

Laporan Tugas Kecil 1 IF2211 Strategi Algoritma

Semester II Tahun Akademik 2021/2022

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



Disusun oleh :

Muhammad Gilang Ramadhan

13520137

K-02

**Program Studi S1 Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2022**

BAB 1. Algoritma *Brute force* untuk menyelesaikan *Word Search Puzzle*

World Search Puzzle adalah sebuah game mencari kata-kata yang tertulis dari soal di game tersebut pada sebuah matriks (*puzzle*) yang berisi alfabet acak dengan jawaban yang tersembunyi pada matriks (*puzzle*) tersebut. Adapun berikut langkah-langkah penyelesaian persoalan tersebut dengan algoritma *Brute Force*, yaitu :

1. Tinjau setiap alfabet pada setiap baris dan setiap kolom untuk memeriksa huruf pertama setiap kata yang ingin dicari.
2. Apabila huruf pertama dari kata tersebut cocok, maka cukup periksa tetangga dari baris dan kolom tersebut dengan cara memeriksa 8 arah dari tetangganya satu persatu.
3. Dari setiap arah dari 8 arah tersebut, apabila pengecekan huruf kedua pada setiap arah dari 8 arah tersebut cocok, maka lanjutkan ke pengecekan huruf ketiga dengan masing-masing arah sebelumnya.
4. Jika tidak ditemukan alfabet yang sesuai dengan pengecekan alfabet dari kata yang ingin ditemukan, maka cukup dilakukan pemeriksaan ke alfabet pada baris dan kolom selanjutnya.
5. Jika kata yang dicari ditemukan pada matriks (*puzzle*) maka cukup simpan koordinat dari matriks tersebut berupa indeks matriks tersebut sendiri.
6. Lakukan pengulangan dari langkah ke-1 hingga langkah ke-5 tersebut sampai semua alfabet selesai diperiksa.

BAB 2. Source Program dalam bahasa C++

```
#include <bits/stdc++.h>
#include <windows.h>
using namespace std;

/* Menginisialisasikan Variabel Global */
HANDLE h = GetStdHandle(STD_OUTPUT_HANDLE);
long long banyak_ pengecekan = 0;
vector<vector<int> > vecX;
vector<vector<int> > vecY;
vector<vector<int> > Penunjuk_arah;
vector<string>tampung_kata;
vector<vector<char> >tampung_char;
bool file_berhasil_dibuka = true;
// Menginisialisasikan arah gerak (x,y) yang berguna untuk pengecekan 8 arah
int moveX[] = { -1, -1, -1, 1, 1, 1, 0, 0 };
int moveY[] = { -1, 0, 1, -1, 0, 1, -1, 1 };
```

```
// Procedure untuk memasukkan data isi file txt ke vektor
void convertTXT(){
    // Membuka File Inputan

    ifstream ifs;
    ifs.open("../test/Test 9_large_35 x 35.txt");

    if (ifs.is_open()){

        // Membaca Line

        string line;

        // Memasukkan karakter-karakter yang membentuk matriks dari file txt ke vektor yang menampung
        char-char tersebut dalam bentuk matriks
        while( getline( ifs, line ) )
        {
            vector<char> temp;
            if (!line.empty()){
                int indeks = 0;
                while ( indeks < line.length() ){
                    if (line[indeks] != ' '){
                        temp.push_back(line[indeks]);
                    }
                    indeks++;
                }
                tampung_char.push_back(temp);
            }
            else{
                break;
            }
        }

        // Memasukkan dan menampung karakter yang ingin dicari
        // Ke dalam container berupa vektor
        while( getline( ifs, line ) ){
            tampung_kata.push_back(line);
        }
    }
    else{
        cout << "File tidak bisa dibuka";
        file_berhasil_dibuka = false;
    }
    ifs.close();
}
```

```

// Fungsi pencarian karakter pada kata yang ingin dicari pada delapan arah pencarian di puzzle
bool Search_8_Direction(string word, vector<vector<char> >tampung_char, int row, int col, int moveX,
int moveY)
{
    // Inisialisasikan RD sebagai penunjuk baris dan CD sebagai penunjuk kolom yang akan bergerak
    sesuai dengan moveX dan moveY
    int j = 1, RD = row + moveY, CD = col + moveX;

    // Jika RD dan CD tidak overbound dan setiap huruf ke j cocok
    while ((j < word.length()) && (RD < tampung_char.size() && CD < tampung_char[row].size() && RD >= 0
&& CD >= 0) && (word[j] == tampung_char[RD][CD]))
    {
        // Jika cocok, bergerak lagi
        CD += moveX;
        RD += moveY;
        j++;
        banyak_pengecekan++;
    }

    // Jika semua karakter cocok, maka nilai dari j harus sama dengan panjang dari kata yang ingin
    dicari
    if (j == word.length()){
        return true;
    }
}

```

```

// Procedure untuk solusi dalam bentuk koordinat kata yang ingin dicari
void solve(){
    for (int k = 0; k < tampung_kata.size(); k++){
        // Algoritma Brute Force untuk mencari setiap kata yang ingin dicari di puzzle
        // Kemudian simpan setiap koordinat dari setiap karakter yang terdapat di puzzle dari kata yang
        // ingin dicari
        for (int row = 0; row < tampung_char.size(); row++){
            for (int col = 0; col < tampung_char[row].size(); col++){
                if (tampung_kata[k][0] == tampung_char[row][col])
                {
                    for (int arah = 0; arah < 8; arah++)
                    {
                        // Jika digit ke-2 sampai digit terakhir dari huruf yang akan diperiksa di
                        // Maka masukkan koordinat-koordinat huruf tersebut pada puzzle
                        // Dengan cara memasukkan masing-masing absis dan ordinat tersebut ke dalam
                        tempX dan tempY
                        // kemudian masukkan ke dalam vecX dan VecY
                        if (Search_8_Direction(tampung_kata[k], tampung_char, row, col, moveX[arah],
                        moveY[arah])){
                            vector<int> tempPenunjukArah;
                            vector<int> tempX;
                            vector<int> tempY;
                            for (int i = 0; i < tampung_kata[k].length(); i++){
                                tempX.push_back(col + (i * moveX[arah]));
                                tempY.push_back(row + (i * moveY[arah]));

                                // Menyimpan arah dari setiap titik
                                if (moveX[arah] == -1 & moveY[arah] == -1){
                                    tempPenunjukArah.push_back(1);
                                }
                                else if (moveX[arah] == -1 & moveY[arah] == 1){
                                    tempPenunjukArah.push_back(2);
                                }
                                else if (moveX[arah] == 1 & moveY[arah] == -1){
                                    tempPenunjukArah.push_back(3);
                                }
                                else if (moveX[arah] == 1 & moveY[arah] == 1){
                                    tempPenunjukArah.push_back(4);
                                }
                                else if (moveX[arah] == 0 & moveY[arah] == -1){
                                    tempPenunjukArah.push_back(5);
                                }
                                else if (moveX[arah] == -1 & moveY[arah] == 0){
                                    tempPenunjukArah.push_back(6);
                                }
                                else if (moveX[arah] == 0 & moveY[arah] == 1){
                                    tempPenunjukArah.push_back(7);
                                }
                                else if (moveX[arah] == 1 & moveY[arah] == 0){
                                    tempPenunjukArah.push_back(8);
                                }
                            }
                            vecX.push_back(tempX);
                            vecY.push_back(tempY);
                            Penunjuk_arah.push_back(tempPenunjukArah);
                        }
                    }
                }
            }
        }
        banyak_pengecekan++;
    }
}
}
}
}

```

```

// Fungsi untuk mengecek apakah indeks ke indeks_row = vecY dan dan indeks_col = vecX
// Jika benar akan mengembalikan sesuai dengan arahnya yaitu 1,2,3,...,8
// Jika salah maka akan mengembalikan 0
int Check_Indeks(vector<vector<int> > vecX, vector<vector<int> > vecY, int indeks_row, int indeks_col){
    for (int k = 0; k < vecX.size(); k++){
        for (int L = 0; L < vecX[k].size(); L++){
            if ((indeks_row == vecY[k][L]) & (indeks_col == vecX[k][L])){
                return Penunjuk_arah[k][L];
            }
        }
    }
    return 0;
}

```

```

// Procedure untuk menampilkan Solusi berbentuk matriks ke layar
void PrintSolusi(){
    for (int i = 0; i < tampung_char.size(); i++){
        for (int j = 0; j < tampung_char[i].size(); j++){
            // Mewarnai karakter dengan warna tertentu
            // Sesuai dengan arahnya
            if (Check_Indeks(vecX, vecY, i, j) == 1){
                SetConsoleTextAttribute(h, 1);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 2){
                SetConsoleTextAttribute(h, 2);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 3){
                SetConsoleTextAttribute(h, 4);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 4){
                SetConsoleTextAttribute(h, 5);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 5){
                SetConsoleTextAttribute(h, 6);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 6){
                SetConsoleTextAttribute(h, 9);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 7){
                SetConsoleTextAttribute(h, 3);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 8){
                SetConsoleTextAttribute(h, 13);
                cout << tampung_char[i][j] << " ";
            }
            else if (Check_Indeks(vecX, vecY, i, j) == 0){
                // Jika tidak warnai putih pada setiap karakter yang akan diprint
                SetConsoleTextAttribute(h, 7);
                cout << tampung_char[i][j] << " ";
            }
        }
        cout << endl;
    }
    SetConsoleTextAttribute(h, 7);
}

```

```

// Main Program
int main()
{
    convertTXT();
    if (file_berhasil_dibuka){
        // Waktu eksekusi dimulai
        clock_t start, end;
        start = clock();

        // Proses melakukan eksekusi brute force
        solve();

        // Waktu eksekusi selesai
        end = clock();
        double time_taken = double(end - start) / double(CLOCKS_PER_SEC);

        // Menampilkan solusi
        PrintSolusi();

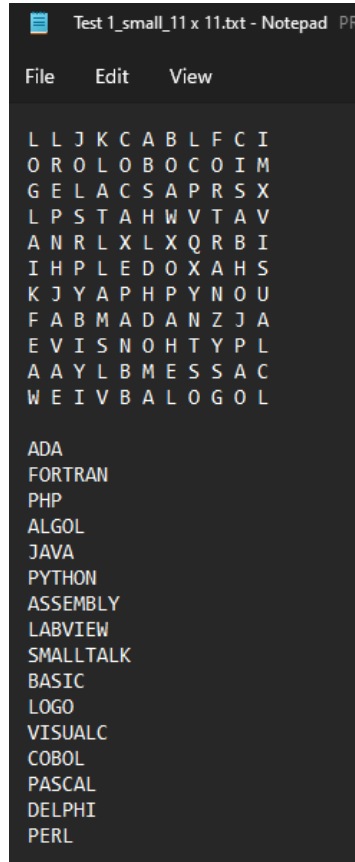
        cout << endl;
        // Menampilkan waktu eksekusi program
        cout << "Waktu Eksekusi Program : " << fixed << time_taken * 1000 << setprecision(5);
        cout << " ms " << endl;

        // Menampilkan jumlah pengecekan kata pada program
        cout << "Jumlah Total pengecekan Huruf : " << banyak_pengecekan << endl;
    }
    return 0;
}

```

BAB 3. Pengujian (Input/Output)

3.1 Test Case 1 (ukuran *small*)

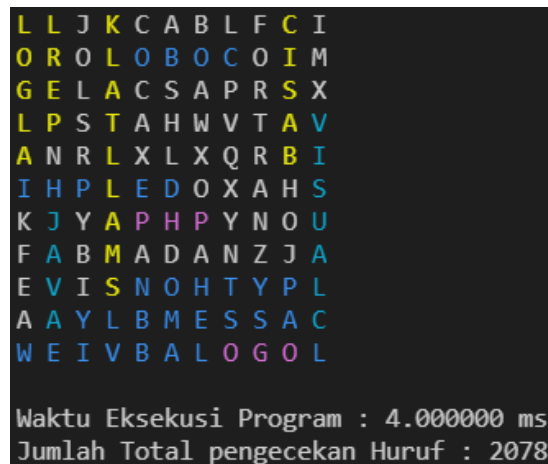


```
File Edit View

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

ADA
FORTRAN
PHP
ALGOL
JAVA
PYTHON
ASSEMBLY
LABVIEW
SMALLTALK
BASIC
LOGO
VISUALC
COBOL
PASCAL
DELPHI
PERL
```

Gambar 3.1.1 Input Test Case 1

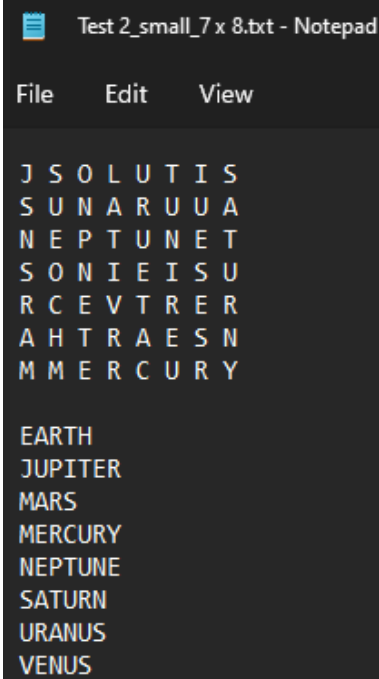


```
L L J K C A B L F C I
O R O L O B O C O I M
G E L A C S A P R S X
L P S T A H W V T A V
A N R L X L X Q R B I
I H P L E D O X A H S
K J Y A P H P Y N O U
F A B M A D A N Z J A
E V I S N O H T Y P L
A A Y L B M E S S A C
W E I V B A L O G O L

Waktu Eksekusi Program : 4.000000 ms
Jumlah Total pengecekan Huruf : 2078
```

Gambar 3.1.2 Output Test Case 1

3.2 Test Case 2 (ukuran *small*)

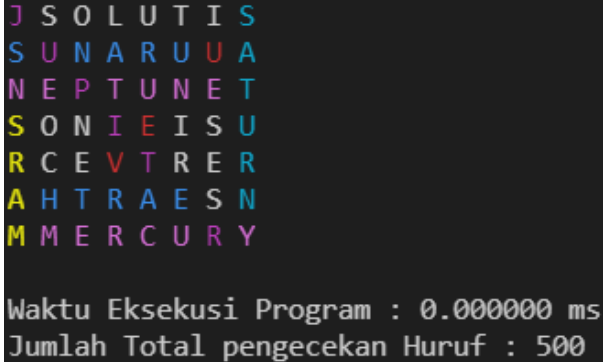


```
Test_2_small_7 x 8.txt - Notepad
File Edit View

JSOLUTIS
SUNARUUA
NEPTUNET
SONIEISU
RCEVTRER
AHTRAESN
MMERCURY

EARTH
JUPITER
MARS
MERCURY
NEPTUNE
SATURN
URANUS
VENUS
```

Gambar 3.2.1 Input Test Case 2

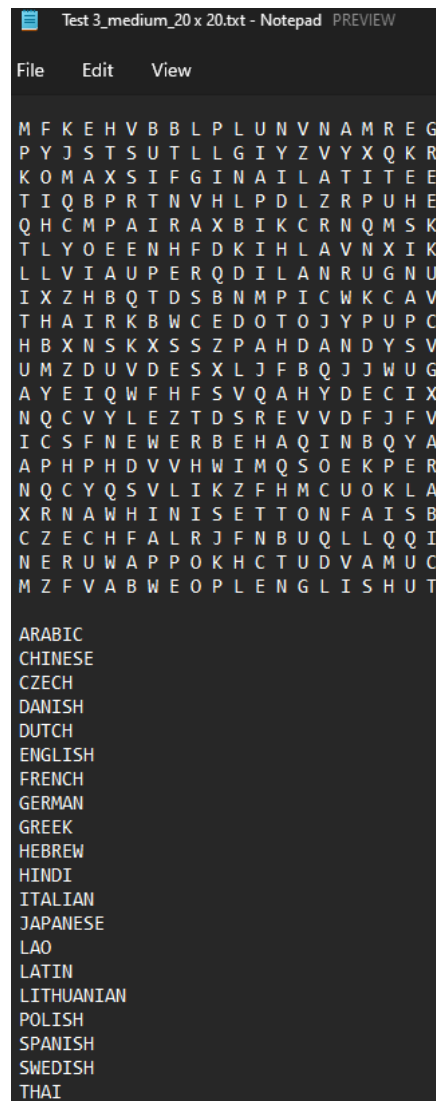


```
JSOLUTIS
SUNARUUA
NEPTUNET
SONIEISU
RCEVTRER
AHTRAESN
MMERCURY

Waktu Eksekusi Program : 0.000000 ms
Jumlah Total pengecekan Huruf : 500
```

Gambar 3.2.2 Output Test Case 2

3.3 Test Case 3 (ukuran *medium*)



Gambar 3.3.1 Input Test Case 3

```

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

Waktu Eksekusi Program : 13.000000 ms
Jumlah Total pengecekan Huruf : 8243

```

Gambar 3.3.2 Output Test Case 3

3.4 Test Case 4 (ukuran *medium*)

```

test_4_medium_20 x 20.txt - Notepad PREVIEW
File Edit View

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

BERKELEY
BRANDEIS
BROWN
COLUMBIA
CORNELL
DARTMOUTH
DUKE
EMORY
GEORGETOWN
HARVARD
NORTHEASTERN
NOTREDAME
OXFORD
PRINCETON
PURDUE
RICE
STANFORD
TUFTS
VANDERBILT
YALE

```

Gambar 3.4.1 Input Test Case 4

```

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

Waktu Eksekusi Program : 16.000000 ms
Jumlah Total pengecekan Huruf : 8281

```

Gambar 3.4.2 Output Test Case 4

3.5 Test Case 5 (ukuran *medium*)

```

Test_5_medium_22 x 22.txt - Notepad PREVIEW
File Edit View

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

ALBANIA
GERMANY
NORWAY
ANDORRA
GREECE
POLAND
ARMENIA
HUNGARY
PORTUGAL
AUSTRIA
ICELAND
ROMANIA
AZERBAIJAN
IRELAND
RUSSIA
BELARUS
ITALY
SANMARINO
BELGIUM
KAZAKHSTAN
SERBIA
BOSNIAHERZEGOVINA
LATVIA
SLOVAKIA
BULGARIA
LIECHTENSTEIN
SLOVENIA

```

Gambar 3.5.1 Input Test Case 5

```

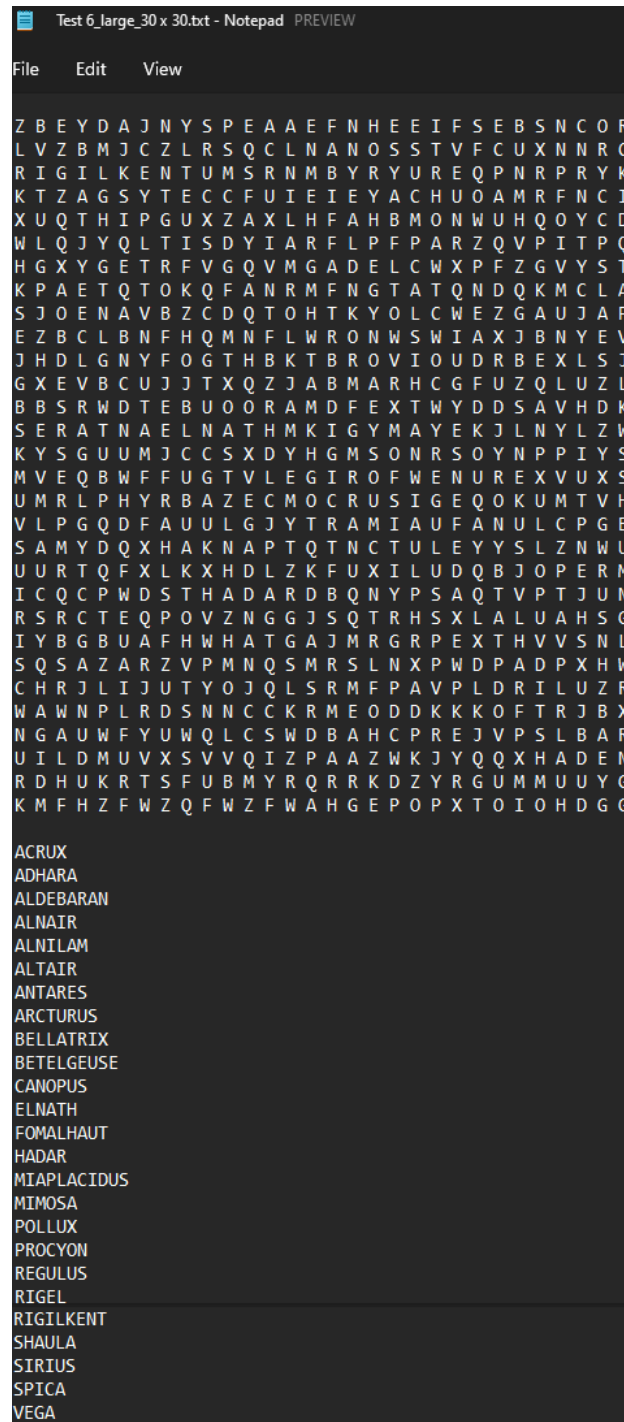
T C S D N A L R E H T E N D N A L O P Y S M
R R I A I N A U H T I L I G I D S K B D L O
Q S B Y V Q U A L D T B K T R E M N K N O N
M W E L B Y I F E B I R A R R E A H S A V A
A E L P A R A I R A G L U B A T E U O L A C
C D A O T B F K J A Y V I S S M R C P R K O
E E R S O N I R A M N A S H S P N U E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

Waktu Eksekusi Program : 37.000000 ms
Jumlah Total pengecekan Huruf : 13594

```

Gambar 3.5.2 Output Test Case 5

3.6 Test Case 6 (ukuran *large*)



```
Test_6_large_30 x 30.txt - Notepad PREVIEW
File Edit View

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

ACRUX
ADHARA
ALDEBARAN
ALNAIR
ALNILAM
ALTAIR
ANTARES
ARCTURUS
BELLATRIX
BETELGEUSE
CANOPUS
ELNATH
FOMALHAUT
HADAR
MIAPLACIDUS
MIMOSA
POLLUX
PROCYON
REGULUS
RIGEL
RIGILKENT
SHAULA
SIRIUS
SPICA
VEGA
```

Gambar 3.6.1 Input Test Case 6

```

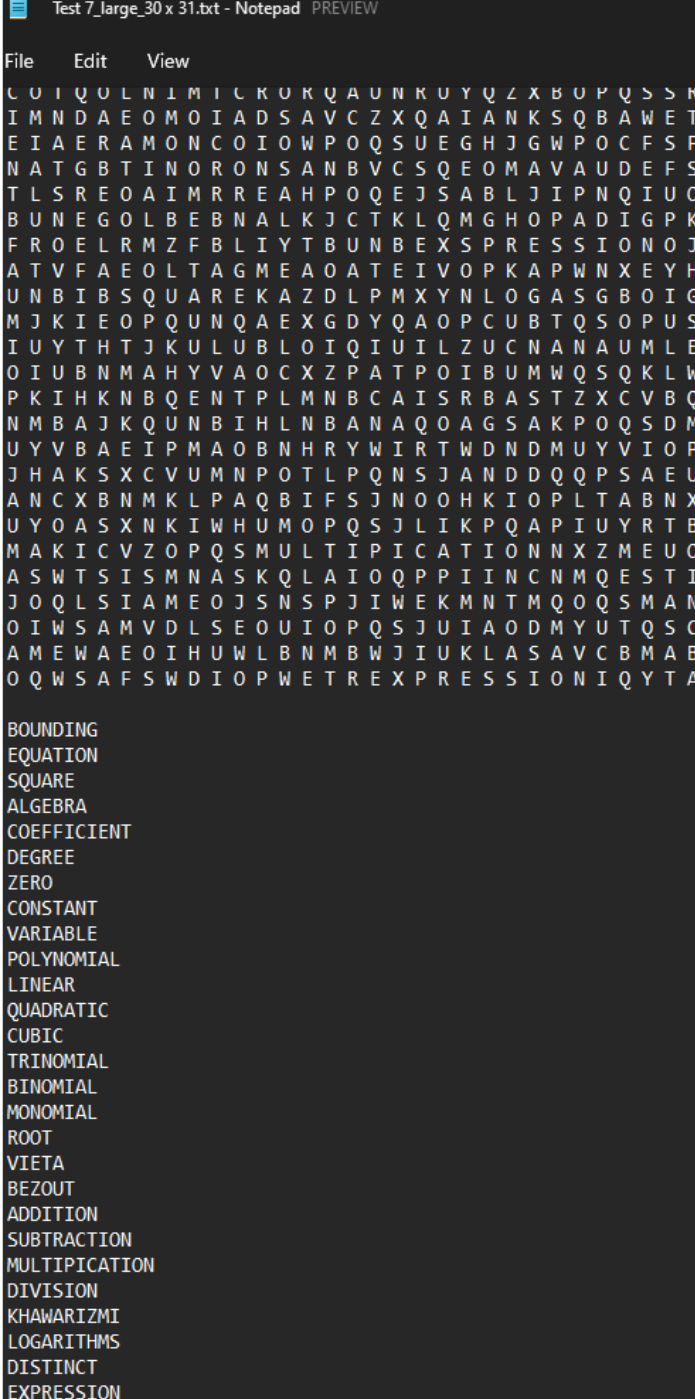
ZBEYDAJNYSPEAAEFNHEEIFSEBSNCR
LVZBMJCZLRSQCLNANOSSTVFCUXNNRC
RIGILKENTUMSRNMBYRYUREQPNRPRYK
KTZAGSYTECCFUIEIEYACHUOAMRFNCI
XUQTHIPGUXZAXLHFAHBMONWUHQOYCD
WLQJYQLTISDYIARFLPFPAARZQVPITPQ
HGXYGERTRFVGVGMGADEL CWXP FZGVYST
KPAETQTOKQFANR MFNGTATQNDQKMCCLA
SJOENAVBZCDQTOHTKYOLCWEZGAUJAP
EZBCLBNFHQMNF LWRONWSWIAXJBNEYEV
JHDLGNYFOGTHBKT BROVIOUDRBEXLSJ
GXEVBCUJJTXQZJABMARHCGFUZQLUZL
BBSRWDTEBUOORA MDFEXTWYDD SAVHDK
SERATNAELNATHMKIGYMAYEKJLNYLZW
KYSGUUMJCCSX DYHGMSONRSOYNPPIYS
MVEQBWF FUGTVLEGIROFWENUREXVUXS
UMRLPHYRBAZE CMOCRUSIGEOKUMTVH
VLPQGQDFAUULGJYTRAMI AU FANULCPGB
SAM YDQXHAKNAPTQTNC TULEYYS LZNWU
UURTQFXLKXHD LZKFUXILUDQBJ OPERM
ICQCPWDSTHADARDBQNYPSAQTVPTJUN
RSRCTEQPOVZNGGJSQTRHSXLALUAHSG
IYBGBUAFHWHATGAJMRGRPEXTHVVSNL
SQSAZAZVPMNQSMRSLNXPWDPADPXHW
CHRJLIJUTYOJQLSRMFPAVPLDRILUZR
WAWNPLRDNNCCKRMEODDKKKOFTRJJBX
NGAUWFYUWQLCSWDBAHCPREJVPSSLBAR
UITLDMUVXS VVQIZPA AZWKJYQQXHADEN
RDHUKRTSFUBMYRQR RKDZYRGUMMUUYG
KMFHZFWZQFWZFWAHGEP OPXT OI OHDGG

```

Waktu Eksekusi Program : 64.000000 ms
Jumlah Total pengecekan Huruf : 23057

Gambar 3.6.2 Output Test Case 6

3.7 Test case 7 (ukuran *large*)



```
Test_7_large_30 x 31.txt - Notepad PREVIEW
File Edit View
C O I Q U L N I M I C K O R Q A U N K U Y Q Z X B O P Q S S K
I M N D A E O M O I A D S A V C Z X Q A I A N K S Q B A W E T
E I A E R A M O N C O I O W P O Q S U E G H J G W P O C F S F
N A T G B T I N O R O N S A N B V C S Q E O M A V A U D E F S
T L S R E O A I M R R E A H P O Q E J S A B L J I P N Q I U O
B U N E G O L B E B N A L K J C T K L Q M G H O P A D I G P K
F R O E L R M Z F B L I Y T B U N B E X S P R E S S I O N O J
A T V F A E O L T A G M E A O A T E I V O P K A P W N X E Y H
U N B I B S Q U A R E K A Z D L P M X Y N L O G A S G B O I G
M J K I E O P Q U N Q A E X G D Y Q A O P C U B T Q S O P U S
I U Y T H T J K U L U B L O I Q I U I L Z U C N A N A U M L E
O I U B N M A H Y V A O C X Z P A T P O I B U M W Q S Q K L W
P K I H K N B Q E N T P L M N B C A I S R B A S T Z X C V B Q
N M B A J K Q U N B I H L N B A N A Q O A G S A K P O Q S D M
U Y V B A E I P M A O B N H R Y W I R T W D N D M U Y V I O P
J H A K S X C V U M N P O T L P Q N S J A N D D Q Q P S A E U
A N C X B N M K L P A Q B I F S J N O O H K I O P L T A B N X
U Y O A S X N K I W H U M O P Q S J L I K P Q A P I U Y R T B
M A K I C V Z O P Q S M U L T I P I C A T I O N N X Z M E U O
A S W T S I S M N A S K Q L A I O Q P P I I N C N M Q E S T I
J O Q L S I A M E O J S N S P J I W E K M N T M Q O Q S M A N
O I W S A M V D L S E O U I O P Q S J U I A O D M Y U T Q S C
A M E W A E O I H U W L B N M B W J I U K L A S A V C B M A B
O Q W S A F S W D I O P W E T R E X P R E S S I O N I Q Y T A

BOUNDING
EQUATION
SQUARE
ALGEBRA
COEFFICIENT
DEGREE
ZERO
CONSTANT
VARIABLE
POLYNOMIAL
LINEAR
QUADRATIC
CUBIC
TRINOMIAL
BINOMIAL
MONOMIAL
ROOT
VIETA
BEZOUT
ADDITION
SUBTRACTION
MULTIPLICATION
DIVISION
KHAWARIZMI
LOGARITHMS
DISTINCT
EXPRESSION
```

Gambar 3.7.1 Input Test Case 7


```

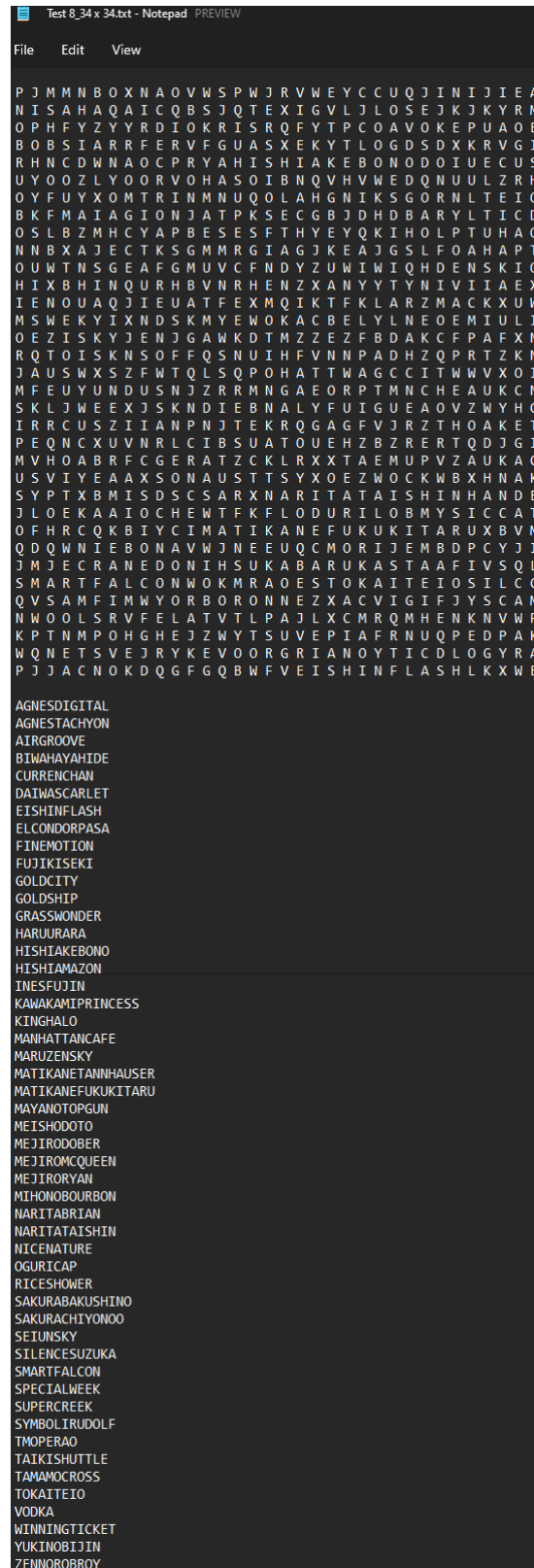
CAEA EVTMMQGEMKOUSKAPLKPOQPOMASV
ONOI TAUQEURLASQEAIOTNANJLOPQSAM
ETERTRPANACITIMMMJKLASWVCZHGVBSSE
FRBDMI OI LDUNQMAHSQAWEBK EWCSXANM
FIBLNALLARB EAZAOTUIIOANC SFQCSFKL
INEENBYAIIAIAPI MPKIMPTJWKLWEUOPU
COTQOLNIMTCRORQAUNRU YQZXBOPQSSSR
IMND A EOMOIADS AV CZXQA IANKSQ BAWET
EIAERAMONCOIOWPOQSU EGHJGWP OCFSSF
NATGBTINORONS ANBVCSQE OMAVA UDEFS
TLSREOAIMRREAHPOQEJSABLJIPNQIUO
BUNEGOLBEBNALKJCTKLQM GHOPADIGPK
FROELRMZFB LIYTBU NBEXSPRESS IONOJ
ATVF AEOLTAGMEA OATEI VOPKAPWNXEYH
UNBIBSQUAR EKA ZDLPMXYN LOGAS GBOIG
MJKIEOPQUNQA EXGDYQA OPCUBTQSO PUS
IUYTH TJKULUBLOIQIU ILZUCNANAUMLE
OIUBNM AHYVAOCXZPATPOIBUMWQS QKLW
PKIHKNBQENTPLMNB CAISRBA STZXCVBQ
NMB AJKQUNBIHLNB ANAQOAGSAKPOQS DM
UYVB AEIPMA OBNHRYWIRTDNDMU YVIOP
JHAKSXCVUMNPOTLPQNSJANDDQQP SAEU
ANCXBNMKLPAQBIF SJNOOHKIOPLTABNX
UYOASXNKIWHUMOPQSJLIK PQAPIUYRTB
MAKICVZOPQSMULTIPI CATIONNXZMEUO
ASWT SISMNASKQLAIOQPP IINCNMQESTI
JOQLS IAMEOJSNSPJ IWEKMNTMQOQSMAN
OIWSAMVDLSEOU IOPQSJU IAODMYUTQSC
AMEWAE OIHUWLB NMBWJIUKLASAVCBMAB
OQWSA FSWDIOPWETREXPRESSIONIQYTA

```

Waktu Eksekusi Program : 87.000000 ms
Jumlah Total pengecekan Huruf : 25789

Gambar 3.7.2 Output Test Case 7

3.8 Test Case 8 (ukuran *large*)



Test_8_34 x 34.txt - Notepad PREVIEW

File Edit View

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

AGNESDIGITAL
AGNESTACHYON
AIRGROOVE
BIWAHAYAHIDE
CURRENCHAN
DAIWASCARLET
EISHINFLASH
ELCONDORPASA
FINEMOTION
FUJIKISEKI
GOLD CITY
GOLD SHIP
GRASSWONDER
HARUURARA
HISHIAKEBONO
HISHIAMAZON
INESFUJIN
KAWAKAMIPRINCESS
KINGHALO
MANHATTANCAFE
MARUZENSKY
MATIKANETANNHAUSER
MATIKANEFUKUKITARU
MAYANOTOPGUN
MEISHODOTO
MEJIRODOBER
MEJIROMCQUEEN
MEJIROYAN
MIHONOBOURBON
NARITABRIAN
NARITATAISHIN
NICENATURE
OGURICAP
RICESHOWER
SAKURABAKUSHINO
SAKURACHIYONOO
SEIUNSKY
SILENCESUZUKA
SMARTFALCON
SPECIALWEEK
SUPERCREEK
SYMBOLIRUDOLF
TMOPERA
TAIKISHUTTLE
TAMAMOCROSS
TOKAITEIO
VODKA
WINNINGTICKET
YUKINOBIJIN
ZENNOROBROY

Gambar 3.8.1 Input Test Case 8

```

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

Waktu Eksekusi Program : 248.000000 ms
Jumlah Total pengecekan Huruf : 59571

```

Gambar 3.8.2 Output Test Case 8

Lampiran

Lampiran 1

Checklist penilaian :

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

Lampiran 2

Link *Repository* github : https://github.com/gilanglahat22/Tucil1_13520137