

# **IF2211 – Strategi Algoritma**

## **Tugas Kecil 1**

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



Oleh:

**13520118 Mohamad Daffa Argakoesoemah**

**PROGRAM STUDI SARJANA INFORMATIKA  
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG  
2022**

## I. Algoritma Brute Force dalam Penyelesaian Word Search Puzzle

Algoritma *Brute Force* merupakan algoritma dengan pendekatan yang lempang atau *straightforward* untuk memecahkan suatu persoalan. Salah satu pemanfaatan algoritma ini adalah untuk menyelesaikan Word Search Puzzle. Dalam program ini, huruf-huruf pada Word Search Puzzle disimpan di dalam matriks dua dimensi, sedangkan kata-kata yang merupakan jawabannya disimpan di dalam sebuah larik. Kata-kata dalam Word Search Puzzle bisa ditemukan dalam delapan arah, yaitu vertikal ke atas, vertikal ke bawah, horizontal ke kanan, horizontal ke kiri, diagonal ke kanan atas, diagonal ke kanan bawah, diagonal ke kiri atas, dan diagonal ke kiri bawah. Oleh karena itu, terdapat 8 metode dalam kelas `WordSearchPuzzle` yang masing-masing memeriksa kemungkinan delapan arah tersebut. Langkah-langkah penyelesaiannya adalah sebagai berikut:

1. Untuk setiap kata, mula-mula disejajarkan dengan huruf paling ujung pada matriks yang bergantung pada arah pencarian. Rinciannya adalah sebagai berikut:
  - a. Horizontal ke kanan: dilakukan pada setiap baris mulai dari baris pertama dan huruf paling ujung kiri
  - b. Horizontal ke kiri: dilakukan pada setiap baris mulai dari baris pertama dan huruf paling ujung kanan
  - c. Vertikal ke atas: dilakukan pada setiap kolom mulai dari kolom terakhir dan huruf paling ujung kiri bawah
  - d. Vertikal ke bawah: dilakukan pada setiap kolom mulai dari kolom pertama dan huruf paling ujung kiri atas
  - e. Diagonal ke kanan bawah: dilakukan pada setiap baris mulai dari baris pertama dan huruf paling ujung kiri. Khusus baris pertama akan dilakukan untuk tiap kolomnya.
  - f. Diagonal ke kiri bawah: dilakukan pada setiap baris mulai dari baris pertama dan huruf paling ujung kanan. Khusus baris pertama akan dilakukan untuk tiap kolomnya.
  - g. Diagonal ke kanan atas: dilakukan pada setiap baris mulai dari baris terakhir dan huruf paling ujung kiri. Khusus baris terakhir akan dilakukan untuk tiap kolomnya.
  - h. Diagonal ke kiri atas: dilakukan pada setiap baris mulai dari baris terakhir dan huruf paling ujung kanan. Khusus baris terakhir akan dilakukan untuk tiap kolomnya.
2. Lalu, setiap elemen matriks ditelusuri kesesuaian huruf dengan kata pada kunci jawaban. Hal ini dilakukan dengan membandingkan setiap huruf pada matriks dengan huruf pada kata pada kunci jawaban. Rinciannya sebagai berikut:
  - a. Horizontal ke kanan: ditelusuri ke kolom berikutnya (ke kanan)
  - b. Horizontal ke kiri: ditelusuri ke kolom sebelumnya (ke kiri)
  - c. Vertikal ke atas: ditelusuri ke baris sebelumnya (ke atas)
  - d. Vertikal ke bawah: ditelusuri ke baris setelahnya (ke bawah)
  - e. Diagonal ke kanan bawah: penelusuran dengan melakukan *increment* pada baris dan kolom
  - f. Diagonal ke kiri bawah: penelusuran dengan melakukan *increment* pada baris dan *decrement* pada kolom
  - g. Diagonal ke kanan atas: penelusuran dengan melakukan *increment* pada kolom dan *decrement* pada baris
  - h. Diagonal ke kiri atas: penelusuran dengan melakukan *decrement* pada baris dan kolom

Hal ini dilakukan sampai semua yang huruf dibandingkan cocok dengan huruf pada kata dalam kunci jawaban (kata berhasil ditemukan) atau huruf tidak cocok dengan huruf pada kata dalam kunci jawaban (tidak ada kecocokan).

3. Jika tidak ada kecocokan dan pencarian belum mencapai batas, dilakukan pergeseran ke huruf berikutnya dengan ketentuan seperti rincian pada langkah dua lalu kembali dilakukan langkah dua.

## II. Source Code Program

### A. File WordSearchPuzzle.java

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
import java.util.Scanner;

public class WordSearchPuzzle {
    ArrayList<ArrayList<String>> alphabet;
    ArrayList<String> word;

    // Konstruktor, membaca word search puzzle dari file teks
    WordSearchPuzzle(String fileName) throws FileNotFoundException {
        this.alphabet = new ArrayList<ArrayList<String>>();
        this.word = new ArrayList<String>();
        File file = new File(fileName);
        Scanner scan = new Scanner(file);

        while (scan.hasNextLine()) {
            String next = scan.nextLine();
            if (next.isEmpty()) { // membaca kunci jawaban (setelah line kosong)
                while (scan.hasNextLine()) {
                    this.word.add(scan.next());
                }
            } else { // membaca puzzle
                String[] a = next.split("\\s+");
                List<String> tmp2 = (List<String>) Arrays.asList(a);
                ArrayList<String> tmp = new ArrayList<String>(tmp2);
                this.alphabet.add(tmp);
            }
        }
    }
}
```

```
// Pencarian horizontal dari kiri

public void fromLeftHorizontal() {
    for (int j = 0; j < this.word.size(); j++) { // loop untuk tiap kata dalam kunci jawaban
        int lettersCompared = 0;
        int wordLength = this.word.get(j).length();
        for (int i = 0; i < this.alphabet.size(); i++) { // loop untuk tiap baris matriks
            boolean found = false;
            int textLength = this.alphabet.get(i).size();
            int k = 0;
            while (k <= (textLength - wordLength) && !found) { // loop dalam pemeriksaan karakter
                int l = 0;
                while ((l < wordLength) && (this.alphabet.get(i).get(k + l).equals(String.valueOf(this.word.get(j).charAt(l))))) { // loop dalam pemeriksaan karakter
                    lettersCompared++;
                    l++;
                }
                if (l == wordLength) {
                    int printed = k + wordLength - 1;
                    for (int row = 0; row < this.alphabet.size(); row++) {
                        for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                            if (row == i && col >= k && col <= printed) {
                                System.out.print(this.alphabet.get(row).get(col) + " ");
                            } else {
                                System.out.print("- ");
                            }
                        }
                        System.out.print("\n");
                    }
                    System.out.println("Jumlah komparasi huruf: " + (lettersCompared));
                    System.out.println("\n");
                    found = true;
                } else {
                    lettersCompared++;
                    k++;
                }
            }
        }
    }
}
}
```

```
// Pencarian horizontal dari kanan

public void fromRightHorizontal() {
    for (int j = 0; j < this.word.size(); j++) { // loop untuk tiap kata dalam kunci jawaban
        int lettersCompared = 0;
        for (int i = 0; i < this.alphabet.size(); i++) { // loop untuk tiap baris matriks
            boolean found = false;
            int textLength = this.alphabet.get(i).size();
            int wordLength = this.word.get(j).length();
            int k = textLength - 1;
            while (k >= (wordLength - 1) && !found) { // loop dalam pemeriksaan karakter
                int l = 0;
                while ((l < wordLength) && (this.alphabet.get(i).get(k-l).equals(String.valueOf(this.word.get(j).charAt(l))))) { // loop dalam pemeriksaan karakter
                    lettersCompared++;
                    l++;
                }
                if (l == wordLength) {
                    int printed = k-l+1;
                    for (int row = 0; row < this.alphabet.size(); row++) {
                        for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                            if (row == i && col >= printed && col <= printed + wordLength - 1) {
                                System.out.print(this.alphabet.get(row).get(col) + " ");
                            } else {
                                System.out.print("- ");
                            }
                        }
                        System.out.print("\n");
                    }
                    System.out.println("Jumlah komparasi huruf: " + (lettersCompared));
                    System.out.println("\n");
                    found = true;
                } else {
                    lettersCompared++;
                    k--;
                }
            }
        }
    }
}
}
```

```
// Pencarian vertical dari atas

public void fromTopVertical() {
    for (int j = 0; j < this.word.size(); j++) { // loop untuk tiap kata dalam kunci jawaban
        int lettersCompared = 0;
        int wordLength = this.word.get(j).length();
        for (int i = 0; i < this.alphabet.get(0).size(); i++) { // loop untuk tiap kolom matriks
            boolean found = false;
            int textLength = this.alphabet.size();
            int k = 0;
            while (k <= (textLength - wordLength) && !found) { // loop dalam pemeriksaan karakter
                int l = 0;
                while ((l < wordLength) && (this.alphabet.get(k + l).get(i).equals(String.valueOf(this.word.get(j).charAt(l))))) { // loop dalam pemeriksaan karakter
                    lettersCompared++;
                    l++;
                }
                if (l == wordLength) {
                    int printed = k + wordLength - 1;
                    for (int row = 0; row < this.alphabet.size(); row++) {
                        for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                            if (col == i && row >= k && row <= printed) {
                                System.out.print(this.alphabet.get(row).get(col) + " ");
                            } else {
                                System.out.print("- ");
                            }
                        }
                        System.out.print("\n");
                    }
                    System.out.println("Jumlah komparasi huruf: " + (lettersCompared));
                    System.out.println("\n");
                    found = true;
                } else {
                    lettersCompared++;
                    k++;
                }
            }
        }
    }
}

}
```

```
// Pencarian vertical dari bawah

public void fromBottomVertical() {
    for (int j = 0; j < this.word.size(); j++) { // loop untuk tiap kata dalam kunci jawaban
        int lettersCompared = 0;
        int wordLength = this.word.get(j).length();
        for (int i = 0; i < this.alphabet.get(0).size(); i++) { // loop untuk tiap kolom matriks
            boolean found = false;
            int textLength = this.alphabet.size();
            int k = textLength - 1;
            while (k >= (wordLength - 1) && !found) { // loop dalam pemeriksaan karakter
                int l = 0;
                while ((l < wordLength) && (this.alphabet.get(k - l).get(i).equals(String.valueOf(this.word.get(j).charAt(l))))) { // loop dalam pemeriksaan karakter
                    lettersCompared++;
                    l++;
                }
                if (l == wordLength) {
                    int printed = k + 1;
                    for (int row = 0; row < this.alphabet.size(); row++) {
                        for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                            if (col == i && row >= printed && row <= printed + wordLength - 1) {
                                System.out.print(this.alphabet.get(row).get(col) + " ");
                            } else {
                                System.out.print("- ");
                            }
                        }
                        System.out.print("\n");
                    }
                    System.out.println("Jumlah komparasi huruf: " + (lettersCompared));
                    System.out.println("\n");
                    found = true;
                } else {
                    lettersCompared++;
                    k--;
                }
            }
        }
    }
}

}
```

```

public void fromLeftTopDiagonal() {
    for (int j = 0; j < this.word.size(); j++) { // loop untuk tiap kata dalam kunci jawaban
        int lettersCompared = 0;
        int wordLength = this.word.get(j).length();
        for (int i = 0; i < this.alphabet.size(); i++) { // loop tiap baris puzzle
            if (i == 0) {
                for (int for_row_0 = 0; for_row_0 < this.alphabet.get(0).size(); for_row_0++) { // khusus baris pertama akan diloop untuk tiap kolomnya
                    int kol = for_row_0;
                    boolean found = false;
                    int ct_row = i;
                    int ct_col = kol;
                    int textlength = 0;
                    while (ct_row <= this.alphabet.size() && ct_col <= this.alphabet.get(0).size()) {
                        textlength++;
                        ct_row++;
                        ct_col++;
                    }
                    int k = i;
                    while (k <= textlength - wordLength && !found) {
                        int l = 0;
                        while (l < wordLength && (this.alphabet.get(k + l).get(kol + l).equals(String.valueOf(this.word.get(j).charAt(l))))) {
                            lettersCompared++;
                            l++;
                        }
                        if (l == wordLength) { // match pada baris k dan kolom kol
                            int row_printed = k;
                            int col_printed = kol;
                            for (int row = 0; row < this.alphabet.size(); row++) {
                                for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                                    if (row == row_printed && col == col_printed) {
                                        System.out.print(this.alphabet.get(row).get(col) + " ");
                                        if (col_printed < kol + wordLength - 1 && row_printed < k + wordLength - 1) {
                                            row_printed++;
                                            col_printed++;
                                        }
                                    } else {
                                        System.out.print("- ");
                                    }
                                }
                                System.out.print("\n");
                            }
                        }
                    }
                    System.out.println("Jumlah komparasi huruf: " + (lettersCompared));
                    System.out.println("\n");
                }
            }
        }
    }
}

```

```

        found = true;
    } else {
        lettersCompared ++;
        k ++;
        kol ++;
    }
}

} else {
    int kol = 0;
    boolean found = false;
    int ct_row = i;
    int ct_col = kol;
    int textLength = 0;
    while (ct_row <= this.alphabet.size() && ct_col <= this.alphabet.get(0).size()) {
        textLength ++;
        ct_row ++;
        ct_col ++;
    }
    int k = i;
    while (kol <= textLength - wordLength && !found) {
        int l = 0;
        while (l < wordLength && (this.alphabet.get(k + 1).get(kol + 1).equals(String.valueOf(this.word.get(j).charAt(l)))) {
            lettersCompared ++;
            l ++;
        }
        if (l == wordLength) { // match pada baris k dan kolom kol
            int row_printed = k;
            int col_printed = kol;
            for (int row = 0; row < this.alphabet.size(); row++) {
                for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                    if (row == row_printed && col == col_printed) {
                        System.out.print(this.alphabet.get(row).get(col) + " ");
                        if (col_printed < kol + wordLength - 1 && row_printed < k + wordLength - 1) {
                            row_printed ++;
                            col_printed ++;
                        }
                    } else {
                        System.out.print("- ");
                    }
                }
            }
            System.out.print("\n");
        }
    }
}

```

```

    } else {
        System.out.print("- ");
    }
}
System.out.print("\n");
}
System.out.println("Jumlah komparasi huruf: " + (lettersCompared));
System.out.println("\n");
found = true;
} else {
    lettersCompared ++;
    k ++;
    kol --;
}
}
} else {
    int kol = this.alphabet.get(0).size() - 1;
    boolean found = false;
    int ct_row = i;
    int ct_col = kol;
    int textLength = 0;
    while (ct_row <= this.alphabet.size() && ct_col >= 0) {
        textLength ++;
        ct_row ++;
        ct_col --;
    }
    int k = i;
    while (k <= textLength - wordLength && !found) {
        int l = 0;
        while (l < wordLength && (this.alphabet.get(k + 1).get(kol - l).equals(String.valueOf(this.word.get(j).charAt(l)))))) {
            lettersCompared ++;
            l ++;
        }
        if (l == wordLength) { // match pada baris k dan kolom kol
            int row_printed = k;
            int col_printed = kol;
            for (int row = 0; row < this.alphabet.size(); row++) {
                for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                    if (row == row_printed && col == col_printed) {
                        System.out.print(this.alphabet.get(row).get(col) + " ");
                        if (col_printed > kol - wordLength + 1 && row_printed < k + wordLength - 1) {
                            row_printed ++;
                            col_printed --;

```







```

        } else {
            lettersCompared++;
            k--;
            kol--;
        }
    }
} else {
    int kol = this.alphabet.get(0).size() - 1;
    boolean found = false;
    int ct_row = i;
    int ct_col = kol;
    int textlength = 0;
    while (ct_row >= 0 && ct_col >= 0) {
        textlength++;
        ct_row--;
        ct_col--;
    }
    int k = i;
    while (k >= wordLength - 1 && !found) {
        int l = 0;
        while (l < wordLength && (this.alphabet.get(k - 1).get(kol - 1).equals(String.valueOf(this.word.get(j).charAt(l)))) {
            lettersCompared++;
            l++;
        }
        if (l == wordLength) { // match pada baris k dan kolom kol
            int row_printed = k - wordLength + 1;
            int col_printed = kol - wordLength + 1;
            for (int row = 0; row < this.alphabet.size(); row++) {
                for (int col = 0; col < this.alphabet.get(0).size(); col++) {
                    if (row == row_printed && col == col_printed) {
                        System.out.print(this.alphabet.get(row).get(col) + " ");
                        if (col_printed < kol && row_printed < k) {
                            row_printed++;
                            col_printed++;
                        }
                    } else {
                        System.out.print("- ");
                    }
                }
                System.out.print("\n");
            }
        }
    }
}

System.out.println("Jumlah komparasi huruf: " + (lettersCompared));
System.out.println("\n");
found = true;
} else {
    lettersCompared++;
    k--;
    kol--;
}
}
}
}

public void solvePuzzle() {
    System.out.println("Berikut adalah solusi dari puzzle yang disubmit: ");
    System.out.println("\n");
    fromLeftHorizontal();
    fromRightHorizontal();
    fromTopVertical();
    fromBottomVertical();
    fromLeftTopDiagonal();
    fromRightTopDiagonal();
    fromLeftBottomDiagonal();
    fromRightBottomDiagonal();
}
}

```

## B. File App.java

```
import java.util.Scanner;
import java.lang.System;

public class App {
    Run | Debug
    public static void main(String[] args) throws Exception {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan nama file (.txt): ");
        String filename = scanner.nextLine();
        WordSearchPuzzle x = new WordSearchPuzzle(filename);
        Long startTime = System.nanoTime();
        x.solvePuzzle();
        Long endTime = System.nanoTime();
        Long totalTime = endTime - startTime;
        System.out.println("Total waktu eksekusi program: " + totalTime + " ns");
        System.out.println("\n");
    }
}
```

## III. Pengujian Program

### A. Ukuran kecil 1

#### 1. Input

```
CMNORBKGCNTCFNT
LHRTIDNIAHRTSRI
LIIRCANVOIEAOTE
OFDMTORTOISEWT
TLEEPIDEERRRTRL
HYDSWALLOWNOEAT
CHICKENLSRAIRAH
WOLFLSCZKEPANDA
OLHENOQAEWALRUS
CAIAWNNUTEELNEE
TOIZMGHKIPLRRDA
OLRRASMETDEPSDD
PLVRTRTSSPIDERO
UDOAHEDEYGTSLU
SOCTESPARROWIES

BIRD
CAT
CHEETAH
CHICKEN
CHIMPANZEE
COW
DEER
FLY
HAMSTER
KANGAROO
LIZARD
OCTOPUS
OTTER
OWL
PANDA
PIG
RAT
SEAL
SNAIL
SPARROW
SPIDER
SQUID
SWALLOW
TORTOISE
WALRUS
WOLF
```

## 2. Output

Masukkan nama file (.txt): small1.txt  
Berikut adalah solusi dari puzzle yang disubmit:

CHICKEN

Jumlah komparasi huruf: 64

DEER

Jumlah komparasi huruf: 60

```
Jumlah komparasi huruf: 93
```

```
- - - - - SPARROW - - -  
Jumlah komparasi huruf: 146
```

```
Jumlah komparasi huruf: 139
```

Jumlah komparasi huruf: 55

[illegible][illegible]

```
- - - - -  
- - - - -  
- - - - -  
- - - - -  
W O L F - - - - -  
- - - - -  
- - - - -  
- - - - -  
Jumlah komparasi huruf: 90
```

```
- - - - -
```

```
F      
```

```
L      
```

```
Y      
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
Jumlah komparasi huruf: 19
```

```
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
O - - - -  
C - - - -  
T - - - -  
O - - - -  
P - - - -  
U - - - -  
S - - - -  
Jumlah komparasi huruf: 16
```

Jumlah komparasi huruf: 152

```
Jumlah komparasi huruf : 8
```

Jumlah komparasi huruf: 106

Jumlah komparasi huruf: 62

```
S -  
E  
A  
L
```

Jumlah komparasi huruf: 36

```

- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - H - - - - -
- - A - - - - -
- - M - - - - -
- - S - - - - -
- - T - - - - -
- E - - - - -
- R - - - - -
Jumlah komparasi huruf: 103

```

```
B  
- - I -  
- R -  
D -
```

Jumlah komparasi huruf: 76

```

- - - - - K - - - - -
- - - - - A - - - - -
- - - - - N - - - - -
- - - - - G - - - - -
- - - - - A - - - - -
- - - - - R - - - - -
- - - - - O - - - - -
- - - - - O - - - - -
Jumlah komparasi huruf: 50

```

Total waktu eksekusi program: 189830100 ns

FIREPLACEVTVBAT  
 HMUFFLERKWSWLMH  
 LJSEASONJEHBANC  
 GMMDDETWSIWFNFF  
 HQVOKSKQTGIMKIU  
 MWOTEGNEQYVOEFV  
 SHKYFIGUIKAXTIP  
 TRFEUMJANUARYZZ  
 XKIGJZWMMAO CGWP  
 SSNETANREBIHJYH  
 KETXPFUHEIYKHVI  
 PPDECEMBERYYLDR  
 CPXRERLIFYOZQEOSV  
 WPTHGLUVOCZOGWV  
 ZOGCABVCNTAQYEH

BLANKET  
 HIBERNATE  
 MUFFLER  
 COZY  
 HOOD  
 PENGUIN  
 DECEMBER  
 ICY  
 SEASON  
 FIREPLACE  
 JANUARY  
 WHITE

## 2. Output

```
-----
- S E A S O N -
-----
Jumlah komparasi huruf: 28

F I R E P L A C E -----
-----
Jumlah komparasi huruf: 9

-----
Masukkan nama file (.txt): small2.txt
Berikut adalah solusi dari puzzle yang disubmit:

-----
- M U F F L E R -
-----
Jumlah komparasi huruf: 17

-----
- D E C E M B E R -
-----
Jumlah komparasi huruf: 101

-----
- J A N U A R Y -
-----
Jumlah komparasi huruf: 77

-----
- E T A N R E B I H -
-----
Jumlah komparasi huruf: 78

-----
- N
- I
- U
- G
- N
E
P
-----
Jumlah komparasi huruf: 79

-----
- Y
- C
- I
-----
Jumlah komparasi huruf: 71

Total waktu eksekusi program: 87153000 ns

-----
- Y
- Z
- O
- C
-----
Jumlah komparasi huruf: 76

-----
- D
- O
- O
- H
-----
Jumlah komparasi huruf: 150

-----
- B
- L
- A
- N
- K
- E
- T
-----
Jumlah komparasi huruf: 116

-----
- W
- H
- I
- T
- E
-----
Jumlah komparasi huruf: 29
```

### C. Ukuran kecil 3

## 1. Input

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

BEAR  
 ELEPHANT  
 GIRAFFE  
 GOAT  
 LION  
 PANDA  
 PENGUIN  
 TIGER

## 2. Output

```
Masukkan nama file (.txt): small3.txt
Berikut adalah solusi dari puzzle yang disubmit:
```

```
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
T I G E R - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Jumlah komparasi huruf: 47
```

```
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
T A O G - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Jumlah komparasi huruf: 78
```

```

- - - - - E -
- - - - - F -
- - - - - F -
- - - - - A -
- - - - - R -
- - - - - I -
- - - - - G -
- - - - -
Jumlah komparasi huruf: 65

E - - - - -
L - - - - -
- E - - - - -
- P - - - - -
- H - - - - -
- A - - - - -
- N - - - - -
- T - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Jumlah komparasi huruf: 8
```

```
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - B - -  
- - E - -  
- A - - -  
R - - - -  
- - - - -  
- - - - -  
  
Jumlah komparasi huruf: 60  
  
- - - - - P  
- - - - - E  
- - N - -  
- G - - -  
U - - - -  
I - - - -  
N - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
  
Jumlah komparasi huruf: 7
```



[illegible]

#### D. Ukuran medium 1

## 1. Input

MOTRNBTEBNNMRYKJHHH  
ZOXJOOHUFHVEURBCNE  
WPUGATXOYFKVROUYON  
EPODPZATMNAEETQAGG  
EAZKEUNGOEGRCSDFCU  
TADNAPOMIIEUIHPNRT  
LNNPFKIRTLFRVGDENV  
GORFWCLUPLLYKNCOAO  
XWPZSTHOOJRAIABYQG  
XZIIAGJIXXOKHJKTEKD  
DKOATXENWGV EIBADR  
NRTWJAENNBBLEYWJCQ  
ELEPHANTAHYCKFWGDB  
QYFWPHCYFPFDJJSBHFVE  
RINQVJJSXLQGFVPYXKL  
ZSUMSIKBMGMBYGDPTA  
DQMUMSNGIGDPBNVOIS  
JEBINQIJMSNIQHIIYDV

ALLIGATOR  
FROG  
GIRAFFE  
LION  
PANDA  
TIGER  
ELEPHANT  
TOAD  
PENGUIN  
GOAT  
MEERKAT  
MONKEY  
STORK

## 2. Output

Masukkan nama file (.txt): medium1.txt  
Berikut adalah solusi dari puzzle yang disubmit:

ELEPHANT

Jumlah komparasi huruf: 148

G O R F

Jumlah komparasi huruf: 128

Jumlah komparasi huruf: 106

Jumlah komparasi huruf: 207

- A D N A P

Jumlah komparasi huruf: 90

P  
E  
N  
G  
U  
I  
N

Jumlah komparasi huruf: 220

-	-	-	-	-	-	N
-	-	-	-	-	-	O
-	-	-	-	-	-	I
-	-	-	-	-	-	L

Jumlah komparasi huruf: 106

[illegible]

Jumlah komparasi huruf: 207

M  
E  
E  
R  
K  
A  
T

Jumlah komparasi huruf: 67

T  
 O  
 A  
 D

```
Jumlah komparasi huruf: 120
```



E. Ukuran medium 2

1. Input

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

ALLIGATOR
BEAR
ELEPHANT
FROG
GIRAFFE
LION
MEERKAT
MONKEY
STORK
TIGER
TOAD
```

## 2. Output

Masukkan nama file (.txt): medium2.txt  
Berikut adalah solusi dari puzzle yang disubmit:



ALLIGATOR

— — — — —

Jumlah komparasi huruf: 174

- - - - - G I R A F F E - - - - -

Jumlah komparasi huruf: 44

TAKREEM

Jumlah komparasi huruf: 31

MONKEY

Jumlah komparasi huruf: 113



Jumlah komparasi huruf: 62

Jumlah komparasi huruf: 246

Jumlah komparasi huruf: 6

Total waktu eksekusi program: 130432000 ns

F. Ukuran medium 3

1. Input

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

ASSIGN
CONCAVE
DEPARTED
DOCTOR
FATTEST
GAMING
INVOLVE
LAUGHED
LUNA
OBOE
PULVERIZED
SNUGNESS
SOBER
```



## 2. Output

Masukkan nama file (.txt): medium3.txt  
Berikut adalah solusi dari puzzle yang disubmit:

[illegible]

```

- - - - - P U L V E R I Z E D
Jumlah komparasi huruf: 290

```

[illegible]

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- A - - - - -
```

```
- S - - - - -
```

```
- S - - - - -
```

```
- I - - - - -
```

```
- G - - - - -
```

```
- N - - - - -
```

```
- - - - -
```

```
- - - - -
```

```
- - - - -
```

```
Jumlah komparasi huruf: 37
```

[illegible][illegible]

Jumlah komparasi huruf: 56

[illegible]



## 2. Output

Masukkan nama file (.txt): large1.txt  
Berikut adalah solusi dari puzzle yang disubmit:

Jumlah komparasi huruf: 519

Jumlah komparasi huruf: 686

- - - - - C A T - - - - -

Jumlah komparasi huruf: 435

```
- - - - G O D - - - -
```

Jumlah komparasi huruf: 58

- - - - T A K R E E M

Jumlah komparasi huruf: 570

NIUGNEP -

Jumlah komparasi huruf: 292

M  
O  
N  
K  
E  
Y

Jumlah komparasi huruf: 85

T  
I  
G  
E  
R

Jumlah komparasi huruf: 563





S  
T  
O  
R  
K

Jumlah komparasi huruf: 125

E  
L  
E  
P  
H  
A  
N  
T

Jumlah komparasi huruf: 257

R  
A  
E  
B

Jumlah komparasi huruf: 564

D  
R  
I  
B

Jumlah komparasi huruf: 374



## H. Ukuran besar 2

### 1. Input

```
ZSDIDXWJOAQENZCUQTJZXDYPIBJKQQV
MQGGZUMZHYLKAXGAVAPEFILSYVLRCAH
ZFGNJIEIQZIRWAWJMDOTSLRYCAPVAYBL
KHKDYNKBRMPDJPOUBSCASFAMIBZTWXE
PTBQBFTFMRJVAALDDHIYTSUHKKZSXMQ
WXLERYIESSOMWBEZACSXZEEZUUAEWCT
DJTVNBMMWNHHKDAADRIMUSGNMLEEJKTUD
BQLFYOBDBGBHOLLGD AFGKSRZGEKYOFYZ
EJLGRAQWHA EISEUER RBPUORGCKNMLIH
HGYADVPTYTYTFWJTBCORXVYJGOYOSUQE
SYNJTHKXBT SJOREAUOEYGRCTPZNMCTB
YDZCXIDSABSRBR SVIKYHBQEPYWSDSNM
ANYVILJAEHFALRENFAMJLZBFHMUTVHA
IZAXIITXWJGMKMRHDRFU FFXS VX YUBVY
ZJZCVYCHYBXIBMFOAEQPAIXJICUGLCO
MACAROONEPLPFLGAXNITTDOTOMJMKWN
BEEFDYAXMWTIOQIPBLDLVCBANFTNYWN
O AIFVMLQSAXZLDXLGLZQPTCPDNITZCA
VMFXTAQCYRCASYWGHJGUSMELHWHTROI
VRPOYFRWKFTHDVVZWN NHBMIRGXXP HOS
ORMOCFKCDNQHQXOZBRRLYOTKU HLEVTE
CGWATWAMREYSIRUGIXWBDIWNXSHRAFJ
RVARNPZJ OEXVYQB FZUWLXBQIEZLYPVR
EVOMYALRUOGIXUEDTBKHHKAENHXORCC
EDDSYJGHGXF TUDTLFXUCHUMJWLZTYM
SEIFXCKEOITW IETGSZNV TQNV RH SNTAH
GMEVKXKFMPXRTBDW WKRCOPWGEOTIJW
OOFVPWLSZEZYUHRHUWXHGSEGKHPXXFR
JDPZRDBSDNNXPOEDBEBZZWUOMRZDYCW
DXXXTPFEXNTVKWPKUYAWOEQH QXXXOL
DSCSEENLSULJVPILYHZUPKOFPECZZZU
```

ATTILA  
BEEF  
BEFOREHAND  
CAJOLED  
DISCHARGE  
GROUP  
IMPLIED  
JAZZ  
JUDICIALLY  
MACARON  
MANAGEMENT  
MAYONNAISE  
MEMORANDA  
MOVE  
SMOKELESS  
SMOKESTACK  
TEAMS

## 2. Output

Masukkan nama file (.txt): large2.txt  
Berikut adalah solusi dari puzzle yang disubmit:

B E E F

Jumlah komparasi huruf: 473

MACARON

Jumlah komparasi huruf: 382

P U O R G

Jumlah komparasi huruf: 234

E V O M

Jumlah komparasi huruf: 702







B  
 E  
 F  
 O  
 R  
 E  
 H  
 A  
 N  
 D

Jumlah komparasi huruf: 66

M  
A  
N  
A  
G  
E  
M  
E  
N  
T

Jumlah komparasi huruf: 547

Jumlah komparasi huruf: 184

Jumlah komparasi huruf: 285

Jumlah komparasi huruf: 618

Jumlah komparasi huruf: 432



