

Laporan Tugas Kecil 1

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

IF2211 Strategi Algoritma

Fawwaz Anugrah Wiradhika Dharmasatya – 13520086 – K02

I. Algoritma

Implementasi penyelesaian *Word Search Puzzle* dengan algoritma *brute force* dilakukan dalam bahasa Java dan dijalankan di terminal. Program berupa sebuah file .jar. Untuk menjalankan program tersebut, disediakan sebuah file run.bat.

Program ini pertama-tama akan meminta input sebuah nama file beserta ekstensinya (misal: “input.txt”). File diasumsikan sudah diletakkan di folder ‘test’. Setelah file telah dibuka, file tersebut akan dibaca dan di-*parsing* untuk mendapatkan matriks soal *Word Search Puzzle* serta sebuah list berisi kata-kata yang perlu dicari.

Langkah selanjutnya adalah penyelesaian puzzle menggunakan algoritma *brute force*. Sebelum algoritma dijalankan, dideklarasikan terlebih dahulu list kosong untuk menyimpan kata yang sudah ditemukan serta sebuah matriks integer untuk penandaan pada proses pewarnaan hasil nanti.

Proses kerja algoritma ini adalah dengan melakukan pemeriksaan pada semua elemen pada matriks soal. Program akan berhenti ketika semua kata telah ditemukan atau elemen terakhir matriks sudah diperiksa. Pada tiap elemen matriks, akan dilakukan pencarian ke delapan arah untuk menemukan kata-kata pada list. Arah tersebut antara lain: atas, kanan-atas, kanan, kanan-bawah, bawah, kiri-bawah, kiri, dan kiri-atas. Pada setiap pemeriksaan arah, akan dilakukan pengecekan untuk mencari apakah ada kata pada list kata yang terdapat dalam arah tersebut.

Proses pencarian pada tiap arah tersebut dilakukan dengan melakukan pemeriksaan untuk setiap kata pada list kata, dimana pencarian tersebut dimulai dari elemen matriks yang sedang diperiksa sebagai indeks pertama, lalu bergerak sesuai arah yang diperiksa sampai ada elemen matriks yang berbeda dengan karakter pada kata yang sedang dibandingkan atau jika jumlah pengulangan sudah sebanyak panjang kata. Pada kasus terdapat karakter yang berbeda, maka pengulangan akan langsung dilanjutkan ke kata selanjutnya. Namun pada kasus jumlah pengulangan sudah sebanyak panjang kata, maka kata tersebut terdapat pada arah tersebut. Kata tersebut akan dihapus dari list kata dan dimasukkan ke list baru yang berisi kata yang sudah ditemukan. Lalu, matriks penanda posisi tersebut akan diperbaharui untuk menandakan kata yang sudah ditemukan tersebut. Setelah semua kata sudah diperiksa pada suatu arah, maka akan dilanjutkan ke

arah lain. Setelah semua arah sudah diperiksa, maka akan dilanjutkan ke elemen matriks berikutnya.

II. Source Code Program

A. Main.java

```
import java.util.ArrayList;
import java.util.Scanner;
import java.io.File;
import java.util.Date;

public class Main {
    //program utama
    public static void main(String[] args) {
        //Input nama file
        System.out.println("Masukkan nama file:");
        Scanner sc = new Scanner(System.in);
        String filename = sc.nextLine();
        File file = new File("../test/"+filename);
        //Membaca isi file
        Parser parser = new Parser(file);
        ParserResult res = parser.parse();
        //jumlah kata yang harus dicari
        int totalWords = res.getListKata().length;
        //Menyelesaikan persoalan
        final long startTime = System.currentTimeMillis();
        Solver hasil = new Solver(res);
        QueryResult finalResult = hasil.solve();
        final long endTime = System.currentTimeMillis();
        ArrayList<String> listKata = hasil.getFoundWord();
        //jumlah kata yang berhasil ditemukan
        int numOfFoundedWords = listKata.size();
        //Memproses hasil
        String hasilAkhir = Main.prosesMatrix(finalResult);
        //menambahkan list kata yang ada
        hasilAkhir += "\n";
        hasilAkhir += Main.listOfWords(listKata);
        //menampilkan jumlah waktu tempuh
        long elapsedTime = endTime-startTime;
        hasilAkhir+= "\n";
        hasilAkhir += "Total waktu untuk menyelesaikan puzzle: ";
        hasilAkhir += Long.toString(elapsedTime);
        hasilAkhir += "ms";
        //menampilkan jumlah kata
        hasilAkhir += "\n";
        hasilAkhir += "Jumlah total perbandingan huruf yang dilakukan:";
        hasilAkhir += finalResult.getStep();
        //menampilkan jumlah kata yang berhasil ditemukan
        hasilAkhir += "\nJumlah kata yang berhasil ditemukan: ";
        hasilAkhir += Integer.toString(numOfFoundedWords);
        hasilAkhir += "/";
        hasilAkhir += Integer.toString(totalWords);
        //Menampilkan hasil
        System.out.println(hasilAkhir);
        System.out.println("\n Tekan tombol apa saja lalu tekan enter untuk
```

```

keluar...");
    String end = sc.next();
}
//menambahkan list kata ke hasil akhir
private static String listOfWords(ArrayList<String> listKata){
    String stringkata = "";
    for(int i=0;i<listKata.size();i++){
        stringkata+= Color.getColor(i);
        stringkata+= listKata.get(i);
        stringkata += "\u001B[0m";
        stringkata += "\n";
    }
    return stringkata;
}
//cetak matrix
private static String prosesMatrix(QueryResult result) {
    String[][] matriksKata = result.getMatrix();
    int[][] matriksPenunjuk = result.getMappedMatrix();
    String hasilAkhir = "";
    int rows = matriksKata.length;
    int columns = matriksKata[0].length;
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < columns; j++) {
            hasilAkhir += Color.getColor(matriksPenunjuk[i][j]);
            hasilAkhir += matriksKata[i][j];
            hasilAkhir += "\u001B[0m";
            hasilAkhir += " ";
        }
        hasilAkhir += "\n";
    }
    return hasilAkhir;
}
}

```

B. Color.java

```

class Color{
    public static String getColor(int idx){
        String color="";
        if(idx== -1){//default
            color = "\u001b[0m";//reset
        }
        else {
            switch (idx % 6) {
                case 0:
                    color = "\u001b[31m";//red
                    break;
                case 1:
                    color = "\u001b[32m";//green
                    break;
                case 2:
                    color = "\u001b[33m";//yellow
                    break;
                case 3:
                    color = "\u001b[34m";//blue
                    break;
                case 4:

```

```

        color = "\u001b[35m";//magenta
        break;
    case 5:
        color = "\u001b[36m";//cyan
        break;
    }
}
return color;
}
}

```

C. Parser.java

```

import java.awt.desktop.SystemEventListener;
import java.io.*;
import java.io.File;
import java.util.Locale;

public class Parser {
    private File file;
    public Parser(File file){
        this.file = file;
    }
    public ParserResult parse(){
        int rows;
        int columns;
        String isiMatriks = "";
        String[][] matriksHasil = null;
        String[] listKata = null;
        try{
            BufferedReader reader = new BufferedReader(new
            FileReader(this.file));
            String line = reader.readLine();
            if(line!=null) {
                //isi matriks
                while (line != null && line.length() > 0) {
                    isiMatriks += line.toUpperCase();
                    isiMatriks += '\n';
                    line = reader.readLine();
                }
                //isi listkata
                String isiListKata = "";
                line = reader.readLine();
                while (line != null) {
                    isiListKata += line.toUpperCase();
                    isiListKata += '\n';
                    line = reader.readLine();
                }
                //proses matriks
                String[] matriksKataDipecahBaris = isiMatriks.split("\n");
                rows = matriksKataDipecahBaris.length;
                String[] tempArr = matriksKataDipecahBaris[0].split(" ");
                columns = tempArr.length;
                matriksHasil = new String[rows][columns];
                for(int i=0;i<rows;i++){
                    matriksHasil[i] = matriksKataDipecahBaris[i].split("
");
                }
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

```

```

        }
        //proses listKata
        listKata = isiListKata.split("\n");
        //menampilkan status
        System.out.println("File:"+this.file.getName()+" berhasil
dibuka.");
    }
}
catch(FileNotFoundException fnf ){
    System.out.println("File tidak ditemukan!");
}
catch(IOException io){
    System.out.println("File Kosong!");
}
return new ParserResult(matriksHasil,listKata);
}
}

```

D. ParserResult.java

```

public class ParserResult {
    private String[][] matriksSoal;
    private String[] listKata;
    //constructor
    public ParserResult(String[][] matriksSoal, String[] listKata){
        this.matriksSoal = matriksSoal;
        this.listKata = listKata;
    }
    public String[][] getMatriksSoal(){
        return this.matriksSoal;
    }
    public String[] getListKata(){
        return this.listKata;
    }
}

```

E. QueryResult.java

```

public class QueryResult {

    private String[][] matrix;
    private int[][] mappedMatrix;//matrix menentukan warna untuk sel yang akan
dicetak
    private int step;//langkah yang ditempuh untuk mencari jawaban

    public QueryResult(String[][] mat,int[][] matrix,int step){
        this.matrix = mat;
        this.mappedMatrix = matrix;
        this.step = step;
    }
    public String[][] getMatrix() {
        return matrix;
    }
    public int[][] getMappedMatrix() {
        return mappedMatrix;
    }
    public int getStep() {

```

```

        return step;
    }
}

```

F. Solver.java

```

import java.util.ArrayList;
import java.util.Collections;
import java.util.Iterator;

public class Solver {
    private ParserResult parserResult;
    private static int colorCode=0;//kode untuk menentukan warna
    private ArrayList<String> foundWords;
    public Solver(ParserResult parserResult){
        this.parserResult = parserResult;
    }

    public QueryResult solve(){
        //loop tiap elemen
        String[][] matrix = this.parserResult.getMatriksSoal();//matriks soal yang akan diperiksa
        String[] listKata = this.parserResult.getListKata();//kata-kata yang akan dicari
        int i_inner;//variabel untuk looping baris tiap pencarian kata
        int j_inner;//variabel untuk looping kolom tiap pencarian kata
        int rows = matrix.length;
        ArrayList<String> foundWords = new ArrayList<String>();//list berisi kata yang sudah ditemukan
        int step = 0;//jumlah perbandingan untuk menyelesaikan permainan
        int k;//variabel untuk looping tiap kata, di loop sebanyak jumlah huruf pada kata tersebut
        boolean isSame;//bernilai false jika ada karakter yang tidak sama saat pemeriksaan
        if(rows>0) {
            int columns = matrix[0].length;
            int[][] foundLocationMatrix = createIntMapMatrix(rows,columns);//matriks untuk menandakan warna sel yang akan dicetak
            //sediakan list kata untuk diperiksa
            ArrayList<String> listKataTiapItem = new ArrayList<String>();
            Collections.addAll(listKataTiapItem,listKata);
            Iterator<String> iter;
            for (int i = 0; i < rows; i++) {
                if(listKataTiapItem.size()<=0) {//udah ketemu semua
                    break;
                }
                for (int j = 0; j < columns;j++){
                    if(listKataTiapItem.size()<=0) {//udah ketemu semua
                        break;
                    }
                    //cek semua arah
                    //ke atas
                    if(i>0 && listKataTiapItem.size()>0){
                        iter = listKataTiapItem.iterator();
                        while(iter.hasNext()){
                            isSame = true;
                            String kata = iter.next();
                            i_inner = i;
                            j_inner = j;
                            k=0;
                            while(k<kata.length()&& i_inner>=0 && isSame) {
                                step++;
                                if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                                    isSame = false;
                                }
                                else {
                                    i_inner--;
                                    k++;
                                }
                            }
                            if(k==kata.length()){//kata ketemu
                                foundWords.add(kata);
                                iter.remove();
                                foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"N",i,j,k);
                            }
                        }
                    }
                    //ke kanan atas
                    if(i>0 && j<columns-1 && listKataTiapItem.size()>0){
                        iter = listKataTiapItem.iterator();
                        while(iter.hasNext()){
                            isSame = true;
                            i_inner = i;
                            j_inner = j;
                            String kata = iter.next();
                            k=0;
                            while(k<kata.length()&& i_inner>=0 && j_inner<=columns-1 && isSame) {
                                step++;
                                if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                                    isSame = false;
                                }
                                else {
                                    i_inner--;
                                    j_inner++;
                                    k++;
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```

        if(k==kata.length()){//kata ketemu
            foundWords.add(kata);
            iter.remove();
            foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"NE",i,j,k);
        }
    }
}
//ke kanan
if(j<columns-1 && listKataTiapItem.size()>0){
    iter = listKataTiapItem.iterator();
    while(iter.hasNext()){
        String kata = iter.next();
        i_inner = i;
        j_inner = j;
        k=0;
        isSame = true;
        while(k<kata.length()&& j_inner<=columns-1 && isSame) {
            step++;
            if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                isSame = false;
            }
            else {
                j_inner++;
                k++;
            }
        }
        if(k==kata.length()){//kata ketemu
            foundWords.add(kata);
            iter.remove();
            foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"E",i,j,k);
        }
    }
}
//ke kanan bawah
if(i<rows-1 && j<columns-1 && listKataTiapItem.size()>0){
    iter = listKataTiapItem.iterator();
    while(iter.hasNext()){
        i_inner = i;
        j_inner = j;
        String kata = iter.next();
        k=0;
        isSame = true;
        while(k<kata.length()&& i_inner<=rows-1 && j_inner<=columns-1 && isSame) {
            step++;
            if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                isSame = false;
            }
            else {
                i_inner++;
                j_inner++;
                k++;
            }
        }
        if(k==kata.length()){//kata ketemu
            foundWords.add(kata);
            iter.remove();
            foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"SE",i,j,k);
        }
    }
}
//ke bawah
if(i<rows-1&& listKataTiapItem.size()>0){
    iter = listKataTiapItem.iterator();
    while(iter.hasNext()){
        isSame = true;
        i_inner = i;
        j_inner = j;
        String kata = iter.next();
        k=0;
        while(k<kata.length()&& i_inner<=rows-1&&isSame) {
            step++;
            if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                isSame = false;
            }
            else {
                i_inner++;
                k++;
            }
        }
        if(k==kata.length()){//kata ketemu
            foundWords.add(kata);
            iter.remove();
            foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"S",i,j,k);
        }
    }
}
//ke kiri bawah
if(i<rows-1 && j>0&&listKataTiapItem.size()>0){
    iter = listKataTiapItem.iterator();
    while(iter.hasNext()){
        i_inner = i;
        j_inner = j;
        k=0;
        String kata = iter.next();
        isSame = true;

```

```

        while(k<kata.length() && i_inner<=rows-1 && j_inner>=0 && isSame) {
            step++;
            if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                isSame = false;
            }
            else {
                i_inner++;
                j_inner--;
                k++;
            }
        }
        if(k==kata.length()){//kata ketemu
            foundWords.add(kata);
            iter.remove();
            foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"SW",i,j,k);
        }
    }
}
//ke kiri
if(j>0 && listKataTiapItem.size()>0){
    iter = listKataTiapItem.iterator();
    while(iter.hasNext()){
        i_inner = i;
        j_inner = j;
        k=0;
        isSame = true;
        String kata = iter.next();
        while(k<kata.length() && j_inner>=0 && isSame) {
            step++;
            if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                isSame = false;
            }
            else {
                j_inner--;
                k++;
            }
        }
        if(k==kata.length()){//kata ketemu
            foundWords.add(kata);
            iter.remove();
            foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"W",i,j,k);
        }
    }
}
//ke kiri atas
if(i>0 && j>0 && listKataTiapItem.size()>0){
    iter = listKataTiapItem.iterator();
    while(iter.hasNext()){
        i_inner = i;
        j_inner = j;
        k=0;
        String kata = iter.next();
        isSame = true;
        while(k<kata.length() && i_inner>=0 && j_inner>=0 && isSame) {
            step++;
            if(kata.charAt(k)!=matrix[i_inner][j_inner].charAt(0)){
                isSame = false;
            }
            else {
                i_inner--;
                j_inner--;
                k++;
            }
        }
        if(k==kata.length()){//kata ketemu
            foundWords.add(kata);
            iter.remove();
            foundLocationMatrix = updateIntMapMatrix(foundLocationMatrix,"NW",i,j,k);
        }
    }
}
}
}

this.foundWords = foundWords;
return new QueryResult(matrix,foundLocationMatrix,step);
}
else {
    this.foundWords = foundWords;
    return null;
}
}

//dapatkan urutan kata yang sudah ditemukan
public ArrayList<String> getFoundWord(){
    return this.foundWords;
}

private int[][] updateIntMapMatrix(int[][] matrix, String direction, int _i, int _j, int length) {
    int j;
    int i;
    switch(direction){
        case "N"://ke atas
            for(i=_i;i>_i-length;i--){
                matrix[i][_j] = Solver.colorCode;
            }
            break;
    }
}

```



```

        case "NE"://ke kanan atas
            j = _j;
            for(i=_i;i> i-length;i--) {
                matrix[i][j] = Solver.colorCode;
                j++;
            }
            break;
        case "E"://ke kanan
            for (j = _j; j < _j + length; j++) {
                matrix[_i][j] = Solver.colorCode;
            }
            break;
        case "SE"://ke kanan bawah
            j = _j;
            for(i=_i;i<i+length;i++) {
                matrix[i][j] = Solver.colorCode;
                j++;
            }
            break;
        case "S"://ke bawah
            for(i=_i;i<i+length;i++){
                matrix[i][_j] = Solver.colorCode;
            }
            break;
        case "SW"://ke kiri bawah
            j = _j;
            for(i=_i;i<i+length;i++) {
                matrix[i][j] = Solver.colorCode;
                j--;
            }
            break;
        case "W"://ke kiri
            for (j = _j; j > _j - length; j--) {
                matrix[_i][j] = Solver.colorCode;
            }
            break;
        case "NW"://ke kiri atas
            j = _j;
            for(i=_i;i> i-length;i--) {
                matrix[i][j] = Solver.colorCode;
                j--;
            }
            break;
        default:
            System.out.println("Arah Invalid!");
    }
    Solver.colorCode++;
    return matrix;
}

public int[][] createIntMapMatrix(int rows,int columns){
    int[][] matriks = new int[rows][columns];
    for(int i=0;i<rows;i++){
        for(int j=0;j<columns;j++){
            matriks[i][j] = -1;//default valuenya -1
        }
    }
    return matriks;
}
}

```

III. Hasil Eksekusi

a) Ukuran Kecil (Antara 14x12 hingga 18x16)

1. File “small-1.txt”

- Jumlah baris:15
- Jumlah kolom:18

Input	Output
<pre> G Y G R H X F K O R E A R C D P M S U R D D K G Q B L F V G K B N S A R S O E D N Y B Y L R R L P B A F M D V S L J O A C D P A W A V X L J L C U E H Y M C M X I J H Y N O G O U T T V I J W D K O H R C G X C N B K B I R H R N Q B H T T U K U W E H M F W Y J A P F V H O T T M M M K S W U M N L E Z T X K I X O G I F O Z N W L O L C O Q U J K F Y I F T A R F V P B I J Q G U O T X T N B X R J R Z T O T S A G Z B W T R M A T A R A M B T H W C A M I N G F Q U D P H J T L P A O A E J D R O V X E Q S E G S O H D C R U S S I A A I S S U R p Y PRUSSiA OTTOMAN FRANCE ENGLAND RUSSIA POLAND TIMURID MING DELHI MUGHAL MATARAM TOKUGAWA KOREA MAMLUK </pre>	<pre> Masukkan nama file: small-1.txt File:small-1.txt berhasil dibuka. G Y G R H X F K O R E A R C D P M S U R D D K G Q B L F V G K B N S A R S O E D N Y B Y L R R L P B A F M D V S L J O A C D P A W A V X L J L C U E H Y M C M X I J H Y N O G O U T T V I J W D K O H R C G X C N B K B I R H R N Q B H T T U K U W E H M F W Y J A P F V H O T T M M M K S W U M N L E Z T X K I X O G I F O Z N W L O L C O Q U J K F Y I F T A R F V P B I J Q G U O T X T N B X R J R Z T O T S A G Z B W T R M A T A R A M B T H W C A M I N G F Q U D P H J T L P A O A E J D R O V X E Q S E G S O H D C R U S S I A A I S S U R P Y KOREA MAMLUK DELHI FRANCE TOKUGAWA ENGLAND MUGHAL OTTOMAN TIMURID POLAND MATARAM MING RUSSIA PRUSSIA Total waktu untuk menyelesaikan puzzle: 6ms Jumlah total perbandingan huruf yang dilakukan:14662 Jumlah kata yang berhasil ditemukan: 14/14 Tekan tombol apa saja lalu tekan enter untuk keluar... </pre>

2. File “small-2.txt”

- Jumlah baris: 14
- Jumlah kolom:16

Input	Output
<pre> T I U L F F A M A N T L E M D L Q F J M L W P E N O C F T B P N I T V S L F Y A G H G K K Q V Y F O W O T L Q Q R C A L L E R E C I N O I X B U K S S I Z F Q E Q K B X H W V M A R O N V C P E D C P J X A U Y E V O N J R D I Y E M E U N W V B R E D A O B P W P Z T N T E I B W L R C K Y K M S O C B E C S N V W E W Z G Q I I C H U D E K I G R V U F N V E R J E E A W N H F E A R T E V W X P D Z I X X W U Z D A B W Y S K H U E B H O A R D T F G J M BRONZE CALLER CONE ETCHED EVER HOARD ITLL MANTLE PARSON QUAVER RECODE SPECK WANTED WINGED </pre>	<pre> Masukkan nama file: small-2.txt File:small-2.txt berhasil dibuka. T I U L F F A M A N T L E M D L Q F J M L W P E N O C F T B P N I T V S L F Y A G H G K K Q V Y F O W O T L Q Q R C A L L E R E C I N O I X B U K S S I Z F Q E Q K B X H W V M A R O N V C P E D C P J X A U Y E V O N J R D I Y E M E U N W V B R E D A O B P W P Z T N T E I B W L R C K Y K M S O C B E C S N V W E W Z G Q I I C H U D E K I G R V U F N V E R J E E A W N H F E A R T E V W X P D Z I X X W U Z D A B W Y S K H U E B H O A R D T F G J M MANTLE PARSON CONE QUAVER CALLER ITLL WANTED ETCHED WINGED EVER BRONZE SPECK RECODE HOARD Total waktu untuk menyelesaikan puzzle: 4ms Jumlah total perbandingan huruf yang dilakukan:10162 Jumlah kata yang berhasil ditemukan: 14/14 Tekan tombol apa saja lalu tekan enter untuk keluar... </pre>

3. File "small-3.txt"

- Jumlah baris: 14
- Jumlah kolom:16

Input	Output
<pre> T M R A U K U H A E E S E P Q L I P D P Z A Z Z C V T T G C O Q Z X O W C K A O N E A R M F U L Y L R A E Y N W E K P Z R V D R A M Q H F F P P E G O T I E M M F A T F E Q S W S N L E T L U C S G Q R M T K W I D O G Z O N H M N Q H Y E E M M L Y D I O J V A U I M R U O S E L B E E F V S C M P C C D B P B K U H J F D N E B R E E Z E A I X G O V E R T E S R X M Y Y T A F M Z J K A U L U J V H E L E L E A P E F J J X Y K M O G F R A S C A L T F B </pre> <p> ARMFUL BREEZE CONFER CULT DOMINO FEEBLE LEAP MACE MAGNUM OVERT RASCAL STEEP TAPS YEARLY </p>	<pre> Masukkan nama file: small-3.txt File:small-3.txt berhasil dibuka. T M R A U K U H A E E S E P Q L I P D P Z A Z Z C V T T G C O Q Z X O W C K A O N E A R M F U L Y L R A E Y N W E K P Z R V D R A M Q H F F P P E G O T I E M M F A T F E Q S W S N L E T L U C S G Q R M T K W I D O G Z O N H M N Q H Y E E M M L Y D I O J V A U I M R U O S E L B E E F V S C M P C C D B P B K U H J F D N E B R E E Z E A I X G O V E R T E S R X M Y Y T A F M Z J K A U L U J V H E L E L E A P E F J J X Y K M O G F R A S C A L T F B </pre> <p> STEEP CONFER ARMFUL YEARLY MAGNUM CULT MACE FEEBLE DOMINO BREEZE OVERT TAPS LEAP RASCAL </p> <p> Total waktu untuk menyelesaikan puzzle: 10ms Jumlah total perbandingan huruf yang dilakukan:12363 Jumlah kata yang berhasil ditemukan: 14/14 Tekan tombol apa saja lalu tekan enter untuk keluar... </p>

b) Ukuran Sedang (Antara 20x18 hingga 24x22)

1. File “medium-1.txt”

- Jumlah baris:20
- Jumlah kolom:22

• Input

P	H	E	C	W	I	E	I	C	B	Q	V	O	L	K	H	M	N	D	I	T		ABASH
A	S	S	E	R	T	I	N	G	I	V	Y	W	S	H	Y	D	B	U	N	P	Y	ASSERTING
G	B	W	R	I	C	N	F	H	S	X	O	R	C	M	J	M	U	N	J	R	N	ASTHMA
P	L	K	A	S	C	Q	S	S	E	L	E	C	R	U	O	S	W	M	T	M	J	CELSIUS
U	O	A	J	Y	R	Q	E	E	J	G	N	I	S	I	P	S	E	D	D	I	Y	DECOMPRESS
P	A	I	N	A	B	R	Q	D	W	W	P	S	W	A	D	S	Q	F	B	E	Q	DESPISING
T	I	J	C	C	P	R	G	U	U	N	P	U	X	D	Q	K	G	D	W	B	L	DRAUGHT
O	H	S	O	M	E	F	U	I	C	V	D	H	N	E	W	S	F	L	A	S	H	GLANCE
U	O	X	O	U	T	P	M	J	W	F	W	E	G	O	Z	B	C	N	L	C	G	LAUGH
X	G	C	D	D	Y	T	S	C	B	B	I	X	K	L	Y	V	Y	N	V	S	K	LEES
G	E	A	N	Y	K	R	G	T	P	G	R	Y	L	X	A	Y	P	T	U	O	D	LEFTY
D	F	J	A	A	M	O	K	H	C	S	R	W	B	V	A	R	R	U	F	N	R	MIGHT
D	R	S	U	M	I	P	A	G	W	Q	U	S	C	O	Y	B	B	H	D	E	Z	NEWSFLASH
Q	H	X	J	H	G	S	S	U	F	E	W	R	G	N	X	S	A	C	X	L	L	OSCAR
C	U	E	Z	T	H	S	U	A	H	W	F	A	U	L	A	L	T	S	C	K	I	PASSPORT
Z	B	G	E	S	T	A	G	R	G	D	Q	X	R	A	Y	I	F	U	H	Q	R	SLIME
X	L	X	R	A	D	P	Q	D	U	O	K	Y	O	P	S	M	P	N	G	K	J	SMUG
A	U	F	B	K	P	B	F	E	A	Q	L	P	S	L	E	E	R	K	V	P	F	SOURCELESS
Y	W	K	Z	E	X	O	L	Q	L	S	B	H	G	K	X	D	H	S	S	X	K	THESAURUS
K	X	I	K	O	S	U	I	S	L	E	C	J	T	L	E	E	S	T	T	T	T	WARPED

- Output

```
Masukkan nama file:
medium-1.txt
File:medium-1.txt berhasil dibuka.
O H E C W W I 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 I Y
P A I N A B R Q D W W P S W A D S Q F B E Q
T 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 E 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

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

Total waktu untuk menyelesaikan puzzle: 11ms
Jumlah total perbandingan huruf yang dilakukan:41037
Jumlah kata yang berhasil ditemukan: 20/20

Tekan tombol apa saja lalu tekan enter untuk keluar...
```

2. File “medium-2.txt”

- Jumlah baris:18
- Jumlah kolom:24

- Input

L	W	H	M	I	E	R	E	N	Q	M	F	X	X	V	J	V	I	S	S	U	J	K	J	EREN
Z	L	K	Z	Y	O	Z	M	A	H	S	A	S	J	R	E	C	E	G	W	Q	S	D	R	MIKASA
E	E	P	K	C	E	I	P	J	P	U	I	V	S	V	A	I	H	C	B	A	D	Y	T	ARMIN
J	D	C	O	O	E	S	F	J	E	R	W	I	N	F	N	F	A	D	E	E	A	L	X	JEAN
J	P	G	K	U	I	G	H	O	W	N	M	Q	V	R	I	R	N	F	R	M	N	T	X	CONNIE
U	L	O	Y	S	N	F	U	N	V	D	F	L	L	F	V	J	X	T	O	N	O	T	SASHA	
S	P	H	S	B	N	E	L	K	O	V	M	B	H	Q	T	V	I	E	H	Z	I	P	I	REINER
D	X	O	Z	C	O	U	N	O	O	T	N	Q	N	V	E	S	E	J	O	M	E	L	P	BERTHOLD
E	A	J	R	P	C	G	W	Z	T	D	X	V	V	W	W	W	O	C	L	E	C	Y	E	ANNIE
W	R	I	A	C	Y	J	H	M	H	C	U	L	O	M	O	M	C	E	D	M	J	H	Y	FLOTCH
C	M	A	X	L	O	U	F	T	E	B	H	L	S	L	X	S	U	W	I	S	I	C	A	HITCH
G	I	R	S	U	B	E	W	V	S	V	R	I	R	O	E	X	N	K	V	T	W	I	Q	MARLOWE
I	N	R	E	I	N	E	R	L	G	A	Q	A	F	E	H	X	A	C	C	E	R	P	L	ERWIN
F	M	T	D	B	T	O	D	O	E	K	M	E	J	V	L	S	Z	H	L	O	H	C	V	HANJI
A	P	Z	H	B	S	K	L	F	V	C	K	A	X	B	A	S	K	I	T	O	A	L	H	LEVI
L	L	E	V	I	G	Z	D	N	I	B	R	L	E	C	R	A	N	S	B	Y	M	I	R	PIECK
C	S	H	G	D	E	K	Z	W	J	R	J	A	P	E	S	D	I	J	D	A	I	F	E	PORCO
O	W	J	H	S	G	D	F	C	X	Z	V	N	M	P	O	H	T	Y	U	I	G	R	E	MARCEL
																								YMIR
																								HISTORIA
																								FALCO
																								GABI

- Output

```
Masukkan nama file:
medium-2.txt
File:medium-2.txt berhasil dibuka.
L W H M I E R E N Q M F X X V J V I S S U J K J
Z L K Z Y O Z M A H S A S J R E C E G W Q S D R
E E P K C E I P J P U I V S V A I H C B A D Y T
J D C O O E S F J E R W I N F N F A D E E A L X
J P G K U I G H O W N M Q V R I R N F R M N T X
U L O Y S N F U N V D F L L L F V J X T O N O T
S P H S B N E L K O V M B H Q T V I E H Z I P I
D X O Z C O U N O O T N Q N V E S E J O M E L P
E A J R P C G W Z T D X V V W W W O C L E C Y E
W R I A C Y J H M H C U L O M O M C E D M J H Y
C M A X L O U F T E B H L S L X S U W I S I C A
G I R S U B E W V S V R I R O E X N K V T W I Q
I N R E I N E R L G A Q A F E H X A C C E R P L
F M T D B T O D O E K M E J V L S Z H L O H C V
A P Z H B S K L F V C K A X B A S K I T O A L H
L L E V I G Z D N I B R L E C R A N S B Y M I R
C S H G D E K Z W J R J A P E S D I J D A I F E
O W J H S G D F C X Z V N M P O H T Y U I G R E
```

```

EREN
JEAN
SASHA
PIECK
HANJI
BERTHOLD
ERWIN
ANNIE
FLOTCH
PORCO
ARMIN
CONNIE
MIKASA
HITCH
REINER
FALCO
MARLOWE
LEVI
YMIR
MARCEL
HISTORIA
GABI

Total waktu untuk menyelesaikan puzzle: 8ms
Jumlah total perbandingan huruf yang dilakukan:34747
Jumlah kata yang berhasil ditemukan: 22/22

Tekan tombol apa saja lalu tekan enter untuk keluar...

```

3. File “medium-3.txt”

- Jumlah baris:22
- Jumlah kolom:20

• Input

T M D H D O B B K E I V F Q V F Z K U J	
T W A A J A I L Y M U Z S U K A U A T Q	AKU
O E L S N F N Y B X K L X W T K M M I H	SUKA
O L P E I A G A M Z A P G G A Z I U G R	SAMA
W Z R A U H U M F U G K I P O T F L E Y	KAMU
S A X Y T D N V U K N Q D M S W G I B F	BELUM
K W I Q Y Y G Y Z N E P G M S K I F S O	BERANI
I O H W T D K Y X O M N Z C M A A R O Y	MENGAKUI
I N K P P L E N S M G Y A O O D T H J R	KARENA
S E A F I J E E Z H B H G M K L T A H P	MERASA
N K A R O O C K R Q Y C E P D V Y T J C	MASIH
J X H P E A G A R H B N L L U P K B N J	BINGUNG
G C S F Z B M Z M V C S D O A W R B K I	MENCARI
T F G K M F U G S A U F V V P T B E X L	TOPIK
M E R A S A T T R A E N C L U U I W V D	SUDAH
M U L E B E F I G D M P D L L S S B K N	BEGITU
W E V E T Y D V N B S A H R U N K J B A	SAJA
A J P A V V K U T Q G C A A H V K P H M	TEPAT
F Z P Z D R Y Q Z N S J V M D K M E D U	DUAPULUH
W I E Q G X M R U W A S V W D U R X J N	
H L W D P D Y V F S S V M X K F S V T E	
X U Z N S L D B B N Z J M M D U L Z T T	

- Output

```
Masukkan nama file:
medium-3.txt
File:medium-3.txt berhasil dibuka.
T M D H D O B B K E I V F Q V F Z K U J
T W A A J A I L Y M U Z S U K A U A T Q
O E L S N F N Y B X K L X W T K M M I H
O L P E I A G A M Z A P G G A Z I U G R
W Z R A U H U M F U G K I P O T F L E Y
S A X Y T D N V U K N Q D M S W G I B F
K W I Q Y Y G Y Z N E P G M S K I F S O
I O H W T D K Y X O M N Z C M A A R O Y
I N K P P L E N S M G Y A O O D T H J R
S E A F I J E E Z H B H G M K L T A H P
N K A R O O C K R Q Y C P D V Y T J C
J X H P E A G A R H B N L L U P K B N J
G C S F Z B M Z M V C S D O A W R B K I
T F G K M F U G S A U F V V P T B E X L
M E R A S A T T R A E N C L U U I W V D
M U L E B E F I G D M P D L L S S B K N
W E V E T Y D V N B S A H R U N K J B A
A J P A V V K U T Q G C A A H V K P H M
F Z P Z D R Y Q Z N S J V M D K M E D U
W I E Q G X M R U W A S V W D U R X J N
H L W D P D Y V F S S V M X K F S V T E
X U Z N S L D B B N Z J M M D U L Z T T
```

```
MASIH
BINGUNG
KAMU
TEPAT
SUKA
AKU
TOPIK
BEGITU
KARENA
MENGAKUI
MENCARI
DUAPULUH
BERANI
SAMA
MERASA
BELUM
SAJA
SUDAH

Total waktu untuk menyelesaikan puzzle: 6ms
Jumlah total perbandingan huruf yang dilakukan:22734
Jumlah kata yang berhasil ditemukan: 18/18

Tekan tombol apa saja lalu tekan enter untuk keluar...
```

c) Ukuran Besar(Antara 32x30 hingga 36x34)

1. File “large-1.txt”

- Jumlah baris:32
- Jumlah kolom:31

- Input

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

THE
 WORLD
 IS
 CRUEL
 BUT
 ILL
 STILL
 LOVE
 YOU
 NO
 MATTER
 WHAT
 HAVE
 TO
 SACRIFICE
 CONTINUE
 PROTECT
 INDUSTRIALIZATION
 CAPITALIST
 LIBERAL
 COMMUNIST
 CONSERVATIVE
 PROGRESSIVE
 ANARCHY
 AUTHORITARIAN
 EGALITARIAN
 RACISM
 EQUALITY
 UNITY
 DIVERSITY

- Output

Masukkan nama file:

large-1.txt

File:large-1.txt berhasil dibuka.

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

BUT
IS
EQUALITY
DIVERSITY
EGALITARIAN
ILL
PROGRESSIVE
NO
TO
THE
SACRIFICE
CONSERVATIVE
STILL
LIBERAL
MATTER
ANARCHY
AUTHORITARIAN
CAPITALIST
WHAT
UNITY
CONTINUE
COMMUNIST
WORLD
INDUSTRIALIZATION
HAVE
LOVE
YOU
PROTECT
RACISM
CRUEL

Total waktu untuk menyelesaikan puzzle: 41ms
Jumlah total perbandingan huruf yang dilakukan: 111395
Jumlah kata yang berhasil ditemukan: 30/30
Tekan tombol apa saja lalu tekan enter untuk keluar...

2. File “large-2.txt”

- Jumlah baris:30
- Jumlah kolom:36

- Input

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

ABATEMENT	
ADOPTED	
AGAINST	
AMBIDEXTROUS	
BENEFACTOR	
BIFOCAL	
BUREAUCRAT	
CANNERY	
CHEF	
CITYSCAPE	
CLERGY	
CREAKING	
DEGREE	
DESPAIRE	
DOORKNOB	
DRAMATIZING	
DRAT	
EXACERBATE	
EXCEEDING	
EXERTING	
FEATURE	PROCTORS
FEEBLY	PUNISHING
FONDLING	REFINER
GRAVITATE	REPROVE
GRAVY	RESEMBLE
HIND	REVERE
INDUCED	SCAPEGOAT
JEEP	SINCERITY
LIKEN	SPARTAN
MEMENTO	SURPASS
MISTY	SYNCHRONIZE
OBSESSIVE	TURPENTINE
PARADIGM	WAKING
PILGRIM	WINDSWEPT

- Output

```
Masukkan nama file:
large-2.txt
File:large-2.txt berhasil dibuka.
K Q G B T C M M C G U E W G V O U L K I R E B U I J R Y J T A R D H Y I
U B A H A B I M B E N E F A C T O R I O W W F T O Y G L A G N T U D Z E
K K Q Q R V I T U J J K Y B C T F P D O N A E A H Y R E P R D C D T E X
R Q B S C F Y Q Y W T D N Z T L K D G O P J A V I D A U D X W D R H Y E
K V Q X U Y Q R F S N F A Y S M W V O N W J T K K O V A E D F H A A L R
P F C P A C G E E G C P E F N D C P F J I L U M F T I N S D I I M I N T
C E P I E B U R W N Y A U W I F R I G E N D R Y S C T F P L U Y A O E I
I H B R R B N X E F N O P R A M E I W Q E F E S Y R A N A X E I T Q P N
X C C U U B E Y M L X A G E G H A K X L E B B E H L T D I Z G U I T J G
T I D Z B V U X H Q C G C Q A J K J T J R N L F C Q E A R O H A Z D V Z
Q H X M A M B I D E X T R O U S I O C C G P Z Y R X P I E Q F N I O M D
R L B I F O C A L L K W C Q A A N Z T B E I S N V I E G D A F D N A D O
D D A D H L P X D E T P O D A K G D V E D O O R K N O B M R T J G V F P
B N H A V A M S H F Q P P Z U C L V H V H C L E C E T A B R E C A X E W
D F T F V D Q F T N E M E T A B A Z I A O S G N I L D N O F X U F G Q B
T Z L B D C E A M Z W Y R V R Z L D T A O G E P A C S V I J H K D W Z X
H D M J S Z A L U I O M X Q U O R R C R T E T G D W S U R P A S S Y A R
A K R W B X I S N N P M J Z V X I I M D U Q E V O R P E R L X Y V V O E
T V H B W K K D I E U Q P E E J P Y K S R G D H H V A M P A O B I N E S
D F I J E D U A D S N Q G Y T A R N C X P M W W A K I N G Y T Y E D A E
V E T N J C E Y R M I C P R V X Q W S M E J B H H D L R A N B T R N C M
I E U H E M S O M W S R A O J A P U I D N C M C S V E I H F R E E F K B
H Y P D K D T N P O H U R R E M R S N B T Z R W I N D S W E P T V F O L
I Z X R J C O X M Y I Y A L M G O G C N I A L I I C R W P A V E E S A E
N O C B O Z D I G T N D D S I W T J E Y N N Z F A M C M S S R B R P B G
D R C R W W R G O W G U I L S A N T R P E K E I B E U R H F J U D A Z G
Z F P Y Y G P Q T F M E G D T T E K I D Y R J Q Z B M H U T P M K R D N
B P S X L L I D Z E O K M K Y F M D T L S E Z I N O R H C N Y S V T K K
T P O I E L I T X U Y C J X U J E K Y R H L J D V N W M L D U G J A N M
A A P Z A E V I S S E S B O U X M H R L I E E Q H I Z F P W L X F N O T
```


CITYSCAPE
DRAT
BENEFACTOR
FEATURE
GRAVITATE
EXERTING
DRAMATIZING
DESPAIED
CREAKING
FEEBLY
CHEF
BUREAUCRAT
CLERGY
CANNERY
AGAINST
AMBIDEXTROUS
BIFOCAL
EXCEEDING
ADOPTED
DEGREE
DOORKNOB
EXACERBATE
ABATEMENT
FONDLING
SCAPEGOAT
LIKEN
INDUCED
TURPENTINE
SURPASS
RESEMBLE
PUNISHING
REPROVE
JEEP
WAKING
PARADIGM
SINCERITY
HIND
WINDSWEPT
MISTY

GRAVY
SPARTAN
REVERE
PROCTORS
REFINER
SYNCHRONIZE
PILGRIM
OBSESSIVE
MEMENTO

Total waktu untuk menyelesaikan puzzle: 26ms

Jumlah total perbandingan huruf yang dilakukan: 202953

Jumlah kata yang berhasil ditemukan: 48/48

Tekan tombol apa saja lalu tekan enter untuk keluar...

3. File “large-3.txt”

- Jumlah baris:30
- Jumlah kolom:36

- Input

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

ABEYANCE	
ADDRESS	
ANISEED	
ASHAMED	
ASSIMILATE	
BAPTIZING	
BASIN	
BIDED	
CHART	
CONDIMENT	
CONSULT	
CONTAINMENT	
CRIME	
DETERMINIST	
DOWNSTAGE	
EMPIRICIST	
EXPERT	
FEED	
FRIGID	
GERMANY	SHEPHERD
HEARING	SKYSCRAPER
HOOT	SMART
IDIOM	SMOULDERED
INTEGRATION	SNAKED
LINEAR	SNEER
MANLINESS	SUNS
MARTINGALE	SWINE
NURTURING	TICKER
PASTURELAND	TIGHTENING
PHOTOGRAPHY	TRAFFIC
PRACTICE	USURP
PRINTABLE	WHEREON
REBATE	YULE
REPRISE	

- Output

Masukkan nama file:

large-3.txt

File:large-3.txt berhasil dibuka.

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

HOOT
IDIOM
ADDRESS
CHART
ABEYANCE
EXPERT
CRIME
INTEGRATION
BIDED
DETERMINIST
FEED
ASSIMILATE
BAPTIZING
BASIN
ASHAMED
ANISEED
CONSULT
EMPIRICIST
GERMANY
CONTAINMENT
DOWNSTAGE
CONDIMENT
HEARING
FRIGID
SKYSCRAPER
USURP
YULE
MARTINGALE
NURTURING
PHOTOGRAPHY
TIGHTENING
SMART
SWINE
TICKER
REBATE
PRINTABLE
WHEREON
MANLINESS
SNAKED

PRACTICE
TRAFFIC
LINEAR
SNEER
SHEPHERD
REPRISE
SMOULDERED
PASTURELAND
SUNS

Total waktu untuk menyelesaikan puzzle: 31ms
Jumlah total perbandingan huruf yang dilakukan: 202447
Jumlah kata yang berhasil ditemukan: 48/48

Tekan tombol apa saja lalu tekan enter untuk keluar...

IV. Alamat *Repository* Beserta Form Penilaian

Alamat *repository* program: <https://github.com/Wiradhika6051/Tucil-1-Stima>

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

Tabel 4.1. Poin Kelengkapan Spek Tugas Kecil 1