S B R G B T U E R C J P G C M E S |
| A A N X X B C O M W T B U P G B E N X E R D |
| G V P C L S M W T T O R E R W L A A Y W T Y |
| V W H E B F S B L F N J G X O A H T W B I W |
| K B A A R S I E B A L J B H R P W C S R F P |
| K B R J G B R P N Y I U P I R Y S E X E Y B |
| Z Q V X X X A U F O M I Z P I T U R Q L I H |
| H I T I H A T C B T B C M G S E R K Z A N Y |
| J E E V L K E V K E T Y N T O H G R O X G A |
| X E D I S G N I R E C M K J M C G Y R D F R |
| B M X N H E O Q E O D B P L E R W R H U E W |
| Z Z R B W L M F M B D V L I B A Y X Q Q L C |
| M P F X X E O M Y G I L Q Y H V M S Z J O S |
| V K L B L N E N A Y L S S E R P X E N Q D K |
| T X F L S N Z L G G Y W U B K D O U B T O G |
| P R E Z T Y A U F I H U O B M M R T Q Q D K |
| L Z M S B U S K H P N I S R P I Z F D V W Z |
| T A C L U M I N A R Y G N C B V H N N A L E |
=====
```

```

===== STATISTICS =====
- ARCHETYPAL || Time taken: 762700 ns - Comparison count: 1173
- BUSK || Time taken: 969700 ns - Comparison count: 2817
- CERTIFYING || Time taken: 505699 ns - Comparison count: 1455
- COMMENTS || Time taken: 369901 ns - Comparison count: 1226
- DOUBT || Time taken: 591200 ns - Comparison count: 2315
- EXPRESSLY || Time taken: 424399 ns - Comparison count: 1553
- FEET || Time taken: 364400 ns - Comparison count: 1981
- GALA || Time taken: 268200 ns - Comparison count: 2195
- JUICY || Time taken: 100501 ns - Comparison count: 788
- LONGING || Time taken: 201101 ns - Comparison count: 1578
- LUMINARY || Time taken: 339600 ns - Comparison count: 1965
- MONETARISM || Time taken: 198600 ns - Comparison count: 1132
- PAPERBACK || Time taken: 28600 ns - Comparison count: 186
- RECTANGLE || Time taken: 140600 ns - Comparison count: 800
- RELAX || Time taken: 112100 ns - Comparison count: 893
- RINGSIDE || Time taken: 170300 ns - Comparison count: 1255
- SCRAM || Time taken: 68499 ns - Comparison count: 458
- SHUTTLE || Time taken: 16201 ns - Comparison count: 72
- TAHITI || Time taken: 158301 ns - Comparison count: 1132
- WORRISOME || Time taken: 100601 ns - Comparison count: 401
=====
Total Solving Time = 5891203 ns
Total Comparison = 25375
Average Solving Time = 294560 ns
Average Comparison Count = 1268

```

7. Test case puzzle ukuran besar ke-1 (32x34)

```

Enter file path: ../test/test1-large.txt

===== WORD SEARCH PUZZLE (32x34) =====
| R P H O D P W I P L A G R E E A E J R E A W K W W P A I K O N C N C |
| E S Z H K T T R J G M J W H G N Y J T V I N B R E D H K C F A G Q C |
| U Q K Q W C O L N A D G R A N D F A T H E R S I B W R M Q O F N Q O |
| R R L K I F K I B B B C U D I K E V S G G P G F W Z L P S H N C B T |
| G K A L O U T W E Y Z Y U R T W U F K J F A N A D U L A T I O N M Z |
| V T E Y Q A C A A G C V Q S E Z G S I I J R I I M W I H O G U A V H |
| Y R K O O C S W A Z N G U R A L O P H Q Q T L O L B P G Y R G Q A N |
| J C I L M T D E D H S I H A J F L B E S T N L R F Y O Z O F G N L Y |
| Z M G B U A C Z D X H T G O T L O H M G W E I R O N J N P R W E I V |
| Y W H W O C R Y S I R K S G F G E Y A N Y R M G G S S T J D W G A V |
| B G T R O D L N T E T K J Q A Z D J I L R I B J Z G V W I G L L N H |
| F X J C Q G O A V M W O C O R R I A S Y B N A Y G Q B D A H Q E T R |
| K L A Y J D S O S J Z D R A N N D U I H C G L E E Q R B L Y Y C L M |
| T H O Z S X E F S S E O C Q M G O D L M U W O R J E Q H L R E T Y G |
| E A J W B C K T A T I S K C A I X N P Q S F H G V I L Q A U O F J S |
| K E Z L N A U M A E O C Q S G V G V W E D U M O N E G T R N A U P S |
| A T K S W D B N Y Z G W A I U Z P R F G R I X P E I E O D U S L Z H |
| T N M I I G I B K K R A L L H V R J E U O F S Q X M P W C Z P D N T |
| P I X O L C B H L I Y E D L L T P E O E V Y E A O G O P K K O Y N N |
| U A S E S Y Z S V E R R D A Q Y E D X P V Z U C P X B L I I V I I K |
| T K Z A D N D E Q K R B A E C S E M E X S W X L T P V D Z Z L T H I |
| J N F G E K R A D R L Z S H G N I M R I F F A D S I R F E C X R T C |
| Z A C N P M U D L A E I S B O L M V E L I B O S N F N O I I I E Q H |
| X D V I O X Q X S X P J I W B E K D M V R I H G B S G V E M D X B |
| G F C Z R Q B T P V G F T Q O N Q G P W R P D A R P B I E A E J Y P |
| D Z H I T Z I N N X Q I U L I M O D R A T S Y E F D W E Y K L V T Y |
| U B M M M N M F E U T S I M D C J T R V I S K B B C K Z A M J M K W |
| N A G O G D E M E E D O A T U O E D I H B O R V F G E O Q H G V O H |
| J T C N G U C V P X B T V U D M J D V J O E B H S C C U M K R L L E |
| P X Y O O U Q L Q R S W Z K Y Z E Z Q L P N E N A E H U T X A K G S |
| H T M C S W H O A I P M M X D X X A S U R K G X Q G K H O L W R E E |
| N R F E K C D P H E Q T E C A L O S S R V Y I X I I N P M D I T F E |
=====

WORDS LIST (48 words):
- ADULATION
- AFFIRMING
- BABBLE
- BEAST
- BEST
- BLASTING
- CLASSICALLY

```

- COMETARY
- DEEMED
- DEPORT
- DISAPPROVAL
- DRAGGING
- ECONOMIZING
- EDITOR
- EGGS
- FASCINATED
- FLOW
- GLOATING
- GONG
- GRANDFATHER
- HIDEOUT
- HISTAMINE
- ICON
- IDEOLOGUE
- INBRED
- LADYLIKE
- LINT
- LOOKER
- MILLING
- NEGLECTFUL
- OVERDID
- OVERTHREW
- PARBOIL
- PARTNERING
- PERFECTING
- PETITIONED
- RELICT
- RELIGIOUSLY
- RIVER
- ROADWAY
- SOLACE
- STARDOM
- STUDIO
- SUPERB
- TINGE
- UPTAKE
- VALIANTLY
- ZIPPING

```

===== SOLUTION =====
| RPHODPWIPLAGREFAEJREAWKWPAIKONCNC|
| ESZHKTTIRJGMJWHGNYJTVINBREDHKCFAGQC|
| UQKQWCOLNADGRANDFATHERSIBWRMQOFNQO|
| RRLKIFKIBBBCUDIKEVSGGPGFWZLPSHNCBT|
| GKALOUITWEYZYURTWUFKJFANADULATIIONMZ|
| VT EYQACAAAGCVQSEZGSIIJRIMWIHOGUAVH|
| YKOOCSWAZNGURALOPHQQTLOLBPGYRGQAN|
| JCI LMTDE DHSIH AJFLBESTNLRFY OZ OFGNLY|
| ZMGBUACZDXHTGOTLOHMGWEIRONJNPRWEIV|
| YWHWOCRYSIRKS GFG EYANYRMGGSTJ DWGAV|
| BGTRODLNTETKJQAZDJILRIBJZGVWIGL LNH|
| FXJCQGOAVMWOCORRIAS YBNAYGQB DAHQETR|
| KLAYJDSOSSJZDRANN DUIHCGL EEQRBL YCLM|
| THOZS XEF SSEOCQMGQDL MUWORJEQHLRETYG|
| EAJWBCKTATISKCAIXNPQS FHVILQAUFJS|
| KEZLN AUMAEOCQS GVG VWE DUMON EGTRNAUPS|
| ATKSWDBN YZGWA I UZPRFGRI XPEIEODUSLZH|
| TNM IIGIBKKRA LLHVRJEUOFSQXMPWCZPDNT|
| PIXOLCBHLIYE D L LTP EOEVYEAOGOPKKOYNN|
| UASESYZSVERRDAQYEDXPVZUCPXB LIIV I IK|
| TKZADNDEQKRBAECS EMEXSXLTPVDZ ZLTHI|
| JNFGEKRA DRLZSHGNIMRIFFADSIRFECXRTC|
| ZACNPMUDLAEISBOLMV ELIBOSNFNOIIEEQH|
| XDVIOXQXSXXPJ IWB E KDMVRIHGBSGVEMDXB|
| GFCZRQBTPVGF TQONQGPWRPDA RPBIEAEJYP|
| DZHITZINNXQIULIMODRATSY EFDWEYKLV TY|
| UBM MMNMF EU TSIMDCJTRVISKBBCKZAMJMKW|
| NAGO GDEMEEDOATUOEDIHBORVVFGEQHGVOH|
| JTCNGUCVPXBTVUDMJDVJOEBHSCCUMKR LLE|
| PXYOOUQLQRSWZKYZEZQLPNENAEHUTXAKGS|
| HTMCSWHOAI PPMXDXXASURKGXQGGKHOLWREE|
| N RFEKCDPHEQT ECALOSSRVYIXIINPMDITFE|
=====

```

| ===== STATISTICS ===== | | | |
|--------------------------|--|------------------------|--------------------------|
| - ADULATION | | Time taken: 319000 ns | - Comparison count: 710 |
| - AFFIRMING | | Time taken: 1140900 ns | - Comparison count: 4444 |
| - BABBLE | | Time taken: 713400 ns | - Comparison count: 3049 |
| - BEAST | | Time taken: 125600 ns | - Comparison count: 530 |
| - BEST | | Time taken: 313300 ns | - Comparison count: 1673 |
| - BLASTING | | Time taken: 801100 ns | - Comparison count: 4188 |
| - CLASSICALLY | | Time taken: 280300 ns | - Comparison count: 1241 |
| - COMETARY | | Time taken: 724100 ns | - Comparison count: 4063 |
| - DEEMED | | Time taken: 1068200 ns | - Comparison count: 6421 |
| - DEPORT | | Time taken: 743700 ns | - Comparison count: 4589 |
| - DISAPPROVAL | | Time taken: 469000 ns | - Comparison count: 2601 |
| - DRAGGING | | Time taken: 414700 ns | - Comparison count: 2365 |
| - ECONOMIZING | | Time taken: 1182100 ns | - Comparison count: 5374 |
| - EDITOR | | Time taken: 165700 ns | - Comparison count: 1349 |
| - EGGS | | Time taken: 331400 ns | - Comparison count: 3075 |
| - FASCINATED | | Time taken: 419700 ns | - Comparison count: 3946 |
| - FLOW | | Time taken: 242000 ns | - Comparison count: 2601 |
| - GLOATING | | Time taken: 138300 ns | - Comparison count: 1322 |
| - GONG | | Time taken: 128900 ns | - Comparison count: 1546 |
| - GRANDFATHER | | Time taken: 37900 ns | - Comparison count: 326 |
| - HIDEOUT | | Time taken: 540600 ns | - Comparison count: 6021 |
| - HISTAMINE | | Time taken: 562900 ns | - Comparison count: 5887 |
| - ICON | | Time taken: 15600 ns | - Comparison count: 145 |
| - IDEOLOGUE | | Time taken: 205300 ns | - Comparison count: 2002 |
| - INBRED | | Time taken: 27100 ns | - Comparison count: 259 |
| - LADYLIKE | | Time taken: 411200 ns | - Comparison count: 4752 |
| - LINT | | Time taken: 382000 ns | - Comparison count: 5319 |
| - LOOKER | | Time taken: 499900 ns | - Comparison count: 6784 |
| - MILLING | | Time taken: 162400 ns | - Comparison count: 1792 |
| - NEGLECTFUL | | Time taken: 138700 ns | - Comparison count: 1128 |
| - OVERDID | | Time taken: 275400 ns | - Comparison count: 3364 |
| - OVERTHREW | | Time taken: 196300 ns | - Comparison count: 2154 |
| - PARBOIL | | Time taken: 544200 ns | - Comparison count: 6574 |
| - PARTNERING | | Time taken: 60600 ns | - Comparison count: 512 |
| - PERFECTING | | Time taken: 241300 ns | - Comparison count: 2465 |
| - PETITIONED | | Time taken: 484900 ns | - Comparison count: 5117 |
| - RELICT | | Time taken: 98400 ns | - Comparison count: 1076 |
| - RELIGIOUSLY | | Time taken: 328200 ns | - Comparison count: 3484 |
| - RIVER | | Time taken: 331000 ns | - Comparison count: 4155 |
| - ROADWAY | | Time taken: 156400 ns | - Comparison count: 1977 |
| - SOLACE | | Time taken: 471000 ns | - Comparison count: 7090 |
| - STARDOM | | Time taken: 520800 ns | - Comparison count: 5797 |
| - STUDIO | | Time taken: 241100 ns | - Comparison count: 2881 |
| - SUPERB | | Time taken: 584900 ns | - Comparison count: 7099 |
| - TINGE | | Time taken: 75700 ns | - Comparison count: 760 |
| - UPTAKE | | Time taken: 358300 ns | - Comparison count: 4258 |
| - VALIANTLY | | Time taken: 105500 ns | - Comparison count: 850 |
| - ZIPPING | | Time taken: 390100 ns | - Comparison count: 4546 |
| ===== | | | |
| Total Solving Time | | = 18169100 ns | |
| Total Comparison | | = 153661 | |
| Average Solving Time | | = 378522 ns | |
| Average Comparison Count | | = 3201 | |

8. Test case puzzle ukuran besar ke-2 (32x34)

```
Enter file path: ../test/test2-large.txt

===== WORD SEARCH PUZZLE (32x34) =====
| U H L V Z T Y E L L O W P L C L E X O S S K O U J E F O M Q H I B F |
| F H S I X T N G T N O T B F G T S S M J Y I H W T S Y U U R H E T N |
| N U A S R J R E M C T E K W I A R U U L G H V T Q X X Q Z U A Y C R |
| Q O K N E T G L I D R M B T W R G B Q U F I C T X A S W V T B V E O |
| C D X M I L S F C L W I C V A Z M Y V W O V L T S W T Y I V E B B W |
| F L J T Z I T C I K L A M R S C Y E Z H Y T W A O A A F R C I S C K |
| E R S R V X G I B W L U U S O P H X Z Q X M A H Z L Y J Z F C N E G |
| P I W D P M F J M A H D B N N Y Z Y J I B N X D S K B L F U B V Z K |
| S M N U V G H M T I E P T E K Y J S T Z D V L S A A Q A R E J O Y W |
| G P F R Q W K S G V L R C Y E G S V I Q F R L X G D D E B T C B S R |
| U U E I A L K U D H O F M X Z E M J C R E O A Y H L S K R I H W E K |
| K K N M X T M L P V J G K Z N I J F I H Y A O D X T H G O N N J I N |
| L X D I A P W G H M T I E P T E K Y J S T Z D V L S A A Q A R E J O Y W |
| M J G D V N U R A D E K T S M K Z S R N M V A A G A L V C J C H X W |
| Z X K Y O W T C V E H U I S B W K H I F R V B J I P T T H A K A I N |
| W S E H D Z M B K J M A E C E Y W T Y L I H R Q V D O S E N V I S O |
| Q S C W E H M T I E P T E K Y J S T Z D V L S A A Q A R E J O Y W |
| Z B N S P J X B F Y A E J Z W K L O O K C U H C K P A U S U A E D I |
| S W E J E H Z N H I U R J L C E W E X C O I N C I D E T G I F D R M |
| W Z L C E M I P R A J H R I Z E N C R O A C H I N G I G D S C C E A |
| X O U L T N O O U N K O P G Y Y L T P M O R P T N L S L B H B V V V |
| I B B Q A O T C J N W H T T R I R O G S R O T A T I M I L E O Q H K |
| J F R B R A N J W E P Y X L A M M D F L E E C E N V T T I D F Y P T |
| W J U U M T A F Q T I U Z B T N V T J K S C B D C N O T N C G T Z U |
| D S T E R S H Y P N S P B D N J H N P P O Y X J R L V M K S L Q I P |
| U U R V C Q B S H A T E S L E L A N O R G J W A I E P B E E U J P S |
| L C E D F R O W I A O X K H M D A Z O U I W R X B K G L D E B R K T |
| X E E K L X K E I F L P Q I M Q U M M L R O I W E C N N J K N Q A A |
| N A M O W S N I K N G Q R K O R N K E F C Y N T Z R I N A U U W I G |
| S E G A J D Z V C S E O H O C F N B W Q P F V L T V O H V R C N G E |
| D K Q E N P T Q Y A H R D C L F Z A A F H S N R E T R A I N R I M C |
| N T B H L E O O O S O H Y D E Z I N O I L G Q U D I M N I L L A H W |
=====
```

WORDS LIST (48 words):

- AMIGO
- ANGUISHED
- ANTENNA
- ARRANGER
- BEATIFY
- BLINKED
- BLOTCHY
- BROACHED
- CHUCK
- COINCIDE
- COMES
- COMMENTARY
- CONTROVERT
- CREMATORIA
- DOGFISH
- EBULLIENT
- ELDEST
- ELSE
- ENAMEL
- ENCROACHING
- EQUESTRIAN
- FIBER
- FLEECE
- HAILED
- HUSKY
- IMITATOR
- KINSWOMAN
- KNOWN
- LIMITLESS
- LIONIZED
- MOROCCAN
- MUTENESS
- OBSCUREST
- PICKETING

- PISTOL
- POOF
- PROMPTLY
- RATE
- RETRAIN
- SHOP
- SIXTIES
- SMUG
- STALACTITE
- STANDARDIZE
- TURBULENCE
- UPSTAGE
- WINERY
- YELLOW

```

===== SOLUTION =====
| U H L V Z T Y E L L O W P L C L E X O S S K O U J E F O M Q H I B F |
| F H S I X T N G T N O T B F G T S S M J Y I H W T S Y U U R H E T N |
| N U A S R J R E M C T E K W I A R U U L G H V T Q X X Q Z U A Y C R |
| Q O K N E T G L I D R M B T W R G B Q U F I C T X A S W V T B V E O |
| C D X M I L S F C L W I C V A Z M Y V W O V L T S W T Y I V E B B W |
| F L J T Z I T C I K L A M R S C Y E Z H Y T W A O A A F R C I S C K |
| U E I A L K U D H O F M X Z E M J C R E O A Y H L S K R I H W E K |
| P I W D P M F J M A H D B N N Y Z Y J I B N X D S K B L F U B V Z K |
| S M N U V G H M T I E P T E K Y J S T Z D V L S A A Q A R E J O Y W |
| G P F R Q W K S G V L R C Y E G S V I Q F R L X G D D E B T C B S R |
| U U E I A L K U D H O F M X Z E M J C R E O A Y H L S K R I H W E K |
| K K N M X T M L P V J G K Z N I J F I H Y A O D X T H G O N N J I N |
| L X D I A P W N E U L X Z E N F L A U C G T T P N X E Q A G S G T O |
| M J G D V N U R A D E K T S M K Z S R N M V A A G A L V C J C H X W |
| Z X K Y O W T C V E H U I S B W K H I F R V B J I P T T H A K A I N |
| W S E H D Z M B K J M A E C E Y W T Y L I H R Q V D O S E N V I S O |
| Q S C W E Q U E S T R I A N J D E X D T A E D H J W Y H D G Z L K G |
| Z B N S P J X B F Y A E J Z W K L O O K C U H C K P A U S U A E D I |
| S W E J E H Z N H I U R J L C E W E X C O I N C I D E T G I F D R M |
| W Z L C E M I P R A J H R I Z E N C R O A C H I N G I G D S C C E A |
| X O U L T N O O U N K O P G Y Y L T P M O R P T N L S L B H B V V V |
| I B B Q A O T C J N W H T T R I R O G S R O T A T I M I L E O Q H K |
| J F R B R A N J W E P Y X L A M M D F L E E C E N V T T I D F Y P T |
| W J U U M T A F Q T I U Z B T N V T J K S C B D C N O T N C G T Z U |
| D S T E R S H Y P N S P B D N J H N P P O Y X J R L V M K S L Q I P |
| U U R V C Q B S H A T E S L E L A N O R G J W A I E P B E U J P S |
| L C E D F R O W I A O X K H M D A Z O U I W R X B K G L D E B R K T |
| X E E K L X K E I F L P Q I M Q U M M L R O I W E C N N J K N Q A A |
| N A M O W S N I K N G Q R K O R N K E F C Y N T Z R I N A U U W I G |
| S E G A J D Z V C S E O H O C F N B W Q P F V L T V O H V R C N G E |
| D K Q E N P T Q Y A H R D C L F Z A A F H S N R E T R A I N R I M C |
| N T B H L E O O O S O H Y D E Z I N O I L G Q U D I M N I L L A H W |
=====

```

| STATISTICS | | | |
|--------------|--|-----------------------|--------------------------|
| - AMIGO | | Time taken: 972099 ns | - Comparison count: 4844 |
| - ANGUISHED | | Time taken: 432500 ns | - Comparison count: 2804 |
| - ANTENNA | | Time taken: 751000 ns | - Comparison count: 5706 |
| - ARRANGER | | Time taken: 883999 ns | - Comparison count: 6461 |
| - BEATIFY | | Time taken: 19200 ns | - Comparison count: 152 |
| - BLINKED | | Time taken: 450799 ns | - Comparison count: 4507 |
| - BLOTCHY | | Time taken: 167501 ns | - Comparison count: 1374 |
| - BROACHED | | Time taken: 176100 ns | - Comparison count: 1724 |
| - CHUCK | | Time taken: 351700 ns | - Comparison count: 4175 |
| - COINCIDE | | Time taken: 328701 ns | - Comparison count: 3803 |
| - COMES | | Time taken: 374100 ns | - Comparison count: 5111 |
| - COMMENTARY | | Time taken: 423900 ns | - Comparison count: 5355 |
| - CONTROVERT | | Time taken: 58901 ns | - Comparison count: 778 |
| - CREMATORIA | | Time taken: 308000 ns | - Comparison count: 4893 |
| - DOGFISH | | Time taken: 476400 ns | - Comparison count: 6452 |
| - EBULLIENT | | Time taken: 135300 ns | - Comparison count: 1261 |

```

- ELDEST           Time taken: 494701 ns - Comparison count: 4296
- ELSE             Time taken: 616701 ns - Comparison count: 6556
- ENAMEL           Time taken: 605999 ns - Comparison count: 7134
- ENCROACHING      Time taken: 264400 ns - Comparison count: 3546
- EQUESTRIAN       Time taken: 342700 ns - Comparison count: 2921
- FIBER            Time taken: 100301 ns - Comparison count: 1399
- FLEECE           Time taken: 307800 ns - Comparison count: 5131
- HAILED           Time taken: 188900 ns - Comparison count: 3024
- HUSKY            Time taken: 146500 ns - Comparison count: 2594
- IMITATOR         Time taken: 314900 ns - Comparison count: 4692
- KINSWOMAN        Time taken: 387101 ns - Comparison count: 5479
- KNOWN            Time taken: 137701 ns - Comparison count: 2428
- LIMITLESS        Time taken: 110900 ns - Comparison count: 1483
- LIONIZED         Time taken: 436800 ns - Comparison count: 6383
- MOROCCAN         Time taken: 387400 ns - Comparison count: 5697
- MUTENESS         Time taken: 230800 ns - Comparison count: 3038
- OBSCUREST        Time taken: 49500 ns - Comparison count: 589
- PICKETING        Time taken: 374301 ns - Comparison count: 3967
- PISTOL           Time taken: 416901 ns - Comparison count: 5055
- POOF             Time taken: 314300 ns - Comparison count: 3008
- PROMPTLY         Time taken: 358400 ns - Comparison count: 4258
- RATE             Time taken: 421100 ns - Comparison count: 5588
- RETRAIN          Time taken: 439699 ns - Comparison count: 6589
- SHOP             Time taken: 230001 ns - Comparison count: 4481
- SIXTIES          Time taken: 235701 ns - Comparison count: 3414
- SMUG             Time taken: 8400 ns - Comparison count: 102
- STALACTITE       Time taken: 125900 ns - Comparison count: 1341
- STANDARDIZE      Time taken: 193500 ns - Comparison count: 2688
- TURBULENCE       Time taken: 326600 ns - Comparison count: 4685
- UPSTAGE          Time taken: 334099 ns - Comparison count: 5317
- WINERY           Time taken: 351400 ns - Comparison count: 6086
- YELLOW           Time taken: 2500 ns - Comparison count: 29
=====
Total Solving Time      = 15566106 ns
Total Comparison        = 182398
Average Solving Time    = 324293 ns
Average Comparison Count = 3799

```

9. Test case puzzle ukuran besar ke-3 (32x34)

```

Enter file path: ../test/test3-large.txt

===== WORD SEARCH PUZZLE (32x34) =====
| A S Q K K Y R E N N A C Y E I K I S X D V T J K H O N L Y Z T Q Z Z |
| M B V G K B P A Z H K C E R U T A E F C M D B S C A P E G O A T B Q |
| Q C A V S D C F L C Q F H B O Y L Z W B R T C F M A W B I U U I N |
| W R F T R V J M K N M S P F U Y P G O Q H O P I A G D I U F D D F U |
| P E C A E P O Q R Q P U D K N J R Y T S I M T Y P K F T N Z U I W T |
| Q A T Y H M H G B K E P A C S Y T I C A K R P C F U G T H D M J A T |
| E K W D L Q E R E L J H T M U G U T F E E B L Y A N M Q F H U R S P |
| Y I B O J D I N H Z F S K D E V B B P V R P E Q X F T W Z I C C F S |
| M N R C M O A J T W A P K X F W I C G I S B Y S X R E V O U O O E B |
| A G S Y N C H R O N I Z E P Z H F D N G W E A H F L E N A O I C F D |
| T P I S U Y W X Y A B R E L G X O E I J V I N U R Y E E E C D F Z S |
| P F M D E M I O M G T E T I B B C R D X Y U J Y P D R F T B I Y V E |
| N U H Q A S S H F I J M A H H O A I E D T L I J P U G P Q M G K T D |
| F E E S T R U W N Y D U V I E V L A E L H S T A B H E E A R F A Y S |
| Q N E K I L A G R T O T T E I C E P C U C F N F T W D E E O B Q C E |
| W E W C T R J P O B O H D X K X K S X C G C P I S H Y L Y R Q Z F D |
| M P U N I S H I N G R K I C B N L E E E I V V D A O C R E J H O Q X |
| C E P A M D N I H T K C R H O W S D A V W B N P C G U C E P J G R T |
| U D A F Z S Q L B V N W I Z Y W O J T P I I Z C K X A Y Z F N H P U |
| R E I Q J B R P Y I O R S E R S X Z Z O W E A U S X Y L B I I W B S |
| C J J I X E K O O S B R Z R T V M K U O T N E M E M U Z K E H N A G |
| K H A E G W J W T K C Z P F R A Y E N N D H T A W T A A Q N O R E E |
| V R G V O Q C A Y C Y X O W E E T K H D Z A J H U L W M H I S P Z R |
| Q E N I S P M D I S O Q R H P U K I U U S I N C E R I T Y T R A T R |
| Y S I S X C Z O G P X R Q W R B N Q V H L G O Q D R E C W N Z Y C F |
| Z E L S U H T P D A A A P G O J E V V A O J C C G H E Y F E S H Q E |
| M M D E E G N T X R D X F R V E Z H Q K R W L L E B F J U P B F C H |
| H B N S V R E E N T P Y V A E J F M M R I G I N X R Z X R R Q S A C |
| V L O B F V R D K A L X V V T U Y J H O J P Y G G W E K Y U Q U J W |
| I E F O Z W M G T N Z S I Y V J D F V P I D U J I R C V B T B U O D |
| F W G S S A P R U S S Y Q G N I Z I T A M A R D D R E E E Q V O J M |
| N A M B I D E X T R O U S Q K S B B I V F R Y X K F V Q F R H J S T |
=====

```

WORDS LIST (48 words):

- ABATEMENT
- ADOPTED
- AGAINST
- AMBIDEXTROUS
- BENEFACTOR
- BIFOCAL
- BUREAUCRAT
- CANNERY
- CHEF
- CITYSCAPE
- CLERGY
- CREAKING
- DEGREE

- DESPAIRED
- DOORKNOB
- DRAMATIZING
- DRAT
- EXACERBATE
- EXCEEDING
- EXERTING
- FEATURE
- FEEBLY
- FONDLING
- GRAVITATE
- GRAVY
- HIND
- INDUCED
- JEEP
- LIKEN
- MEMENTO
- MISTY
- OBSESSIVE
- PARADIGM
- PILGRIM
- PROCTORS
- PUNISHING
- REFINER
- REPROVE
- RESEMBLE
- REVERE
- SCAPEGOAT
- SINCERITY
- SPARTAN
- SURPASS
- SYNCHRONIZE
- TURPENTINE
- WAKING
- WINDSWEPT

```
===== SOLUTION =====
| A S Q K K Y R E N N A C Y E I K I S X D V T J K H O N L Y Z T Q Z Z |
| M B V G K B P A Z H K C E R U T A E F C M D B S C A P E G O A T B Q |
| Q C A V S D C F L C Q F H B O Y L Z W B R T C F M A W B T U U T I N |
| W R F T R V J M K N M S P F U Y P G O Q H O P I A G D I U F D D F U |
| P E C A E P O Q R Q P U D K N J R Y T S I M T Y P K F T N Z U I W T |
| Q A T Y H M H G B K E P A C S Y T I C A K R P C F U G T H D M J A T |
| E K W D L Q E R E L J H T M U G U T F E E B L Y A N M Q F H U R S P |
| Y I B O J D I N H Z F S K D E V B B P V R P E Q X F T W Z I C C F S |
| M N R C M O A J T W A P K X F W I C G I S B Y S X R E V O U O O E B |
| A G S Y N C H R O N I Z E P Z H F D N G W E A H F L E N A O I C F D |
| T P I S U Y W X Y A B R E L G X O E I J V I N U R Y E E E C D F Z S |
| P F M D E M I O M G T E T I B B C R D X Y U J Y P D R F T B I Y V E |
| N U H Q A S S S H F I J M A H H O A I E D T L I J P U G P Q M G K T D |
| F E E S T R U W N Y D U V I E V L A E L H S T A B H E E A R F A Y S |
| Q N E K I L A G R T O T T E I C E P C U C F N F T W D E E O B Q C E |
| W E W C T R J P O B O H D X K X K S X C G C P I S H Y L Y R Q Z F D |
| M P U N I S H I N G R K I C B N L E E E I V V D A O C R E J H O Q X |
| C E P A M D N I H T K C R H O W S D A V W B N P C G U C E P J G R T |
| U D A F Z S Q L B V N W I Z Y W O J T P I I Z C K X A Y Z F N H P U |
| R E I Q J B R P Y I O R S E R S X Z Z O W E A U S X Y L B I I W B S |
| C J J I X E K O O S B R Z R T V M K U O T N E M E M U Z K E H N A G |
| K H A E G W J W T K C Z P F R A Y E N N D H T A W T A A Q N O R E E |
| V R G V O Q C A Y C Y X O W E E T K H D Z A J H U L W M H I S P Z R |
| Q E N I S P M D I S O Q R H P U K I U U S I N C E R I T Y T R A T R |
| Y S I S X C Z O G P X R Q W R B N Q V H L G O Q D R E C W N Z Y C F |
| Z E L S U H T P D A A A P G O J E V V A O J C C G H E Y F E S H Q E |
| M M D E E G N T X R D X F R V E Z H Q K R W L L E B F J U P B F C H |
| H B N S V R E E N T P Y V A E J F M M R I G I N X R Z X R R Q S A C |
| V L O B F V R D K A L X V V T U Y J H O J P Y G G W E K Y U Q U J W |
| I E F O Z W M G T N Z S I Y V J D F V P I D U J I R C V B T B U O D |
| F W G S S A P R U S S Y Q G N I Z I T A M A R D D R E E E Q V O J M |
| N A M B I D E X T R O U S Q K S B B I V F R Y X K F V Q F R H J S T |
=====
```

| ===== STATISTICS ===== | | | |
|--------------------------|--|------------------------|--------------------------|
| - ABATEMENT | | Time taken: 228600 ns | - Comparison count: 14 |
| - ADOPTED | | Time taken: 1286300 ns | - Comparison count: 4949 |
| - AGAINST | | Time taken: 913900 ns | - Comparison count: 4115 |
| - AMBIDEXTROUS | | Time taken: 1194300 ns | - Comparison count: 4965 |
| - BENEFACTOR | | Time taken: 423600 ns | - Comparison count: 1940 |
| - BIFOCAL | | Time taken: 268000 ns | - Comparison count: 1282 |
| - BUREAUCRAT | | Time taken: 452800 ns | - Comparison count: 2354 |
| - CANNERY | | Time taken: 17300 ns | - Comparison count: 55 |
| - CHEF | | Time taken: 1038000 ns | - Comparison count: 7147 |
| - CITYSCAPE | | Time taken: 152400 ns | - Comparison count: 823 |
| - CLERGY | | Time taken: 544100 ns | - Comparison count: 3805 |
| - CREAKING | | Time taken: 47700 ns | - Comparison count: 317 |
| - DEGREE | | Time taken: 454400 ns | - Comparison count: 3263 |
| - DESPAIRED | | Time taken: 479400 ns | - Comparison count: 3393 |
| - DOORKNOB | | Time taken: 349400 ns | - Comparison count: 2547 |
| - DRAMATIZING | | Time taken: 800500 ns | - Comparison count: 5182 |
| - DRAT | | Time taken: 48000 ns | - Comparison count: 359 |
| - EXACERBATE | | Time taken: 477300 ns | - Comparison count: 4097 |
| - EXCEEDING | | Time taken: 253000 ns | - Comparison count: 3273 |
| - EXERTING | | Time taken: 125300 ns | - Comparison count: 1159 |
| - FEATURE | | Time taken: 32300 ns | - Comparison count: 236 |
| - FEEBLY | | Time taken: 119200 ns | - Comparison count: 1178 |
| - FONDLING | | Time taken: 383700 ns | - Comparison count: 5930 |
| - GRAVITATE | | Time taken: 357600 ns | - Comparison count: 5362 |
| - GRAVY | | Time taken: 309000 ns | - Comparison count: 6093 |
| - HIND | | Time taken: 216200 ns | - Comparison count: 4280 |
| - INDUCED | | Time taken: 41800 ns | - Comparison count: 599 |
| - JEEP | | Time taken: 144100 ns | - Comparison count: 2917 |
| - LIKEN | | Time taken: 180600 ns | - Comparison count: 3218 |
| - MEMENTO | | Time taken: 246900 ns | - Comparison count: 4502 |
| - MISTY | | Time taken: 52700 ns | - Comparison count: 808 |
| - OBSESSIVE | | Time taken: 335200 ns | - Comparison count: 5640 |
| - PARADIGM | | Time taken: 172800 ns | - Comparison count: 3021 |
| - PILGRIM | | Time taken: 373400 ns | - Comparison count: 6254 |
| - PROCTORS | | Time taken: 299800 ns | - Comparison count: 5432 |
| - PUNISHING | | Time taken: 181000 ns | - Comparison count: 3048 |
| - REFINER | | Time taken: 196000 ns | - Comparison count: 3618 |
| - REPROVE | | Time taken: 894300 ns | - Comparison count: 4757 |
| - RESEMBLE | | Time taken: 713900 ns | - Comparison count: 4697 |
| - REVERE | | Time taken: 765300 ns | - Comparison count: 7243 |
| - SCAPEGOAT | | Time taken: 38700 ns | - Comparison count: 257 |
| - SINCERITY | | Time taken: 422900 ns | - Comparison count: 4815 |
| - SPARTAN | | Time taken: 506000 ns | - Comparison count: 5151 |
| - SURPASS | | Time taken: 642800 ns | - Comparison count: 6480 |
| - SYNCHRONIZE | | Time taken: 172000 ns | - Comparison count: 1244 |
| - TURPENTINE | | Time taken: 666200 ns | - Comparison count: 5455 |
| - WAKING | | Time taken: 446800 ns | - Comparison count: 5249 |
| - WINDSWEPT | | Time taken: 354500 ns | - Comparison count: 3843 |
| ===== | | | |
| Total Solving Time | | = 18820000 ns | |
| Total Comparison | | = 166366 | |
| Average Solving Time | | = 392083 ns | |
| Average Comparison Count | | = 3465 | |

D. TAUTAN KODE PROGRAM

Link GitHub: <https://github.com/HanselTanoto/Word-Search-Puzzle-Solver>

E. CHECKBOX

| Poin | Ya | Tidak |
|--|----|-------|
| 1. Program berhasil dikompilasi tanpa kesalahan (<i>no syntax error</i>) | ✓ | |
| 2. Program berhasil <i>running</i> | ✓ | |
| 3. Program dapat membaca file masukan dan menuliskan luaran | ✓ | |
| 4. Program berhasil menemukan semua kata di dalam puzzle | ✓ | |