Laporan Tugas Kecil 1

IF2211 Strategi Algoritma

Penyelesaian *Word Search Puzzle* dengan *Algoritma Brute Force*

oleh

Raden Rifqi Rahman (13520166)

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2022

Penyelesaian *Word Search Puzzle* dengan *Algoritma Brute Force*

Algoritma *brute force* untuk menyelesaikan *word search puzzle*

Untuk menyelesaikan permainan *word search puzzle*, dilakukan langkah-langkah sebagai berikut.

Nyatakan *word search puzzle* sebagai sebuah matriks karakter, yaitu sebuah *list* dua dimensi yang elemennya diawali dengan indeks 0. Untuk setiap "arah" pencarian yang mungkin, yaitu

1. mendatar kanan,
2. mendatar kiri,
3. menurun,
4. menaik,
5. diagonal kanan menurun,
6. diagonal kanan menaik,
7. diagonal kiri menurun, dan
8. diagonal kiri menaik;

hitung rentang indeks karakter pada matriks karakter yang mungkin sebagai huruf awal dari kata yang dicari. Perhitungan dilakukan berdasarkan panjangnya kata atau banyak huruf di dalam kata yang dicari. Sebagai contoh, apabila *puzzle* memuat 4 × 4 huruf dan akan dicari kata yang memuat 3 huruf dengan arah mendatar kanan, maka rentang indeks baris untuk huruf pertama yang mungkin adalah 0 hingga 3, sedangkan rentang indeks kolom untuk huruf pertama yang mungkin adalah 0 dan 1.

Selanjutnya, hitung juga kenaikan (*increment*) indeks baris dan indeks kolom sesuai dengan arah pencarian kata. Sebagai contoh, *increment* indeks baris dan kolom untuk arah diagonal kiri menurun berturut-turut adalah 1 (menurun) dan -1 (ke kiri). Setelah dilakukan perhitungan tersebut, untuk setiap indeks baris dan kolom pada rentang indeks yang mungkin untuk huruf pertama dari kata yang dicari pada matriks karakter, lakukan pencocokan *string*, yaitu membandingkan satu demi satu karakter pada kata yang dicari dan karakter pada matriks karakter dengan melakukan *increment* indeks baris dan kolom matriks sebesar yang telah dihitung untuk setiap kali pencocokan hingga ditemukan ketidakcocokan karakter atau kecocokan semua karakter pada kata yang dicari dan matriks karakter. Sebagai contoh, jika terdapat kata yang memuat 3 huruf dan akan dilakukan pencarian pada matriks karakter 4 × 4 pada arah diagonal kanan menurun (*increment* 1 untuk baris dan kolom) dengan huruf awal berada di indeks (0, 1); maka huruf pertama dicocokkan dengan karakter pada indeks (0, 1), huruf kedua dicocokkan dengan karakter pada indeks (1, 2), dan huruf ketiga dicocokkan dengan karakter pada indeks (2, 3).

Source code program

Berikut adalah *source code* program dalam bahasa Java yang telah diminifikasi untuk menyelesaikan *word search puzzle* dengan algoritma *brute force*.

Main.java

```java
package id.ac.itb.stei.informatika.wsp;
import id.ac.itb.stei.informatika.wsp.type.*;
import id.ac.itb.stei.informatika.wsp.io.*;
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        System.out.println();
        String path = "";
        boolean optimized = false;
        switch (args.length) {
            case 0:
                System.out.print("Input file path: ");
                Scanner scanner = new Scanner(System.in);
                boolean valid = false;
                path = scanner.nextLine();
                while (!valid) {
                    System.out.print("Use optimization ? (Y/n) ");
                    String choice = scanner.nextLine();
                    if (choice.equals("Y") || choice.equals("y")) {
                        optimized = true;
                        valid = true;
                    } else if (choice.equals("N") || choice.equals("n")) {
                        valid = true;
                    }
                }
                System.out.println();
                break;
            case 2:
                String option = args[1];
                if (option.equals("--optimize")) {
                    optimized = true;
                } else {
                    Main.printInvalidArgsErrorMsg();
                }
            case 1:
                path = args[0];
                break;
            default:
                Main.printInvalidArgsErrorMsg();
        }
        FileReader reader = new FileReader();
        String fileContent = "";
        if (reader.readFile(path)) {
            fileContent = reader.result();
        } else {
            System.out.println("File not found.");
            System.exit(1);
        }
        InputParser parser = new InputParser();
        if (parser.parse(fileContent)) {
            Matrix<Character> puzzle = parser.getPuzzle();
            String[] words = parser.getWords();
            long startTime = System.nanoTime();
            Solver solver = new Solver(puzzle);
            if (optimized) {
                solver.optimize();
            }
```

```
            ColoredMatrix<Character> solution = new ColoredMatrix<>(puzzle);
            int[] comparisons = new int[words.length];
            Color[] colors = new Color[words.length];
            for (int i = 0; i < words.length; i++) {
                String word = words[i];
                SolveResult result = solver.search(word);
                Color color = Color.random();
                if (!result.foundAt().equals(Solver.UNDEFINED_COORD)) {
                    solution.colorize(
                            result.foundAt(),
                            result.direction(),
                            word.length(),
                            color
                            );
                    colors[i] = color;
                } else {
                    colors[i] = Color.PLAIN;
                }
                comparisons[i] = result.comparisons();
            }
            long endTime = System.nanoTime();
            long execTimeNs = endTime - startTime;
            double execTimeMs = (double) execTimeNs / 1000000;
            double execTimeSeconds = execTimeMs / 1000;
            System.out.println("SOLUTION");
            System.out.println(solution);
            System.out.println("SUMMARY");
            int totalComparisons = 0;
            for (int i = 0; i < words.length; i++) {
                System.out.print(colors[i].wrap(words[i]) + " - " + comparisons[i]
+ " comparisons");
                if (colors[i] == Color.PLAIN) {
                    System.out.print(" (not found)");
                }
                System.out.println();
                totalComparisons += comparisons[i];
            }
            System.out.println("Total comparisons : " + totalComparisons + "
comparisons");
            System.out.println("Execution time    : " + execTimeMs + " ms");
            System.out.println("                    " + execTimeSeconds + " s");
        } else {
            System.out.println("Invalid input format.");
            System.exit(1);
        }
    }
    private static void printInvalidArgsErrorMsg() {
        System.out.println("\nUsage : [input-file-path [--optimize]]\n");
        System.out.println("  --optimize\t\t\tOptimize solving algorithm using" +
                " heuristic technique.");
        System.exit(0);
    }
}
```

Solver.java

```
package id.ac.itb.stei.informatika.wsp;
import id.ac.itb.stei.informatika.wsp.type.*;
```

```java
public class Solver {
    public static final Point UNDEFINED_COORD = new Point(-1, -1);

    private static final Direction[] DIRECTIONS = {
            Direction.RIGHTWARDS,
            Direction.DOWNWARDS,
            Direction.RIGHT_DOWNWARDS,
            Direction.RIGHT_UPWARDS,
            Direction.UPWARDS,
            Direction.LEFTWARDS,
            Direction.LEFT_UPWARDS,
            Direction.LEFT_DOWNWARDS,
    };
    private final Matrix<Character> puzzle;
    private int comparisons = 0;
    private boolean optimization = false;
    public Solver(Matrix<Character> puzzle) {
        this.puzzle = puzzle;
    }
    public void optimize() {
        this.optimization = true;
    }
    public SolveResult search(String string) {
        this.comparisons = 0;
        Point res = Solver.UNDEFINED_COORD;
        Direction direction = Direction.LEFTWARDS;
        for (Direction dir: Solver.DIRECTIONS) {
            direction = dir;
            res = this.searchAtDirection(string, dir);
            if (!res.equals(Solver.UNDEFINED_COORD)) {
                break;
            }
        }
        return new SolveResult(res, direction, this.comparisons);
    }
    private Range getHorizontalRange(int length, Direction direction) {
        if (direction.isHorizontallyStatic() || !this.optimization) {
            return new Range(0, this.puzzle.cols() - 1);
        } else {
            int increment = length - 1;
            int start = direction.isHorizontallyLeftwards()
                    ? increment
                    : 0;
            int end = this.puzzle.cols() - 1;
            end -= direction.isHorizontallyRightwards()
                    ? increment
                    : 0;
            return new Range(start, end);
        }
    }
    private Range getVerticalRange(int length, Direction direction) {
        if (direction.isVerticallyStatic() || !this.optimization) {
            return new Range(0, this.puzzle.rows() - 1);
        } else {
            int increment = length - 1;
            int start = direction.isVerticallyUpwards()
                    ? increment
                    : 0;
```

```
            int end = this.puzzle.rows() - 1;
            end -= direction.isVerticallyDownwards()
                    ? increment
                    : 0;
            return new Range(start, end);
        }
    }
    private Point searchAtDirection(String string, Direction direction) {
        Range hRange = this.getHorizontalRange(string.length(), direction);
        Range vRange = this.getVerticalRange(string.length(), direction);
        for (int i : vRange.toArray()) {
            for (int j : hRange.toArray()) {
                boolean match = this.matchAt(string, new Point(i, j), direction);
                if (match) {
                    return new Point(i, j);
                }
            }
        }
        return Solver.UNDEFINED_COORD;
    }
    private boolean matchAt(String string, Point coord, Direction direction) {
        int hIncrement = direction.isHorizontallyStatic()
                ? 0
                : direction.isHorizontallyRightwards()
                ? 1
                : -1;
        int vIncrement = direction.isVerticallyStatic()
                ? 0
                : direction.isVerticallyDownwards()
                ? 1
                : -1;
        int length = string.length();
        int x = coord.x();
        int y = coord.y();
        int i = 0;
        boolean match = true;
        while (match && i < length) {
            if (!this.puzzle.hasIndices(x, y)) {
                match = false;
                break;
            }
            char charToSearch = string.charAt(i);
            char charInPuzzle = this.puzzle.get(x, y);
            this.comparisons++;
            if (charToSearch == charInPuzzle) {
                i++;
                x += vIncrement;
                y += hIncrement;
            } else {
                match = false;
            }
        }
        return match;
    }
}
```

SolveResult.java

```java
package id.ac.itb.stei.informatika.wsp;
import id.ac.itb.stei.informatika.wsp.type.Direction;
import id.ac.itb.stei.informatika.wsp.type.Point;
public record SolveResult(Point foundAt, Direction direction, int comparisons) {
}
```

type/Matrix.java

```java
package id.ac.itb.stei.informatika.wsp.type;
import java.util.ArrayList;
public class Matrix<T> {
    private final int rows;
    private final int cols;
    ArrayList<ArrayList<T>> values;
    public Matrix(int rows, int cols) {
        this.rows = rows;
        this.cols = cols;
        this.values = new ArrayList<>(this.rows);
        for (int i = 0; i < this.rows; i++) {
            ArrayList<T> row = new ArrayList<>(this.cols);
            for (int j = 0; j < this.cols; j++) {
                row.add(null);
            }
            this.values.add(row);
        }
    }
    @Override
    public String toString() {
        String res = "";
        for (int i = 0; i < this.rows; i++) {
            for (int j = 0; j < this.cols; j++) {
                if (j != 0) {
                    res += " ";
                }
                res += this.get(i, j).toString();
            }
            res += "\n";
        }
        return res;
    }
    public int rows() {
        return this.rows;
    }
    public int cols() {
        return this.cols;
    }
    public boolean hasIndices(int rowIndex, int colIndex) {
        return rowIndex >= 0 && rowIndex < this.rows &&
                colIndex >= 0 && colIndex < this.cols;
    }
    public void set(int rowIndex, int colIndex, T element) {
        this.values.get(rowIndex).set(colIndex, element);
    }
    public T get(int rowIndex, int colIndex) {
        return this.values.get(rowIndex).get(colIndex);
    }
}
```

type/Point.java

```java
package id.ac.itb.stei.informatika.wsp.type;
public class Point {
    private int x;
    private int y;
    public Point(int x, int y) {
        this.x = x;
        this.y = y;
    }
    @Override
    public String toString() {
        return "(" + this.x + ", " + this.y + ")";
    }
    public boolean equals(Point point) {
        return this.x == point.x() && this.y == point.y();
    }
    public int x() {
        return this.x;
    }
    public int y() {
        return this.y;
    }
}
```

type/Range.java

```java
package id.ac.itb.stei.informatika.wsp.type;
public class Range {
    public int start;
    public int end;
    public Range(int start, int end) {
        this.start = start;
        this.end = end;
    }
    public int[] toArray() {
        int[] list = new int[this.end - this.start + 1];
        for (int i = start; i <= end; i++) {
            list[i - start] = i;
        }
        return list;
    }
}
```

type/Direction.java

```java
package id.ac.itb.stei.informatika.wsp.type;
public enum Direction {
    LEFTWARDS,
    RIGHTWARDS,
    DOWNWARDS,
    UPWARDS,
    LEFT_DOWNWARDS,
    RIGHT_DOWNWARDS,
    LEFT_UPWARDS,
    RIGHT_UPWARDS;
    public boolean isHorizontallyLeftwards() {
```

```
        return this == Direction.LEFTWARDS || this == Direction.LEFT_DOWNWARDS
                || this == Direction.LEFT_UPWARDS;
    }
    public boolean isHorizontallyRightwards() {
        return !this.isHorizontallyLeftwards() &&
                !this.isHorizontallyStatic();
    }
    public boolean isHorizontallyStatic() {
        return this == Direction.DOWNWARDS || this == Direction.UPWARDS;
    }
    public boolean isVerticallyDownwards() {
        return this == Direction.DOWNWARDS || this == Direction.LEFT_DOWNWARDS
                || this == Direction.RIGHT_DOWNWARDS;
    }
    public boolean isVerticallyUpwards() {
        return !this.isVerticallyDownwards() &&
                !this.isVerticallyStatic();
    }
    public boolean isVerticallyStatic() {
        return this == Direction.LEFTWARDS || this == Direction.RIGHTWARDS;
    }
}
```

type/Color.java

```
package id.ac.itb.stei.informatika.wsp.type;
import java.util.Random;
public enum Color {
    PLAIN,
    GREEN,
    RED,
    BLUE,
    MAGENTA,
    CYAN,
    LIME,
    BROWN,
    PURPLE,
    ORANGE;
    private static final Color[] colors = {
            Color.GREEN,
            Color.RED,
            Color.BLUE,
            Color.MAGENTA,
            Color.CYAN,
            Color.LIME,
            Color.BROWN,
            Color.PURPLE,
            Color.ORANGE
    };
    private static final Random randomizer = new Random();
    public static Color random() {
        int index = Color.randomizer.nextInt(Color.colors.length);
        return Color.colors[index];
    }
    public String wrap(String filler) {
        if (this == Color.PLAIN) {
            return filler;
        }
```

```
        String result = "";
        int code = switch (this) {
            case GREEN -> 10;
            case RED -> 9;
            case BLUE -> 45;
            case MAGENTA -> 13;
            case CYAN -> 14;
            case LIME -> 48;
            case BROWN -> 94;
            case PURPLE -> 93;
            case ORANGE -> 202;
            default -> -1;
        };
        result += "\033[1;38;5;" + code + "m";
        result += filler;
        result += "\033[0m";
        return result;
    }
}
```

type/ColoredMatrix.java

```
package id.ac.itb.stei.informatika.wsp.type;
public class ColoredMatrix<T> {
    private final Matrix<T> matrix;
    private final Matrix<Color> colorScheme;
    public ColoredMatrix(Matrix<T> matrix) {
        this.matrix = matrix;
        this.colorScheme = new Matrix<>(matrix.rows(), matrix.cols());
        for (int i = 0; i < matrix.rows(); i++) {
            for (int j = 0; j < matrix.cols(); j++) {
                this.colorScheme.set(i, j, Color.PLAIN);
            }
        }
    }
    public void colorize(Point coord, Direction direction, int length, Color color)
{
        int hIncrement = direction.isHorizontallyStatic()
                ? 0
                : direction.isHorizontallyRightwards()
                ? 1
                : -1;
        int vIncrement = direction.isVerticallyStatic()
                ? 0
                : direction.isVerticallyDownwards()
                ? 1
                : -1;
        int x = coord.x();
        int y = coord.y();
        for (int i = 0; i < length; i++) {
            this.colorScheme.set(x, y, color);
            x += vIncrement;
            y += hIncrement;
        }
    }
    @Override
    public String toString() {
        String res = "";
```

```
        for (int i = 0; i < this.matrix.rows(); i++) {
            for (int j = 0; j < this.matrix.cols(); j++) {
                if (j != 0) {
                    res += " ";
                }
                String filler = this.matrix.get(i, j).toString();
                res += this.colorScheme.get(i, j).wrap(filler);
            }
            res += "\n";
        }
        return res;
    }
}
```

io/FileReader.java

```
package id.ac.itb.stei.informatika.wsp.io;
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;
public class FileReader {
    private String result = "";
    public FileReader() {
    }
    public boolean readFile(String path) {
        this.result = "";
        try {
            File file = new File(path);
            Scanner reader = new Scanner(file);
            if (reader.hasNextLine()) {
                this.result += reader.nextLine();
            }
            while (reader.hasNextLine()) {
                this.result += "\n" + reader.nextLine();
            }
        } catch (FileNotFoundException e) {
            return false;
        }
        return true;
    }
    public String result() {
        return this.result;
    }
}
```

io/InputParser.java

```
package id.ac.itb.stei.informatika.wsp.io;
import id.ac.itb.stei.informatika.wsp.type.Matrix;
public class InputParser {
    private Matrix<Character> puzzle = null;
    private String[] words;
    public InputParser() {}
    public boolean parse(String input) {
        String[] chunks = input.split("\n\n");
        if (chunks.length != 2) {
            return false;
        } else {
```

```
            String rawPuzzle = chunks[0];
            CharacterMatrixParser cmp = new CharacterMatrixParser();
            if (cmp.parse(rawPuzzle)) {
                this.puzzle = cmp.result();
            } else {
                return false;
            }
            String rawWords = chunks[1];
            return this.parseWords(rawWords);
        }
    }
    private boolean parseWords(String rawWords) {
        this.words = rawWords.split("\n");
        for (String word: this.words) {
            if (word.equals("")) {
                return false;
            }
        }
        return true;
    }
    public Matrix<Character> getPuzzle() {
        return this.puzzle;
    }
    public String[] getWords() {
        return this.words;
    }
}
```

io/CharacterMatrixParser.java

```
package id.ac.itb.stei.informatika.wsp.io;
import id.ac.itb.stei.informatika.wsp.type.Matrix;
public class CharacterMatrixParser {
    private Matrix<Character> result = null;
    public boolean parse(String input) {
        String[] rawRowArray = input.split("\n");
        int rows = rawRowArray.length;
        int cols = rawRowArray[0].split("\s").length;
        this.result = new Matrix<>(rows, cols);
        boolean failed = false;
        filling:
        for (int i = 0; i < rows; i++) {
            String[] rawColArray = rawRowArray[i].split("\s");
            for (int j = 0; j < cols; j++) {
                char[] charArray = rawColArray[j].toCharArray();
                if (charArray.length != 1) {
                    failed = true;
                    break filling;
                } else {
                    this.result.set(i, j, charArray[0]);
                }
            }
        }
        return !failed;
    }
    public Matrix<Character> result() {
        return this.result;
```

```
    }
}
```

Contoh penggunaan program

Program tersebut telah diuji dengan 9 kasus uji, yaitu 3 *puzzle* ukuran kecil (14 × 16 karakter, 14 kata), 3 *puzzle* ukuran sedang (20 × 22 karakter, 20 kata), dan 3 *puzzle* ukuran besar (32 × 34 karakter, 48 kata). Berikut adalah *input* dan *output* dalam menjalankan program.

1. *Puzzle* ukuran kecil 1

wsp-small-1.txt

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

```
ALPHA
BECALM
GAIT
HOMELY
HYPE
JOCKEY
KNIT
OXTAIL
QUIET
SELF
SPORTY
TEMPT
TWIN
UPRISE
```

Gambar 1. *Input puzzle* ukuran kecil 1 dan kata-kata yang dicari.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-small-1.txt
SOLUTION
M P Q W V Q S O J E T O X G E P
O X T A I L S P J M C T P Q R U
D O W X P M Y E K C O J S B R Z
I O H E L C W W T E I C F B Q V
O L K A H M N D I T I V Y L W S
H Y C D B U N P Y U P R I S E B
W E R I C N F H X O R C T M J S
B M U N A T J R N P K A S I C Q
W M T M H P J U O S T J Y H N Q
E J D T P M Y P A P I W Y B Q K
T I A G L E D W W O P P I S W A
D S Q F A T B E Q R E I I N J R
Y L E M O H U U N T P U X D Q K
G D W B T E I U Q Y L O H O F I
```

Gambar 2. *Output* solusi *puzzle* ukuran kecil 1.

```
SUMMARY
ALPHA - 719 comparisons
BECALM - 438 comparisons
GAIT - 965 comparisons
HOMELY - 808 comparisons
HYPE - 1314 comparisons
JOCKEY - 702 comparisons
KNIT - 1117 comparisons
OXTAIL - 18 comparisons
QUIET - 925 comparisons
SELF - 1099 comparisons
SPORTY - 309 comparisons
TEMPT - 743 comparisons
TWIN - 510 comparisons
UPRISE - 71 comparisons
Total comparisons : 9738 comparisons
Execution time    : 4.4382 ms
                    0.0044382 s
```

Gambar 3. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran kecil 1.

2. *Puzzle* ukuran kecil 2

wsp-small-2.txt

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

ANEW
BRINE
COCKY
EXCEL
FOCAL
IDLING
JOWL
LOAM
MAINE
ROME
SUITE
THRILL
TOXIC
TUBE
```

Gambar 4. *Input puzzle* ukuran kecil 2 dan kata-kata yang dicari.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-small-2.txt
SOLUTION
P F Y W W K M Z E X O L Q S B H
G K E X S S X A K K S X I K O J
T N T T T D F H I U R X J N C X
A F N R L R C C I N B A J H S D
K B R S F S Q T R O E C O B Z M
L L O A M Z E E B U T C O C K Y
S E N R S J G K B V L E B I U F
I M C A L N N P T L K J V Y Q Z
B O W X I K H H I A P L A C O F
K A A L E L W R E Y K S P T V L
W P D R P C H U N M P P O O W V
R I O I A T G I I G U X J O W D
S M Q X E N X U R Z I F J X U Y
E W X Q E Q G G B C T U E S M C
```

Gambar 5. *Output* solusi *puzzle* ukuran kecil 2.

```
SUMMARY
ANEW - 533 comparisons
BRINE - 758 comparisons
COCKY - 80 comparisons
EXCEL - 1029 comparisons
FOCAL - 864 comparisons
IDLING - 495 comparisons
JOWL - 657 comparisons
LOAM - 73 comparisons
MAINE - 348 comparisons
ROME - 1347 comparisons
SUITE - 1095 comparisons
THRILL - 499 comparisons
TOXIC - 1192 comparisons
TUBE - 935 comparisons
Total comparisons : 9905 comparisons
Execution time    : 4.2818 ms
                    0.0042818 s
```

Gambar 6. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran kecil 2.

3. *Puzzle* ukuran kecil 3

wsp-small-3.txt

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

ABOVE
ASPECT
BECAME
CRAVAT
DIVINE
LITTLE
LOST
MOLE
PLUTO
SHADY
SHINE
SONG
STUN
UPSHOT
```

Gambar 7. *Input puzzle* ukuran kecil 3 dan kata-kata yang dicari.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-small-3.txt
SOLUTION
P L U T O F Q X P G N O S X N D
M L X Y E G F S Y Q B Q A O I O
L B Z D P B E R I I Q K C V L F
R E M N U L A I J D V R I T X Q
Q N T T Y M J C X V A N L C W T
B K Y E L T T I L V E B E F D P
S H I N E D Q P A R T X T R H V
J Z U V F E K T E N P F C Y S E
J H O D B F B U T W U A E L H L
U B L M T E G O C X X T P W A B
A M Q T T G C X U F O B S N D U
S A T O X W U A E L O M A H Y K
M B Q A T T F E M T S O L B R A
I S U U P S H O T E L Z Q H W M
```

Gambar 8. *Output* solusi *puzzle* ukuran kecil 3.

```
SUMMARY
ABOVE - 547 comparisons
ASPECT - 636 comparisons
BECAME - 417 comparisons
CRAVAT - 950 comparisons
DIVINE - 937 comparisons
LITTLE - 745 comparisons
LOST - 1049 comparisons
MOLE - 1010 comparisons
PLUTO - 5 comparisons
SHADY - 310 comparisons
SHINE - 78 comparisons
SONG - 866 comparisons
STUN - 1145 comparisons
UPSHOT - 161 comparisons
Total comparisons : 8856 comparisons
Execution time    : 4.0494 ms
                    0.004049400000000001 s
```

Gambar 9. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran kecil 3.

4. *Puzzle* ukuran sedang 1

wsp-medium-1.txt

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

```
ABASH
ASSERTING
ASTHMA
CELSIUS
DECOMPRESS
DESPISING
DRAUGHT
GLANCE
LAUGH
LEES
LEFTY
MIGHT
NEWSFLASH
OSCAR
PASSPORT
SLIME
SMUG
SOURCELESS
THESAURUS
WARPED
```

Gambar 10. *Input puzzle* ukuran sedang 1 dan kata-kata yang dicari.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-medium-1.txt
SOLUTION
O H E C W W T E I C B Q V O L K H M N D I T
A S S E R T I N G I V Y W S H Y D B U N P Y
G B W R I C N F H S X O R C M J M U N J R N
P L K A S C Q S S E L E C R U O S W M T M J
U O A J Y R Q E E J G N I S I P S E D D T Y
P A I N A B R Q D W W P S W A D S Q F B E Q
I I J C C P R G U U N P U X D Q K G D W B L
O H S O M E F U I C V D H N E W S F L A S H
U O X O U T P M J W F W F G O Z B C N L C G
X G C D D Y T S C B B I X K L Y V Y N V S K
G E A N Y K R G T P G R Y L X A Y P T U O D
D F J A A M O K H C S R W B V A R R U F N R
D R S U M I P A G W Q U S C O Y B B H D E Z
Q H X J H G S S U F E W R G N X S A C X L L
C U E Z T H S U A H W F A U L A L T S C K I
Z B G E S T A G R G D Q X R A Y I F U H Q R
X L X R A D P Q D U O K Y O P S M P N G K J
A U F B K P B F E A Q L P S L E E R K V P F
Y W K Z E X O L Q L S B H G K X D H S S X K
K X I K O S U I S L E C J T L E E S T T T T
```

Gambar 11. *Output* solusi *puzzle* ukuran sedang 1.

```
SUMMARY
ABASH - 982 comparisons
ASSERTING - 23 comparisons
ASTHMA - 1522 comparisons
CELSIUS - 1781 comparisons
DECOMPRESS - 709 comparisons
DESPISING - 1268 comparisons
DRAUGHT - 1368 comparisons
GLANCE - 741 comparisons
LAUGH - 1673 comparisons
LEES - 397 comparisons
LEFTY - 2275 comparisons
MIGHT - 628 comparisons
NEWSFLASH - 125 comparisons
OSCAR - 1128 comparisons
PASSPORT - 1232 comparisons
SLIME - 720 comparisons
SMUG - 1667 comparisons
SOURCELESS - 1168 comparisons
THESAURUS - 1657 comparisons
WARPED - 954 comparisons
Total comparisons : 22018 comparisons
Execution time    : 5.3188 ms
                    0.0053188 s
```

Gambar 12. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran sedang 1.

5. *Puzzle* ukuran sedang 2

wsp-medium-2.txt

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

```
BEEPER
BLASPHEMER
CUBIC
CUTLERY
DEFLECTION
DEPOSITED
FUSION
GARMENT
GENTLEFOLK
GUILLOTINE
HAYFEVER
HAZINESS
MORAL
PAVILION
REGULATE
RELIABLE
RUEFULLY
SPLITTING
SPRINGIEST
TROUNCED
```

Gambar 13. *Input puzzle* ukuran sedang 2 dan kata-kata yang dicari.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-medium-2.txt
SOLUTION
G H A Y F E V E R U L Y R E L T U C A F K E
X U N S P R I N G I E S T G T X F A U F O U
W I I O Z W Q F V J J A O K D I X F Y D W X
L F Q L O Q L A W D G H R O Y P U K V T D N
A A T B L A S P H E M E R I X M T B Q A R Z
N S G V Q O P G V L T J A R U E F U L L Y M
Q K J A H P T W G E N T L E F O L K L Y J N
A Z W L S N O I L I V A P I P M O B J G Y W
Z P P B K H M C N W H U Y P F H O W G H S W
G G A R M E N T E E L X O A Z B Q G E D G M
Y B C I H N M K Y P H M L K Y O V E U U P W
F I Y G Y S R D U M E A D A B L W L D S A F
U V N O I T C E L F E D E U R U R B H P U X
S V G M A B A J G S K E P H B O E A T L C D
I B N J U E X D A U E V O G E O M I J I I O
O R W Y Y C D O U X L B S W E V K L O T B R
N L H Y Z S P H D F P A I M P J F E M T U R
Y R Z G N G X M I N O I T J E E H R F I C X
T U Q R D E C N U O R T E E R H T B S N W I
B M A V H A Z I N E S S D Q O P O U Q G S T
```

Gambar 14. *Output* solusi *puzzle* ukuran sedang 2.

```
SUMMARY
BEEPER - 666 comparisons
BLASPHEMER - 65 comparisons
CUBIC - 1628 comparisons
CUTLERY - 1429 comparisons
DEFLECTION - 1245 comparisons
DEPOSITED - 562 comparisons
FUSION - 610 comparisons
GARMENT - 160 comparisons
GENTLEFOLK - 101 comparisons
GUILLOTINE - 538 comparisons
HAYFEVER - 9 comparisons
HAZINESS - 309 comparisons
MORAL - 2293 comparisons
PAVILION - 1447 comparisons
REGULATE - 800 comparisons
RELIABLE - 1277 comparisons
RUEFULLY - 102 comparisons
SPLITTING - 573 comparisons
SPRINGIEST - 26 comparisons
TROUNCED - 1603 comparisons
Total comparisons : 15443 comparisons
Execution time    : 5.6806 ms
                    0.0056806 s
```

Gambar 15. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran sedang 2.

6. *Puzzle* ukuran sedang 3

wsp-medium-3.txt

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

```
ADHERENCE
AMUSE
BIBLICAL
BRAID
BRINY
DISHONEST
FINAL
INERTNESS
INSCRIBED
IRREGULAR
JUNE
PERJURY
PRESCRIBED
RECOVER
SACKING
SHOWER
SOLID
SPATULA
TOTALLY
URUGUAY
```

Gambar 16. *Input puzzle* ukuran sedang 3 dan kata-kata yang dicari.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-medium-3.txt
SOLUTION
D Z N F B F F F F Y P T H R U X Y B R I N Y
V G C B X D X I R R E G U L A R J Y J I R R
X J L R G M N U U T M B O H J B X T J H F E
I B W G K D W P U S O L I D F L T Q U E E C
I I N E R T N E S S Z E S F W B T D A D C O
O O N F C H B L K D G C V N G Z X E D G D V
S M H I N V N A B C O N C N Y S M B L X I E
O J W R D N C N H O U E J W T M J I L P A R
J Q O M V E A I C B F R G K S L D R O W R U
F Y E N U J I F D P A E Y Y N K A C G X B P
P R E S C R I B E D D H A R A N X S Q F K Z
C K N I I Q G H N X M D I U L R L N P K X D
R U R U G U A Y D G C A C J U R V I M L Q B
K Q C V J W N Z K S Q J V R T W B T J G V I
U H O E D T K H E O G I F E A R S R Q B I B
Y I H E R E W O H S J H Y P P G G Q T P H L
S I S A S Q D E S U M A S J S N W P M A R I
O J O T T O T A L L Y A A O T H U V S H O C
K C K C R E R R T X T O G G N I K C A S C A
T S E N O H S I D G T Z M F U P J M Z J Y L
```

Gambar 17. *Output* solusi *puzzle* ukuran sedang 3.

```
SUMMARY
ADHERENCE - 1031 comparisons
AMUSE - 2022 comparisons
BIBLICAL - 616 comparisons
BRAID - 1481 comparisons
BRINY - 23 comparisons
DISHONEST - 1493 comparisons
FINAL - 1463 comparisons
INERTNESS - 70 comparisons
INSCRIBED - 1058 comparisons
IRREGULAR - 30 comparisons
JUNE - 2038 comparisons
PERJURY - 1324 comparisons
PRESCRIBED - 143 comparisons
RECOVER - 401 comparisons
SACKING - 1777 comparisons
SHOWER - 1861 comparisons
SOLID - 68 comparisons
SPATULA - 1383 comparisons
TOTALLY - 290 comparisons
URUGUAY - 208 comparisons
Total comparisons : 18780 comparisons
Execution time    : 5.0869 ms
                    0.0050869 s
```

Gambar 18. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran sedang 3.

7. *Puzzle* ukuran besar 1

wsp-large-1.txt

```
1  N N M M O T O R E D N O I H S U C N I P G Q W S H V E Q E P G I K M
2  Z I B Q Y U N I K X L B C F G K O I D N W U E F C Y C K Z L V A B N
3  I I V W X U J V F L L J N P S X A K I I U B R U Q O N E Z T I Z G U
4  E N E X J A K M Y E W V F E Z F B K V A R I S S M K N V M U W F E M
5  X D E G F N E E N Q L P Q D M Q A H U T N V W M S I C O B W E X E U
6  M N L V A K M D Z Z Z Q H M H E H W T G B O O E T E V X E C N G N R
7  O F V L I V O D U I K U T N P W L F E F G N X S N O N G I W S B V L
8  V I M R Y T R J A T G G U S O Q M D P F Q J I U C C G S B W Z D J D
9  O H D L I F A Z Y Z S O T L C L F U D Z W L E D Q R H Z S V B X N N
10 V K R Q T S K B G E A V L I C L U X B I I Y U I N T E R F A C E Q M
11 M B I J V O L S L Y C G G O R L V E P H M D A F D M L E N V R G B K
12 D Z L O I R Y C N Y A N B V P G O M P R W Y B P M K O N W Y Z C R R
13 U W L I J E P A M H R U A N A A J T R B B Q H O U N D I N G M O Y N
14 U K E D R N W R U S F D X T H Y T P T R S M Y H R E N O V A T I O N
15 Y U D V C E F E Y K N F L O S S I N G E A M P K K R Y L E L H H H H
16 X Q T Y R S Z D E C Z T I F G I F I I Y D F L C R U N K S W X A Y Z
17 O K A B S S P L I M I B X W R S D L T C I M H T I R O G L A N B B M
18 A S Z H E W N E G V R U L J L Q S Y D E L Y Q Y R E T K O D P A W D
19 R V F Y T Q A Z K L E J E W A J Y V L V C D M E T H D R M Z T E I J
20 V J K Z F U C Y W A Q V X U N R M P R L L J C D Q C X A L N E O B S
21 R X U N Q F C Y V D R I O M E M I W X D A O C P B J D E R G D Y D M
22 D X Y B C J L M O O R A E T M T Y S C K E I Q G J E B J D E T L I W
23 W Y T Y G M A B Z T O L R S O X I C W I R V R C E E E C I D N U A J
24 L N I A T A M Z C J L F H W N K S Q N N M R O O T L A U N C H I N G
25 O L R M B K A L B J L W D S E S H I N D X V P L T X D J H Z D T V G
26 T A E N F E T T L T K O H I H N I A P E K U J A V I T M O P B E V N
27 P C M E G O I S T S H A D E P P A R T R S B F B E I D Y I B B O I Z
28 T O E R Y V O E X N V U F A C E L I F T O H W K E G N E E W V R M J
29 O X T H Y E N W G E C M Z G Z I E K U C O Q B D W M Z G K H C V M E
30 D C V M D R P O U N B E C O M I N G G A D W X Y O I V U N S P O O E
31 R Q I D M O M L V F N A U D I T E J I X A L E L B A K R O W N U H L
32 X B Q I B A M C X C K D J T X E A N T N E M T S I L N E C P Q O P E
33
```

Gambar 19. *Input puzzle* ukuran besar 1.

```
34 ACCLAMATION
35 ALGORITHMIC
36 APOLOGIZE
37 AUDIT
38 BLEND
39 CLOTTED
40 COMMON
41 CRASSNESS
42 DEVOLVING
43 DEWY
44 DISTANCE
45 DRILLED
46 EDITORIALLY
47 ENLISTMENT
48 FACELIFT
49 FLOSSING
50 FRACAS
51 FRINGED
52 GRIT
53 HANDMADE
54 HOUNDING
55 INEVITABLY
56 INTERFACE
57 JAUNDICE
58 KINDER
59 LANK
60 LAUNCHING
61 LOWEST
62 MAKEOVER
63 MEMOIR
64 MIDDLEMEN
65 MOTORED
66 PAIN
67 PHENOMENAL
68 PHILISTINE
69 PINCUSHION
70 REFILE
71 RENOVATION
72 SCARED
73 SHAVE
74 SORENESS
75 SPEAKING
76 TEAROOM
77 TEMERITY
78 TRAPPED
79 UNBECOMING
80 UNWORKABLE
81 WILTED
```

Gambar 20. Kata-kata yang dicari pada *puzzle* ukuran besar 1.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-large-1.txt
SOLUTION
N N M M O T O R E D N O I H S U C N I P G Q W S H V E Q E P G I K M
Z I B Q Y U N I K X L B C F G K O I D N W U E F C Y C K Z L V A B N
I I V W X U J V F L L J N P S X A K I I U B R U Q O N E Z T I Z G U
E N E X J A K M Y E W V F E Z F B K V A R I S S M K N V M U W F E M
X D E G F N E E N Q L P Q D M Q A H U T N V W M S I C O B W E X E U
M N L V A K M D Z Z Z Q H M H E H W T G B O O E T E V X E C N G N R
O F V L I V O D U I K U T N P W L F E F G N X S N O N G I W S B V L
V I M R Y T R J A T G G U S O Q M D P F Q J I U C C G S B W Z D J D
O H D L I F A Z Y Z S O T L C L F U D Z W L E D Q R H Z S V B X N N
V K R Q T S K B G E A V L I C L U X B I I Y U I N T E R F A C E Q M
M B I J V O L S L Y C G G O R L V E P H M D A F D M L E N V R G B K
D Z L O I R Y C N Y A N B V P G O M P R W Y B P M K O N W Y Z C R R
U W L I J E P A M H R U A N A A J T R B B Q H O U N D I N G M O Y N
U K E D R N W R U S F D X T H Y T P T R S M Y H R E N O V A T I O N
Y U D V C E F E Y K N F L O S S I N G E A M P K K R Y L E L H H H H
X Q T Y R S Z D E C Z T I F G I F I I Y D F L C R U N K S W X A Y Z
O K A B S S P L I M I B X W R S D L T C I M H T I R O G L A N B B M
A S Z H E W N E G V R U L J L Q S Y D E L Y Q Y R E T K O D P A W D
R V F Y T Q A Z K L E J E W A J Y V L V C D M E T H D R M Z T E I J
V J K Z F U C Y W A Q V X U N R M P R L L J C D Q C X A L N E O B S
R X U N Q F C Y V D R I O M E M I W X D A O C P B J D E R G D Y D M
D X Y B C J L M O O R A E T M T Y S C K E I Q G J E B J D E T L I W
W Y T Y G M A B Z T O L R S O X I C W I R V R C E E E C I D N U A J
L N I A T A M Z C J L F H W N K S Q N N M R O O T L A U N C H I N G
O L R M B K A L B J L W D S E S H I N D X V P L T X D J H Z D T V G
T A E N F E T T L T K O H I H N I A P E K U J A V I T M O P B E V N
P C M E G O I S T S H A D E P P A R T R S B F B E I D Y I B B O I Z
T O E R Y V O E X N V U F A C E L I F T O H W K E G N E E W V R M J
O X T H Y E N W G E C M Z G Z I E K U C O Q B D W M Z G K H C V M E
D C V M D R P O U N B E C O M I N G G A D W X Y O I V U N S P O O E
R Q I D M O M L V F N A U D I T E J I X A L E L B A K R O W N U H L
X B Q I B A M C X C K D J T X E A N T N E M T S I L N E C P Q O P E
```

Gambar 21. *Output* solusi *puzzle* ukuran besar 1.

```
SUMMARY
ACCLAMATION - 1459 comparisons
ALGORITHMIC - 3912 comparisons
APOLOGIZE - 4879 comparisons
AUDIT - 955 comparisons
BLEND - 6608 comparisons
CLOTTED - 2132 comparisons
COMMON - 6347 comparisons
CRASSNESS - 4846 comparisons
DEVOLVING - 2294 comparisons
DEWY - 3050 comparisons
DISTANCE - 5297 comparisons
DRILLED - 1231 comparisons
EDITORIALLY - 4841 comparisons
ENLISTMENT - 4622 comparisons
FACELIFT - 775 comparisons
FLOSSING - 413 comparisons
FRACAS - 3818 comparisons
FRINGED - 5969 comparisons
GRIT - 6212 comparisons
HANDMADE - 6044 comparisons
HOUNDING - 364 comparisons
INEVITABLY - 1749 comparisons
INTERFACE - 285 comparisons
JAUNDICE - 4666 comparisons
KINDER - 1723 comparisons
LANK - 3134 comparisons
LAUNCHING - 665 comparisons
LOWEST - 4516 comparisons
MAKEOVER - 1701 comparisons
MEMOIR - 5165 comparisons
MIDDLEMEN - 4837 comparisons
MOTORED - 11 comparisons
PAIN - 5700 comparisons
PHENOMENAL - 3422 comparisons
PHILISTINE - 2292 comparisons
PINCUSHION - 3626 comparisons
REFILE - 5564 comparisons
RENOVATION - 374 comparisons
SCARED - 1328 comparisons
SHAVE - 7348 comparisons
SORENESS - 1228 comparisons
SPEAKING - 2500 comparisons
TEAROOM - 4953 comparisons
TEMERITY - 3978 comparisons
TRAPPED - 5104 comparisons
UNBECOMING - 774 comparisons
UNWORKABLE - 4450 comparisons
WILTED - 5120 comparisons
Total comparisons : 162281 comparisons
Execution time    : 15.3371 ms
                    0.0153371 s
```

Gambar 22. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran besar 1.

8. *Puzzle* ukuran besar 2

wsp-large-2.txt

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

Gambar 23. *Input puzzle* ukuran besar 2.

```
ALERT
ANTS
AZURE
BEAVER
CENTRIST
CERVICAL
CINE
COUGHED
DECKED
DISALLOWED
DISMANTLED
EFFECTUAL
ENGARDE
FANTASIA
FOES
HEATED
HEIRESS
HONESTY
IDEALISTIC
LEADING
LENGTHENING
LITHE
MERIT
MOTORIST
OASIS
OVERSIGHT
PANDORA
PROMISING
PUNTER
QUAGMIRES
RECLAIMED
RECRUITER
REMINISCENT
RESERVATION
ROAST
SEGREGATING
SEVEN
SHORTER
SLICED
STEADFAST
STOKE
SUFFICED
UNDERCUT
VILE
VOMITED
WADI
WATCHFUL
WOLFISH
```

Gambar 24. Kata-kata yang dicari pada *puzzle* ukuran besar 2.

```
D:\Semester 4\Stima\Tugas Kecil\Tugas Kecil 1>java -jar bin/wsp.jar test/wsp-large-2.txt
SOLUTION
Y Z G K I I N W S W L Z T N P A N D O R A R P Z S Y K Z K G T N G J
V W A T C H F U L M D V P C M S J C S J A I S A T N A F L A H B B G
W V C T E S A T P W E Y W H L T J S G B O M Y B Y K M A N P K K R N
C H H U G H D Q D L T N R K N D B K H W S E R T C K T Z Y H T B E I
U Z S M E V T D U R I H R A U B C S Q L R R N I P I R L E U R J T D
F J A X I D B I O G M O V G I B I J B E Z I L D U E T A O T C H R A
I K K A H L X D L X O E X I P F Z K M Y H T R W Z I T S Y X O X O E
B D W D G M Y C B T V L D L L N U I O H M S M P D E M O I Q P P H L
R S I D E M I A L C E R M O V E N R F M J Y P A D N D B I L R D S O
Y L M S J U W C Y A N L W D Y I E U W F H B W S I L W T U W A K O T
X B Z B M N O O Y G E F X U S S I T L J P M B E G I U X Y T W E W E
J T Z J R A J L Y S V K D C E D S Y E T D P J L V A K A F Q L H D K
B U S I L J N H V Y E S E R O H C H U G N I T A G E R G E S H F B I
U Z C I F U V T S X S N V I K N J C R W A A S G Y M T C K T P H Y A
U F O L R I Q H L G T A Y I F S R P S Y T H N J F W T P D S P O G O
H F M F V T Z C A E T E B Q U E R H J O U Z C T F Z F I L A D H R M
H G S N T S N T T I D R V F D O A A O D G A F J S G L T S O P H L J
H C H T U R J E O M N Y F N M E H V M U G K C F P H Q W J R W R U I
B L X M E V W N C N J I U I R S E O M F E Y Q P V E S M B J P Z I G
H H E M B A S W M K C U S U F R T D E C K E D O S E R I M G A U Q R
V C X R H Z D Y U E Z I Z J S O F H L X L A U T C E F F E U D R H Q
K Z X N G Y D F D L N A U I R D Y X O B Q A L Q H Q B J J Q F G E V
M B P E C W E A A G J K G I P N X R D I S A L L O W E D J J P M K B
G E U H E C B N U S X H S B V D L S E D T R E L A T C K W P I P L X
K A N R K O H G I F T T D A G B D Y F C K T U K O E Z N Z G F F A A
Q V T J O U I M I C G H Y S B U O H E I R E S S Q P Y D S N P O C H
S E E L T G E N G A R D E L V W Q N G J I U C X J Y L U L Z X X O Q
Q R R F S H I Z L J K W M C E R V I C A L E I E F X W Z I A I D J A
A Y J O O E W P O B P T E K S Y Y T S E N O H T X X J W C N F E Q C
J X O E T D Z D G V O L A K H F Q C O F V D K Y E Y A N E M J Z P Y
E I S S A T O R K P H Y O F M L S F Q T M L W J B R S S D Z K G Y Q
F B P J D Z X T L S L N T S I S A O Q T X T O G N I N E H T G N E L
```

Gambar 25. *Output* solusi *puzzle* ukuran besar 2.

```
SUMMARY
ALERT - 5528 comparisons
ANTS - 2534 comparisons
AZURE - 3436 comparisons
BEAVER - 1740 comparisons
CENTRIST - 5277 comparisons
CERVICAL - 783 comparisons
CINE - 6680 comparisons
COUGHED - 1753 comparisons
DECKED - 600 comparisons
DISALLOWED - 606 comparisons
DISMANTLED - 1856 comparisons
EFFECTUAL - 4558 comparisons
ENGARDE - 788 comparisons
FANTASIA - 4128 comparisons
FOES - 1989 comparisons
HEATED - 6439 comparisons
HEIRESS - 758 comparisons
HONESTY - 5154 comparisons
IDEALISTIC - 4657 comparisons
LEADING - 3474 comparisons
LENGTHENING - 4299 comparisons
LITHE - 5858 comparisons
MERIT - 1093 comparisons
MOTORIST - 6150 comparisons
OASIS - 5722 comparisons
OVERSIGHT - 5842 comparisons
PANDORA - 21 comparisons
PROMISING - 5718 comparisons
PUNTER - 1727 comparisons
QUAGMIRES - 4319 comparisons
RECLAIMED - 4115 comparisons
RECRUITER - 2345 comparisons
REMINISCENT - 4959 comparisons
RESERVATION - 5069 comparisons
ROAST - 4250 comparisons
SEGREGATING - 3864 comparisons
SEVEN - 4153 comparisons
SHORTER - 3541 comparisons
SLICED - 1919 comparisons
STEADFAST - 2198 comparisons
STOKE - 4642 comparisons
SUFFICED - 6187 comparisons
UNDERCUT - 2819 comparisons
VILE - 2202 comparisons
VOMITED - 3375 comparisons
WADI - 3192 comparisons
WATCHFUL - 38 comparisons
WOLFISH - 2690 comparisons
Total comparisons : 165045 comparisons
Execution time    : 15.836 ms
                    0.015836 s
```

Gambar 26. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran besar 2.

9. *Puzzle* ukuran besar 3

wsp-large-3.txt

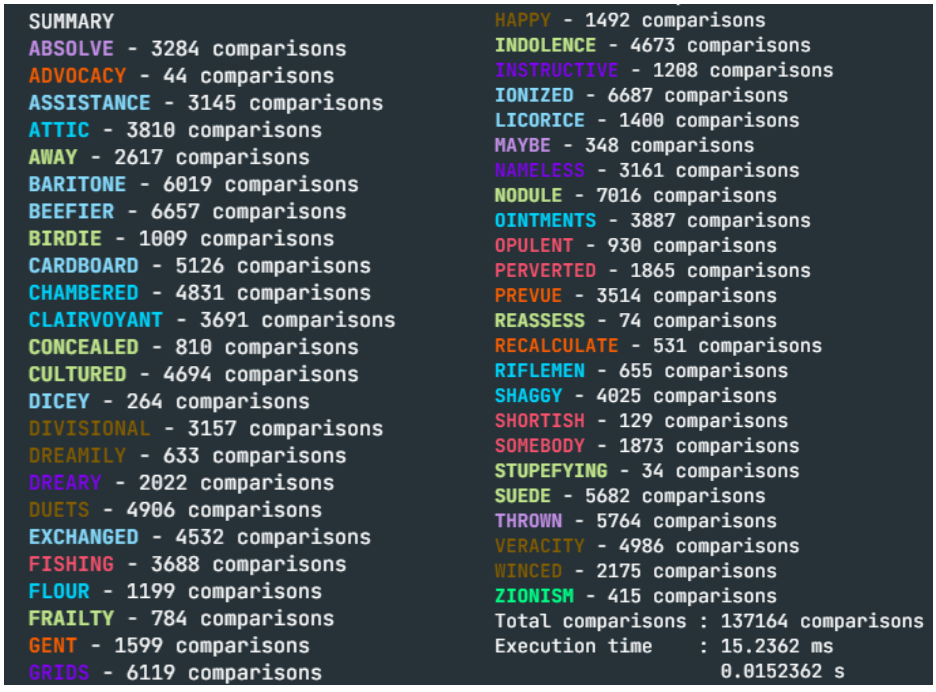
```
1  D D Q W Y G G U N X Q R F G Y C D Z Y M S H L S T U P E F Y I N G Z
2  H B B S A T P C A D V O C A C Y O B G W P U L Y F N S K R K G I Y V
3  X Q I X M H J I E K F R E A S S E S S X D R D F T C E O J L C O T M
4  A A R G P G A T K T E D P F K X C G F E S R C N Q I R I M Y X A Y E
5  V M D S H O R T I S H B E Q S J S T E U D L E N L O C H F E H E U Y
6  A N I B S G X A D G Q F Y R P D C P V L F F I A A S E A U H B E S G
7  K Y E Y E E O T A P A X S R E E D Y H H G L L S R Z V W R L S O Q X
8  B Q C M D X O M Q O G O O K T B R U S I E O S J X Y L B L E A P D Q
9  R B H D O D I C E Y O M U U V G M V Y Z C U M D J S O A A X V T P Y
10 W Z C E Q T N A Y O V R I A L C F A E O N R O W E N B T N G H H B X
11 Y I L I L R O J R U O N K N J O C E H R A D I S R F O C O N I U U W
12 N I N X M A Y B E Y V X W U R S T N Q C T E Y G M R W O I I P Y F P
13 J N W C B I B J H V I Z U O P Q V X S Z S E Z Q U W B B S H E G H S
14 V S O Q E R E U A H W Y O Y R T A Z H Z I S D G Y B A P I S R G H O
15 L T D X H D Z I O N I S M S F H Y N J W S V D T W R P T V I L A A V
16 I R U R E A R L Z K L N P D N B T S U S S W P I I Y D J I F Y H P R
17 C U A G A B W U F D V K N L S H N P K Y A Z K T R T G N D E F S P X
18 O C X E F O N Q B I C D U J P X B H B Y L W O Q F G C L K M V J Y S
19 R T U N K D B F E G K J T M I R Q H D E G N A H C X E N K H L N T P
20 I I V T J E M D I D W N O Q Q H E X Q G E C T Y E W B O U W P I Z U
21 C V E P W P W L R Z W E R E C A L C U L A T E F O U K A O I O E R E
22 E E Q O S X G F Q A E D E R U T L U C F R C L G B S I H Q Z G F H Y
23 D R E A M I L Y W P C O E B N E Y R I F L E M E N O C D P Y Y N F S
24 O M M U V K G V E G J X B W V R R F S X S K R K N F W U B A O P O T
25 J Z D Z Z L I P H F B E L L K E W S T Q K L J I Y X N R N D K H Q N
26 Q A I B E R S P W I E O O Z X R E J L B J R Z H X E N G U R U Y B E
27 L B V P L O V M X F U S P E R L L R S O E E R F R A I L T Y S J Q M
28 R P D A G K R F I O B C U U E Q Z D V G D Q H E C N E L O D N I H T
29 K B W B M L D E O A O V Z M Y Y E D E U S E H I P H U W M A X F M N
30 O A H R I P R V O N E C A Y T G C O N C E A L E D C X Y P I F C E I
31 T H P D V Y U D Y R I N I U U L P M H F D E J O X P I Y F V I X X O
32 T D S T A K B O P U L E N T U E D D G V W F G B X C K I I S U V K W
33
```

Gambar 27. *Input puzzle* ukuran besar 2.

```
34 ABSOLVE
35 ADVOCACY
36 ASSISTANCE
37 ATTIC
38 AWAY
39 BARITONE
40 BEEFIER
41 BIRDIE
42 CARDBOARD
43 CHAMBERED
44 CLAIRVOYANT
45 CONCEALED
46 CULTURED
47 DICEY
48 DIVISIONAL
49 DREAMILY
50 DREARY
51 DUETS
52 EXCHANGED
53 FISHING
54 FLOUR
55 FRAILTY
56 GENT
57 GRIDS
58 HAPPY
59 INDOLENCE
60 INSTRUCTIVE
61 IONIZED
62 LICORICE
63 MAYBE
64 NAMELESS
65 NODULE
66 OINTMENTS
67 OPULENT
68 PERVERTED
69 PREVUE
70 REASSESS
71 RECALCULATE
72 RIFLEMEN
73 SHAGGY
74 SHORTISH
75 SOMEBODY
76 STUPEFYING
77 SUEDE
78 THROWN
79 VERACITY
80 WINCED
81 ZIONISM
```

Gambar 28. Kata-kata yang dicari pada *puzzle* ukuran besar 3.

Gambar 29. *Output* solusi *puzzle* ukuran besar 3.



Gambar 30. *Output* banyak perbandingan dan waktu eksekusi program untuk *puzzle* ukuran besar 3.

Alamat *repository source code* program

Source code program yang telah dibuat dapat diakses melalui tautan ini.

Check list

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