

# **Laporan Tugas Kecil IF2211**

## **Strategi Algoritma**

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



oleh  
**Febryola Kurnia Putri**  
**13520140**  
**K02**

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

## **Daftar Isi**

DAFTAR ISI	1
BAB I : Algoritma Brute Force	2
BAB II : Source Code	3
BAB III : Test Case	19
BAB IV : Lampiran	39

## **BAB I**

### **Algoritma Brute Force**

Berikut penjelasan mengenai algoritma brute force yang dilakukan pada program word search puzzle ini:

1. Dari Operan input file yang ada, puzzle dan word dipisahkan 1 baris, program akan dimulai dengan membaca puzzle dan word.
2. puzzle dibaca dengan menghitung banyak baris dan kolom yang akan digunakan pada perhitungan pencarian
3. dilakukan pembacaan word pertama setelah baris kosong, karakter pertama pada kata pertama di cek pada puzzle
4. pengecekan dilakukan mulai matriks puzzle baris 0 kolom 0, jika sesuai maka akan dilakukan pengecekan selanjutnya
5. pengecekan dilakukan dengan berbagai arah, yaitu kanan, kiri, kiri atas kanan bawah, atas, bawah, kiri bawah kanan atas
6. program hanya dibuat mengecek 4 arah diatas dan sisa arahnya dilakukan dengan menggunakan pembalikan atau pencerminan untuk mengurangi fungsi/prosedur yang diperlukan.
7. pengecekan karakter selanjutnya dimulai dari arah kanan, kiri, kiri atas kanan bawah, atas, bawah, kiri bawah kanan atas secara berurutan
8. jika karakter kedua sesuai pada arah tersebut, maka pencarian hingga karakter word akhir akan dilakukan pada arah yang sama.
9. Jika ada karakter yang tidak sesuai, maka program akan melanjutkan ke arah lainnya yang sesuai dan akan berhenti jika karakter sudah ditemukan semua pada pencarian, dan akan disimpan lalu melanjutkan pencarian hingga pada akhir puzzle.
10. Jika pengecekan sudah dilakukan semua arah dan sudah tidak ditemukan kata yang sesuai, maka pencarian dilanjutkan kembali mulai dari karakter pertama.
11. Ulangi kembali langkah 4 sampai dengan 10 sampai semua puzzle selesai dicek
12. algoritma yang digunakan adalah algoritma exhaustive search karena jika suatu kata sudah ditemukan pada puzzle, program akan tetap mencari sampai ditemukan semua kemungkinan hingga akhir puzzle. Jadi memungkinkan program menghasilkan lebih dari 1 output jika kata yang ingin dicari pada puzzle lebih dari 1.
13. program akan berakhir jika semua kata sudah ditemukan dan semua karakter pada puzzle sudah dilakukan pengecekan

## BAB II

### Source Code

Program dibuat dengan menggunakan bahasa c dengan menggunakan konsep ADT untuk penginputan kata dan pembacaan list kata yang terdapat di dalam file dan Source Codenya dibagi menjadi 2 folder dan 1 file utama, yaitu:

1. Folder ADT\_HEADER

berisikan header dari ADT yang terdiri atas file sebagai berikut:

- bool.h
- listpos.h
- listpoint.h
- mesinkata.h
- mesinkarakter.h

2. Folder ADT\_FILEC

berisikan implementasi dari ADT header dan algoritma pencarian:

- listpos.c
- listpoint.c
- mesinkata.c
- mesinkarakter.c
- pencarian.c

File ini berisikan fungsi dan prosedur yang diperlukan untuk algoritma pencarian, di mana di dalamnya terdapat prosedur yang menampilkan hasil pencarian dan algoritma pencarian bruteforcenya.

3. file BRUTEFORCE.c

Ini merupakan file utama dari program yang akan mengeksekusi seluruh kode dan menghasilkan keluaran yang sesuai dengan spesifikasi. Secara lengkapnya berikut source code dari program search word puzzle dengan bahasa c ini:

#### ADT\_HEADER

1. bool.h

```
#ifndef _bool_h
#define _bool_h

#define bool unsigned char
#define true 1
#define false 0

#endif
```

2. listpos.h

```

#include "bool.h"
#ifndef LISTPOS_H
#define LISTPOS_H
#define KAPASITAS 1000
#define IDX_UNDEF -1
#define VAL_UNDEF 0
#define ELMT(l, i) (l).contents[(i)]

typedef char* ElType; /* type elemen List */
typedef struct {
    ElType contents[KAPASITAS]; /* memori tempat penyimpan elemen (container) */
} ListPos;

void CreateListPos(ListPos *l);
/* I.S. l sembarang */
/* F.S. Terbentuk List l kosong dengan kapasitas KAPASITAS */
/* Proses: Inisialisasi semua elemen List l dengan VAL_UNDEF */

int length(ListPos l);
/* Mengirimkan banyaknya elemen efektif List */
/* Mengirimkan nol jika List kosong */

bool isIdxValid(ListPos l, int i);
/* Mengirimkan true jika i adalah indeks yang valid utk kapasitas List l */
/* yaitu antara indeks yang terdefinisi utk container*/

bool isIdxEff(ListPos l, int i);
/* Mengirimkan true jika i adalah indeks yang terdefinisi utk List l */
/* yaitu antara 0..length(l)-1 */

bool isEmpty(ListPos l);
/* Mengirimkan true jika List l kosong, mengirimkan false jika tidak */

bool isFull(ListPos l);
/* Mengirimkan true jika List l penuh, mengirimkan false jika tidak */

void readList(ListPos *l);
/* I.S. l sembarang */
/* F.S. List l terdefinisi */
/* Proses: membaca banyaknya elemen l dan mengisi nilainya */
/* 1. Baca banyaknya elemen diakhiri enter, misalnya n */
/* Pembacaan diulangi sampai didapat n yang benar yaitu 0 <= n <= KAPASITAS */
/* Jika n tidak valid, tidak diberikan pesan kesalahan */
/* 2. Jika 0 < n <= KAPASITAS; Lakukan n kali:
   | | | Baca elemen mulai dari indeks 0 satu per satu diakhiri enter */
/* Jika n = 0; hanya terbentuk List kosong */

void displayList(ListPos l);
/* Proses : Menuliskan isi List dengan traversal, List ditulis di antara kurung()
   siku; antara dua elemen dipisahkan dengan separator "koma", tanpa tambahan()
   karakter di depan, di tengah, atau di belakang, termasuk spasi dan enter */
/* I.S. l boleh kosong */
/* F.S. Jika l tidak kosong: [e1,e2,...,en] */
/* Contoh : jika ada tiga elemen bernilai 1, 20, 30 akan dicetak: [1,20,30] */
/* Jika List kosong : menulis [] */

ListPos plusMinusTab(ListPos l1, ListPos l2, bool plus);
/* Prekondisi : l1 dan l2 berukuran sama dan tidak kosong */
/* Jika plus = true, mengirimkan l1+l2, yaitu setiap elemen l1 dan l2 pada()
   | | | indeks yang sama dijumlahkan */

```

### 3. listpoint.h

```

#include "bool.h"
#ifndef LISTPOINT_H
#define LISTPOINT_H
#define KAPASITI 1000
#define IDX_UNDEF -1
#define VALUE_UNDEF -999
#define ELMTP(l, i) (l).contents[(i)]
#define ELMT_X(l, i) (l).contents[(i)].x
#define ELMT_Y(l, i) (l).contents[(i)].y

typedef struct {
    int x; /* MENYIMPAN NILAI X*/
    int y; /* MENYIMPAN NILAI Y*/
} Titik;

typedef Titik EltypePoint;
typedef struct {
    Titik contents[KAPASITI];
} ListPoint;

void CreateListPoint(ListPoint *l);
/* I.S. l sembarang */
/* F.S. Terbentuk List l kosong dengan kapasitas KAPASITI */
/* Proses: Inisialisasi semua elemen List l dengan VALUE_UNDEF */

void CreatePoint(Titik *P, int x, int y);
/* Membuat titik dari komponen x dan y*/

Titik buatTitik(int x,int y);
/* mengembalikan titik yang berkomponenkan x dan y */

int panjangTitik(ListPoint l);
/* Mengirimkan banyaknya elemen efektif List */
/* Mengirimkan nol jika List kosong */

void insertLastPoint(ListPoint *l, EltypePoint val);
/* Proses: Menambahkan val sebagai elemen terakhir List */
/* I.S. List l boleh kosong, tetapi tidak penuh */
/* F.S. val adalah elemen terakhir l yang baru */
/* ***** MENGHAPUS ELEMEN ***** */

bool dalamList (ListPoint l, int x, int y);
/*mengembalikan true jika titik berada di dalam list*/

#endif

```

#### 4. mesinkarakter.h

```

#include "bool.h"
#ifndef CHAR_ENGINE_H
#define CHAR_ENGINE_H
#define MARK '.'

extern char currentKarakter;
extern bool eot;
bool fileDitemukan;

void start();
/* Mesin siap dioperasikan. Pita disiapkan untuk dibaca.
   Karakter pertama yang ada pada pita posisinya adalah pada jendela.
   I.S. : sembarang
   F.S. : currentKarakter adalah karakter pertama pada pita
      |   Jika currentKarakter != MARK maka EOP akan padam (false)
      |   Jika currentKarakter = MARK maka EOP akan menyala (true) */

void adv();
/* Pita dimajukan satu karakter.
   I.S. : Karakter pada jendela = currentKarakter, currentKarakter != MARK
   F.S. : currentKarakter adalah karakter berikutnya dari currentKarakter yang lama,
      |   currentKarakter mungkin = MARK
      |   Jika currentKarakter = MARK maka EOP akan menyala (true) */

#endif

```

## 5. mesinkata.h

```

#include "bool.h"
#include "mesinkarakter.h"
#ifndef WORD_ENGINE_H
#define WORD_ENGINE_H
#define CAPACITY 100
#define BLANK ' '
#define NEWLINE '\n'

typedef struct {
    char contents[CAPACITY]; /* container penyimpan kata, indeks yang dipakai [0..CAPACITY-1] */
    int length;
} Kata;

extern bool akhirKata;
extern Kata currentKata;

void ignoreBlank();
/* Mengabaikan satu atau beberapa BLANK
   I.S. : currentChar sembarang
   F.S. : currentChar ≠ BLANK atau currentChar = MARK */

void skipNewline();
/* melakukan skip jika terdapat newline*/

void startWord(char *filename);
/* I.S. : currentChar sembarang
   F.S. : akhirKata = true, dan currentChar = MARK;
      |   atau akhirKata = false, currentKata adalah kata yang sudah diakuisisi,
      |   currentChar karakter pertama sesudah karakter terakhir kata */

```

```

void advWord();
/* T S : currentChar adalah karakter pertama kata yang akan diakuisisi
   ebug (Ctrl+Shift+D) ntKata adalah kata terakhir yang sudah diakuisisi,
   | | | currentChar adalah karakter pertama dari kata berikutnya, mungkin MARK
   | | | Jika currentChar = MARK, akhirKata = true.
   | | | Proses : Akuisisi kata menggunakan procedure copyWord */

void copyWord();
/* Mengakuisisi kata, menyimpan dalam currentKata
   I.S. : currentChar adalah karakter pertama dari kata
   F.S. : currentKata berisi kata yang sudah diakuisisi;
   | | | currentChar = BLANK atau currentChar = MARK;
   | | | currentChar adalah karakter sesudah karakter terakhir yang diakuisisi.
   | | | Jika panjang kata melebihi CAPACITY, maka sisa kata terpotong */

char KataToChar (Kata K);
/* mengembalikan karakter dimana kata diubah menjadi karakter*/

char* KataToString (Kata K);
/* mengembalikan karakter dimana kata diubah menjadi string*/

#endif

```

## ADT\_FILEC

### 1. listpos.c

```

#include <string.h>
#include <stdio.h>
#include "../ADT_HEADER/bool.h"
#include "../ADT_HEADER/listpos.h"

void CreateListPos(ListPos *l)
/* I.S. l sembarang */
/* F.S. Terbentuk List l kosong dengan kapasitas KAPASITAS */
/* Proses: Inisialisasi semua elemen List l dengan VAL_UNDEF */
{
    for (int i = 0; i < KAPASITAS; i++){
        (*l).contents[(i)] = VAL_UNDEF;
    }
}

int length(ListPos l)
/* Mengirimkan banyaknya elemen efektif List */
/* Mengirimkan nol jika List kosong */
{
    bool benar = false;
    int hitung = 0;
    int i = 0;
    while (!benar && i<KAPASITAS)
    {
        if ((l).contents[(i)] != VAL_UNDEF){
            hitung++;
            i++;
        } else {
            benar = true;
        }
    }
    return hitung;
}

```

```

        }
    }
    return hitung;
}

bool isIdxValid(ListPos l, int i)
/* Mengirimkan true jika i adalah indeks yang valid utk kapasitas List l */
/* yaitu antara indeks yang terdefinisi utk container*/
{
    return(i>=0 && i<KAPASITAS);
}

bool isIdxEff(ListPos l, int i)
/* Mengirimkan true jika i adalah indeks yang terdefinisi utk List l */
/* yaitu antara 0..length(l)-1 */
{
    return(i>=0 && i<=(length(l)-1));
}

bool isEmpty(ListPos l)
/* Mengirimkan true jika List l kosong, mengirimkan false jika tidak */
/* *** Test List penuh *** */
{
    bool kosong = true;
    int i = 0;
    while(kosong && (i < KAPASITAS)){
        if ((l).contents[(i)] != VAL_UNDEF){
            kosong = false;
        } else {
            i++;
        }
    }
    return kosong;
}

bool isFull(ListPos l)
/* Mengirimkan true jika List l penuh, mengirimkan false jika tidak */
{
    bool penuh = true;
    int i = 0;
    while (penuh && (i < KAPASITAS))
    {
        if ((l).contents[(i)] == VAL_UNDEF){
            penuh = false;
        } else {
            i++;
        }
    }
    return penuh;
}

void displayList(ListPos l)
/* Proses : Menuliskan isi List dengan traversal, List ditulis di antara kurung siku; antara dua elemen dipisahkan dengan separator "koma", tanpa tambahan karakter di depan, di tengah, atau di belakang, termasuk spasi dan enter */
/* I.S. l boleh kosong */
/* F.S. Jika l tidak kosong: [e1,e2,...,en] */
/* Contoh : jika ada tiga elemen bernilai 1, 20, 30 akan dicetak: [1,20,30] */
/* Jika List kosong : menulis [] */
{

```

```

int i;
if (isEmpty(l)) {
    printf("\n");
} else {
    for (i = 0; i < length(l); i++){
        if(i == length(l)-1){
            printf("%s", (l).contents[(i)]);
        } else {
            printf("%s, ,(l).contents[(i)]);
        }
    }
    printf("\n");
}
}

bool isEqual(ListPos l1, ListPos l2)
/* Mengirimkan true jika l1 sama dengan l2 yaitu jika ukuran l1 = l2 dan semua
   elemennya sama */
{
    int i = 0;
    bool sama = true;
    if (length(l1) == length(l2)){
        while (sama && i <length(l1))
        {
            if (l1.contents[i] != l2.contents[i]) {
                sama = false;
            }
            else {
                i++;
            }
        }
    } else {
        sama = false;
    }
    return sama;
}

int indexOf(ListPos l, ElType val)
/* Search apakah ada elemen List l yang bernilai val */
/* Jika ada, menghasilkan indeks i terkecil, dengan ELMT(l,i) = val */
/* Jika tidak ada atau jika l kosong, mengirimkan IDX_UNDEF */
/* Skema Searching yang digunakan bebas */
{
    bool ketemu = false;
    int i = 0;
    int idx;
    while ( !ketemu && i < length(l) ){
        if (l.contents[i] == val){
            ketemu = true;
        } else {
            i++;
        }
    }
    if (ketemu){
        idx = i;
    } else {
        idx = IDX_UNDEF;
    }
}

```

```

    return idx;
}

void insertLast(ListPos *l, ElType val)
/* Proses: Menambahkan val sebagai elemen terakhir List */
/* I.S. List l boleh kosong, tetapi tidak penuh */
/* F.S. val adalah elemen terakhir l yang baru */
{
    (*l).contents[(length(*l))] = val;
}

void deleteLast(ListPos *l, ElType *val)
/* Proses : Menghapus elemen terakhir List */
/* I.S. List tidak kosong */
/* F.S. val adalah nilai elemen terakhir l sebelum penghapusan, */
/*      Banyaknya elemen List berkurang satu */
/*      List l mungkin menjadi kosong */
{
    *val = (*l).contents[length(*l)-1];
    (*l).contents[(length(*l))-1] = VAL_UNDEF;
}

```

## 2. listpoint.c

```

#include <string.h>
#include <stdio.h>
#include "../ADT_HEADER/bool.h"
#include "../ADT_HEADER/listpoint.h"

void CreateListPoint(ListPoint *l)
/* I.S. l sembarang */
/* F.S. Terbentuk List l kosong dengan kapasitas KAPASITI */
/* Proses: Inisialisasi semua elemen List l dengan VALUE_UNDEF */
{
    for (int i = 0; i < KAPASITI; i++){
        (*l).contents[(i)].x = VALUE_UNDEF;
        (*l).contents[(i)].y = VALUE_UNDEF;
    }
}

int panjangTitik(ListPoint l){
    int i = 0;
    bool ketemu = false;
    int hitung = 0;
    while (!ketemu && i<KAPASITI)
    {
        if ((l).contents[(i)].x != VALUE_UNDEF && (l).contents[(i)].y != VALUE_UNDEF){
            hitung++;
            i++;
        } else {
            ketemu = true;
        }
    }
}

```

```

        return hitung;
    }
/* Mengirimkan banyaknya elemen efektif List */
/* Mengirimkan nol jika List kosong */

void CreatePoint(Titik *P, int a, int b){
    P->x = a;
    P->y = b;
}
/* membuat titik yang berkomponenkan a dan b*/

Titik buatTitik(int x, int y){
    Titik P;
    P.x = x;
    P.y = y;
    return P;
}
/*mengembalikan titik yang berkomponenkan a dan b*/

void insertLastPoint(ListPoint *l, EltypePoint val)
/* Proses: Menambahkan val sebagai elemen terakhir List */
/* I.S. List l boleh kosong, tetapi tidak penuh */
/* F.S. val adalah elemen terakhir l yang baru */
{
    (*l).contents[(panjangTitik(*l))] = val;
}

bool dalamList (ListPoint l, int x, int y){
    bool ketemu = false;
    int i =0;

    while (i < panjangTitik(l) && !ketemu){
        if ((l).contents[(i)].x == x && (l).contents[(i)].y == y){
            ketemu = true;
        } else {
            i++;
        }
    }
    return ketemu;
}
/*mengembalikan true jika titik berada di dalam list*/

```

### 3. mesinkarakter.c

```

#include <stdio.h>
#include "../ADT_HEADER/mesinkarakter.h"

static FILE * namaFile;
static int asign;
bool fileDitemukan;
char currentKarakter;
bool eot;

void start(char *filename) {
    /* Mesin siap dioperasikan. Pita disiapkan untuk dibaca.
     Karakter pertama yang ada pada pita posisinya adalah pada jendela.
     I.S. : sembarang
     F.S. : currentKarakter adalah karakter pertama pada pita
            | Jika currentKarakter != MARK maka EOP akan padam (false)
            | Jika currentKarakter = MARK maka EOP akan menyala (true) */

    if ((namaFile = fopen(filename,"r")) == NULL) {
        eot = true;
        fileDitemukan = false;
    } else {
        eot = false;
        fileDitemukan = true;
        adv();
    }
}

void adv() {
    /* Pita dimajukan satu karakter. ■
     I.S. : Karakter pada jendela = currentKarakter, currentKarakter != MARK
     F.S. : currentKarakter adalah karakter berikutnya dari currentKarakter yang lama,
            | currentKarakter mungkin = MARK
            | Jika currentKarakter = MARK maka EOP akan menyala (true) */
    asign = fscanf(namaFile,"%c",&currentKarakter);
    eot = (currentKarakter == MARK);
    if (eot) {
        | | fclose(namaFile);
    }
}

```

#### 4. mesinkata.c

```

#include <stdio.h>
#include <stdlib.h>
#include "../ADT_HEADER/mesinkarakter.h"
#include "../ADT_HEADER/mesinkata.h"
#include "../ADT_HEADER/bool.h"
bool akhirKata;
Kata currentKata;

void ignoreBlank()
/* Mengabaikan satu atau beberapa BLANK
 I.S. : currentKarakter sembarang
 F.S. : currentKarakter ≠ BLANK atau currentKarakter = MARK */
{
    if (currentKarakter == BLANK) {
        | adv();
    }
}

void skipNewline(){
/* Melakukan skip jika menemui newline*/
    if (currentKarakter == NEWLINE){
        | adv();
    }
}

void startWord(char *filename)
/* I.S. : currentKarakter sembarang
 F.S. : akhirKata = true, dan currentKarakter = MARK;
        atau akhirKata = false, currentKata adalah kata yang sudah diakuisisi,
        currentKarakter karakter pertama sesudah karakter terakhir kata */
{
    start(filename);
    akhirKata = false;
    if(currentKarakter == MARK) {
        akhirKata = true;
    } else {
        ignoreBlank();
        copyWord();
    }
}

void advWord()
/* I.S. : currentKarakter adalah karakter pertama kata yang akan diakuisisi
 F.S. : currentKata adalah kata terakhir yang sudah diakuisisi,
        currentKarakter adalah karakter pertama dari kata berikutnya, mungkin MARK
        Jika currentKarakter = MARK, akhirKata = true.
        Proses : Akuisisi kata menggunakan procedure copyWord */
{
    akhirKata =false;
    if (currentKarakter == MARK){
        akhirKata = true;
    } else {
        copyWord();
        ignoreBlank();
    }
}

void copyWord()
/* Mengakuisisi kata, menyimpan dalam currentKata
 I.S. : currentKarakter adalah karakter pertama dari kata
        currentKarakter ≠ MARK
 F.S. : currentKarakter = MARK
        currentKarakter karakter pertama sesudah karakter terakhir kata
        akhirKata = true
        currentKata = kata yang diakuisisi */
{
    currentKata = currentKarakter;
    while (currentKarakter != MARK) {
        | currentKarakter = nextChar();
        | currentKata += currentKarakter;
    }
}

```

```

int i = 0;
while (currentKarakter != MARK && currentKarakter != BLANK && currentKarakter != NEWLINE && i < CAPACITY){
    currentKata.contents[i] = currentKarakter;
    adv();
    i++;
}
currentKata.length = i;
}

char KataToChar (Kata K)
{
    if (K.length > 1) {
        return '@';
    } else {
        return (K.contents[0]);
    }
}
/* mengembalikan karakter dimana kata diubah menjadi karakter*/

char* KataToString (Kata K){
    char* perkata = (char*)malloc(sizeof(char)*(CAPACITY));
    int i = 0;
    for (i = 0; i < K.length; i++) {
        perkata[i] = K.contents[i];
    }
    perkata[i] = '\0';
    return perkata;
}
/* mengembalikan karakter dimana kata diubah menjadi string*/

```

## 5. pencarian.c

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include "mesinkarakter.c"
#include "mesinkata.c"
#include "../ADT_HEADER/bool.h"
#include "listpoint.c"
#include "listpos.c"

void displayHasil(ListPoint L, int baris, int kolom,char words[baris][kolom]){
    int i; int j;
    for (i = 0; i < baris; i++){
        for (j = 0; j<kolom; j++){
            if (dalamList(L, i,j)){
                printf(" %c ", words[i][j]);
            } else {
                printf(" - ");
            }
        } printf("\n");
    }
    printf("\n \n");
}

void carikeKiriAtasdanKananBawah (int count,int baris, int kolom,char words[baris][kolom], char* word, Titik P){
    ListPoint l;
    CreateListPoint(&l);
    int a = P.x; int b = P.y; int c = 0;
    if (a + strlen(word) -1 < baris && b + strlen(word) -1 < kolom){

```

```

        bool sama = true;
        while (a < (P.x + strlen(word))&& b < (P.y + strlen(word)) && sama){
            if (words[a][b] == word[c]){
                insertLastPoint(&l,buatTitik(a,b));
                a++; c++; b++;
            } else {
                sama = false;
            }
            count++;
        }
        if (sama){
            printf("Total perbandingan Huruf Yang Dilakukan: %d\n", count);
            displayHasil(l,baris,kolom,words);
        }
    }

void carikeKiriBawahdanKananAtas (int count, int baris, int kolom,char words[baris][kolom], char* word, Titik P){

ListPoint l;
CreateListPoint(&l);
int a = P.x; int b = P.y; int c = 0;
if (a - strlen(word) + 1 <= 0 && b + strlen(word) - 1 <= kolom){
    bool sama = true;
    while (a >= (P.x - strlen(word) + 1)&& b < (P.y + strlen(word)) && sama){
        if (words[a][b] == word[c]){
            insertLastPoint(&l,buatTitik(a,b));
            a--; c++; b++;
        } else {
            sama = false;
        }
        count++;
    }
    if (sama){
        printf("Total perbandingan Huruf Yang Dilakukan: %d\n", count);
        displayHasil(l,baris,kolom,words);
    }
}

void cariKiridanKanan (int count, int baris,int kolom,char words[baris][kolom], char* word, Titik P){

ListPoint l;
CreateListPoint(&l);
int a = P.x; int b = P.y; int c = 0;
if (b + strlen(word) - 1 < kolom ){
    bool sama = true;
    while (b < (P.y + strlen(word)) && sama){
        if (words[a][b] == word[c]){
            insertLastPoint(&l,buatTitik(a,b));
            b++; c++;
        } else {
            sama = false;
        }
        count++;
    }
    if (sama){
        printf("Total perbandingan Huruf Yang Dilakukan: %d\n", count);
        displayHasil(l,baris,kolom,words);
    }
}

```

```

void cariAtasdanBawah (int count, int baris, int kolom,char words[baris][kolom], char* word, Titik P){
    ListPoint l;
    CreateListPoint(&l);
    int a = P.x; int b = P.y; int c = 0;
    if (a + strlen(word) -1 < baris ){
        bool sama = true;
        while (a < (P.x + strlen(word)) && sama){
            if (words[a][b] == word[c]){
                insertLastPoint(&l,buatTitik(a,b));
                a++; c++;
            } else {
                sama = false;
            }
            count++;
        }
        if (sama){
            printf("Total perbandingan Huruf Yang Dilakukan: %d\n", count);
            displayHasil(l,baris,kolom,words);
        }
    }
}

```

## BRUTEFORCE.c

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include "ADT_HEADER/mesinkarakter.h"
#include "ADT_HEADER/mesinkata.h"
#include "ADT_HEADER/bool.h"
#include "ADT_HEADER/listpoint.h"
#include "ADT_HEADER/listpos.h"
#include "ADT_FILEC/pencarian.c"

void pengecekanKarakter (int baris, int kolom, char* word,char words[baris][kolom]){
    Titik P;
    bool cocok = false;
    int i = 0;
    int j = 0;
    int count = 0;
    for (i = 0; i < baris; i++){
        count++;
        for(j = 0; j < kolom;j++){
            count++;
            if (words[i][j] == word[0]) {
                P.x = i;
                P.y = j;
                cocok = true;
                cariKiridanKanan(count,baris,kolom,words,word,P);
                cariAtasdanBawah(count,baris,kolom,words,word,P);
                carikeKiriAtasdanKananBawah(count,baris,kolom,words,word,P);
                carikeKiriBawahdanKananAtas(count,baris,kolom,words,word,P);
            }
        }
    }
}

```

```

char* balikString(char *string)
{
    int panjang = strlen(string);
    char* balik = (char*)malloc((panjang+1) * sizeof(char));
    for(int i=0;i<panjang;i++)
    {
        balik[(panjang-1)-i]=string[i];
    }
    balik[panjang] = '\0';
    return balik;
}

int main (){
    char repeat = 'Y';
    while(repeat=='Y'){
        printf("\n-----Selamat Datang di Program Word Search Puzzle-----");
        printf(".....");
        printf("PPPPPPPPPPPPPPPPP   UUUUUUUU   UUUUUUUUZZZZZZZZZZZZZZZZZZZZZL LLLL LLLL LLLL EEEEEEEEEE EEEEEE");
        printf("P:::::P U:::::U U:::::UZ:::::Z Z:::::ZL:::::L E:::::E");
        printf("P:::::PPPPP:::::P U:::::U U:::::UZ:::::Z Z:::::ZL:::::L E:::::E");
        printf("PP:::::P P:::::PUU:::::U U:::::U ZZZZZ Z:::::Z Z:::::Z L:::::L E:::::EEEEE EEEEEE");
        printf(" P:::::P P:::::P U:::::U U:::::U ZZZZZ Z:::::Z Z:::::Z L:::::L E:::::E EEE");
        printf(" P:::::P P:::::P U:::::D D:::::U Z:::::Z Z:::::Z L:::::L E:::::E");
        printf(" P:::::PPPPP:::::P U:::::D D:::::U Z:::::Z Z:::::Z L:::::L E:::::EEEEE EEEEEE");
        printf(" P:::::PPPPPPPPP U:::::D D:::::U Z:::::Z Z:::::Z L:::::L E:::::EEEEE EEEEEE");
        printf(" P:::::P U:::::D D:::::U Z:::::Z Z:::::Z L:::::L E:::::EEEEE EEEEEE");
        printf(" P:::::P U:::::D D:::::U Z:::::Z Z:::::Z L:::::L E:::::E EEE");
        printf(" P:::::P U:::::U ZZZ:::::Z ZZZZZZZZ:::::Z ZZZZZ L:::::L LLLL E:::::E EEE");
        while (currentKarakter != NEWLINE && eot != 1 && currentKarakter != MARK) {
            hitungKarakter++;
            advWord();
            if (currentKarakter == NEWLINE) {
                skipNewline();
                baris++;
            }
        }
        kolom = hitungKarakter/baris;
        printf("jumlah baris: %d\n",baris);
        printf("jumlah kolom: %d\n",kolom);
        printf("Berikut adalah word puzzle yang akan digunakan:\n\n");
        char words[baris][kolom];
        startWord(inputanFile);
        int i = 0; int j = 0;
        while (currentKarakter != NEWLINE){
            advWord();
            words[i][j] = KataToChar(currentKata);
            if (currentKarakter == NEWLINE) {
                skipNewline();
                j = 0;
                i++;
            } else {
                j++;
            }
        }
    }
}

```

```

for (i=0; i<baris; i++)
{
    for(j=0; j<kolom; j++)
    {
        printf("%c ", words[i][j]);
    }
    printf("\n");
}
skipNewline();
advWord();
ListPos word;
ListPoint listTitik;
CreateListPoint(&listTitik);
CreateListPos(&word);
while (currentKarakter != MARK){
    hitungWord++;
    insertLast(&word, KataToString(currentKata));
    insertLast(&word, balikString(KataToString(currentKata)));
    if (currentKarakter == NEWLINE) {
        skipNewline();
    }
    advWord();
}
printf("\nJumlah Kata yang Akan Dicari: %d\n", hitungWord);
printf("\n");
printf("Berikut Proses Pencarian Setiap Kata\n\n");
clock_t mulaiWaktu = clock();
for (i = 0; i < length(word);i++){
    pengecekanKarakter(baris,kolom,word.contents[i],words);
}
clock_t akhirWaktu = clock();
float waktu = (float)(akhirWaktu - mulaiWaktu) / CLOCKS_PER_SEC;
printf("Waktu Eksekusi Program : %f s\n\n", waktu);
printf(".....\n.....\n.....Silakan mencoba kembali di lain waktu.\n.....\n");
printf("Ingin melanjutkan program? (Y/N): ");
scanf(" %c", &repeat);
}
return 0;
}

```

# BAB III

## Test Case

Output awal pembacaan file akan sama yaitu sebagai berikut:

## 1. Test Case 1 - Small

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

**ANIMAL**

**MAKAN**

**SARAPAN**

**SENANG**

**TIDUR**

**CAPEK**

**KENAPA**

**KAPAN**

**BOBO**

**KANGEN**

**BANGET**

```
Masukkan nama file (dalam format.txt): .../test/tes1.txt
jumlah baris: 15
jumlah kolom: 15
Berikut adalah word puzzle yang akan digunakan:
```

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

```
Jumlah Kata yang Akan Dicari: 11
```

```
Berikut Proses Pencarian Setiap Kata
```

```
Total perbandingan Huruf Yang Dilakukan: 91
```

```
- - - - - A - - - - -  
- - - - N  
- - - I  
- - M  
- - A  
- L
```

```
Total perbandingan Huruf Yang Dilakukan: 32
```

```
- - - - - N A K A M - - - - -
```

```
Total perbandingan Huruf Yang Dilakukan: 8
```

```
G - -  
N - -  
A - -  
N - -  
E - -  
S - -
```

```
Total perbandingan Huruf Yang Dilakukan: 174
```

```
Total perbandingan Huruf Yang Dilakukan: 10
```

```
- N - - -  
- - A - -  
- - - P - -  
- - - A - -  
- - R - -  
- - - A - -  
- - - S - -
```

```
T I D U R - - -
```

```
Total perbandingan Huruf Yang Dilakukan: 144
```

```
- - - - - C A P E K - - - - -
```

```
Total perbandingan Huruf Yang Dilakukan: 63
```

```
- K -  
- A -  
- P -  
- A -  
- N -
```

```
Total perbandingan Huruf Yang Dilakukan: 188
```

```
N E G N A K - - -
```

```
Total perbandingan Huruf Yang Dilakukan: 25
```

```
- - - - -  
- K - - -  
- - E - -  
- - - N - -  
- - - A - -  
- - P - -  
- - A - -
```

```
Total perbandingan Huruf Yang Dilakukan: 60
```

```
- - B -  
- - O -  
- - B -  
- - O -
```

```
Total perbandingan Huruf Yang Dilakukan: 48
```

```
T - -  
- E -  
- - G -  
- - N -  
- - A -  
- - B -
```

```
Waktu Eksekusi Program : 0.381000 s
```

## 2. Test Case 2 - Small

```

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

ANCONA

```

CORNISH  
MARANS  
ASEEL  
HOLLAND  
PHOENIX  
BUCKEYE  
JAVA  
SULTAN  
CAMPINE  
LEGHORN  
SUMATRA  
COCHIN  
MALAY  
SUSSEX

Masukkan nama file (dalam format.txt): ../test/tes2.txt  
jumlah baris: 15  
jumlah kolom: 15  
Berikut adalah word puzzle yang akan digunakan:

```

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

```

Jumlah Kata yang Akan Dicari: 15

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 169

- - - - - A N O C N A - - - - -

Total perbandingan Huruf Yang Dilakukan: 32

- - - - - H S I N R O C - - - - -

Total perbandingan Huruf Yang Dilakukan: 69

S - -  
N - -  
A - -  
R - -  
A - -  
M - -

Total perbandingan Huruf Yang Dilakukan: 177

L E E S A

```
Total perbandingan Huruf Yang Dilakukan: 45
- - - - -
- - - - -
- - - D -
- - - N -
- - - A -
- - - L -
- - - L -
- - - O -
- - - H -
```

Total perbandingan Huruf Yang Dilakukan: 11  
- - P H O E N I X - - - -

Total perbandingan Huruf Yang Dilakukan: 240

Total perbandingan Huruf Yang Dilakukan: 157

```
Total perbandingan Huruf Yang Dilakukan: 58
- - - - -
- - - - -
- - S - -
- - U - -
- - - L - -
- - - T - -
- - - A - -
- - - - N -
```

Total perbandingan Huruf Yang Dilakukan: 69

```
Total perbandingan Huruf Yang Dilakukan: 9  
A - - - - - - -  
R - - - - - - -  
T - - - - - - -  
A - - - - - - -  
M - - - - - - -  
U - - - - - - -  
S - - - - - - -
```

Total perbandingan Huruf Yang Dilakukan: 54

```
Total perbandingan Huruf Yang Dilakukan: 54
----- C
----- O
----- C
----- H
----- N
----- I
```

Total perbandingan Huruf Yang Dilakukan: 89

Waktu Eksekusi Program : 0.290000 s

### 3. Test Case 3 - Small

M	D	Z	E	E	D	F	Q	S	S	D	R	A	L	
P	G	E	N	C	I	B	A	F	U	T	E	R	I	A
B	R	U	B	N	N	C	R	L	J	W	C	O	S	Y
H	J	E	A	I	K	E	A	I	O	V	O	L	H	G
E	R	L	S	I	R	C	R	H	N	T	V	F	O	L
S	P	G	N	C	I	C	X	E	F	Y	E	R	N	L
U	J	G	Y	L	R	N	S	L	H	O	R	A	E	P
M	C	T	B	M	Y	I	S	N	F	D	U	L	S	P
A	I	I	A	I	X	X	B	J	I	I	A	U	T	G
V	B	S	S	E	N	T	R	E	N	I	R	G	V	U
B	Y	M	N	B	D	Q	V	V	D	D	L	E	W	O
S	I	B	R	A	I	D	W	D	A	V	O	R	X	L
V	L	X	Z	Q	J	E	I	R	L	E	S	R	R	X
B	Y	Z	L	W	W	K	N	H	H	O	P	I	N	D
P	E	R	J	U	R	Y	K	B	P	D	B	V	U	Q
ADHERENCE														
AMUSE														
BIBLICAL														
BRAID														
BRINY														
DISHONEST														
FINAL														
INERTNESS														
INSCRIBED														
IRREGULAR														
JUNE														
PERJURY														
PRESCRIBED														
RECOVER														

Masukkan nama file (dalam format.txt): .../test/tes3.txt  
jumlah baris: 15  
jumlah kolom: 15  
Berikut adalah word puzzle yang akan digunakan:

```
MDZEEEDFQSSDRADL
PGENCIBAFUTERIA
BRUBNNCRLJWCOSY
HJEAIKEAIOVOLHG
ERLSIRCRHNTVFOL
SPGNICICXEFYERNL
UJGYLRNSLHORAEP
MCTBMYISNFDULSP
AIIAIXXBJIIAUTG
VBSENTRENIRGVU
BYMNBBDQVVDDLEWO
SIBRAIDWDAVORXL
VLXZQJEIRLESRRX
BYZLWWKNHHOPIND
PERJURYKBPDDBUUQ
```

Jumlah Kata yang Akan Dicari: 14

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 14

```
- - E - - - - - - - 
- - C - - - - - - - 
- - - N - - - - - - 
- - - - E - - - - - 
- - - - R - - - - - 
- - - - - E - - - - 
- - - - - H - - - - 
- - - - - D - - - - 
- - - - - A - - - -
```

Total perbandingan Huruf Yang Dilakukan: 71

```
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - -
```

Total perbandingan Huruf Yang Dilakukan: 185

```
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - -
```

Total perbandingan Huruf Yang Dilakukan: 29

```
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - -
```

Total perbandingan Huruf Yang Dilakukan: 24

```
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - - 
- - - - - - - - - - -
```

#### 4. Test Case 4 - Medium

A S O B S C U L P T O R I E S N O X A E	CANCER
G K L I E S N O B V M J A Q U A R I U S	CAPRICORNUS
C F P H O E N I X O R I O M M I H A C F	CARINA
O A E S V K L N J I O R S R S H Y D R A	CASSIOPEIA
X S G E M I Y A T Y E R O A O D K X O S	CETUS
L T A B N L E P U S U N H S E R P E N S	CONSTELLATIONS
C Y S S A I I T S C K L I E M J I O R A	DRACO
A B U I N B D C A S S I O P E I A S I E	GEMINI
P U S G P R H F M J I O R S T A F L F O	HERCULES
R P A V I A N D R O M E D A T J T P T C	HYDRA
I B V M J I O R S R N U R B A N R A C O	LEO
C A N C E R U T R I A T A P A O T H E N	LEPUS
O A U S A G L E N O G U C T H S T A R S	LIBRA
R K L I E R V S U N S R O A F G E I N T	LYNX
N J R T U E I J J I O R S P G C L A P E	LYRA
U A P A V O L N K L I E C H G E N Y M L	NORMA
S O I T I B S U A T C A O E S A M F R L	ORION
T G S U R S E S P A F E R R D U R I A A	PAVO
E U C H G X R E G U L I T C F R O L N T	PEGASUS
T R E A O J P I E S N O I U K L I E S I	PHOENIX
A J S D C M E C L E O F T L S E P U S O	PISCES
U E D H X L N M N F O P D E X U R F T N	SAGITTARIUS
R F H I P U S T A P S G O S T A R I E S	SCORPIUS
U W O P R I O W H J E S I E S N O H G I	SCULPTOR
S A G I T T A R I U S P S C O R P I U S	SERPENS
	STARS
ANDROMEDA	TAURUS
ANTLIA	VIRGO
AQUARIUS	
ARIES	

Masukkan nama file (dalam format.txt): .../test/test4.txt  
jumlah baris: 25  
jumlah kolom: 20  
Berikut adalah word puzzle yang akan digunakan:

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

Jumlah Kata yang Akan Dicari: 32

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 205

- - - - - A N D R O M E D A - - - - -

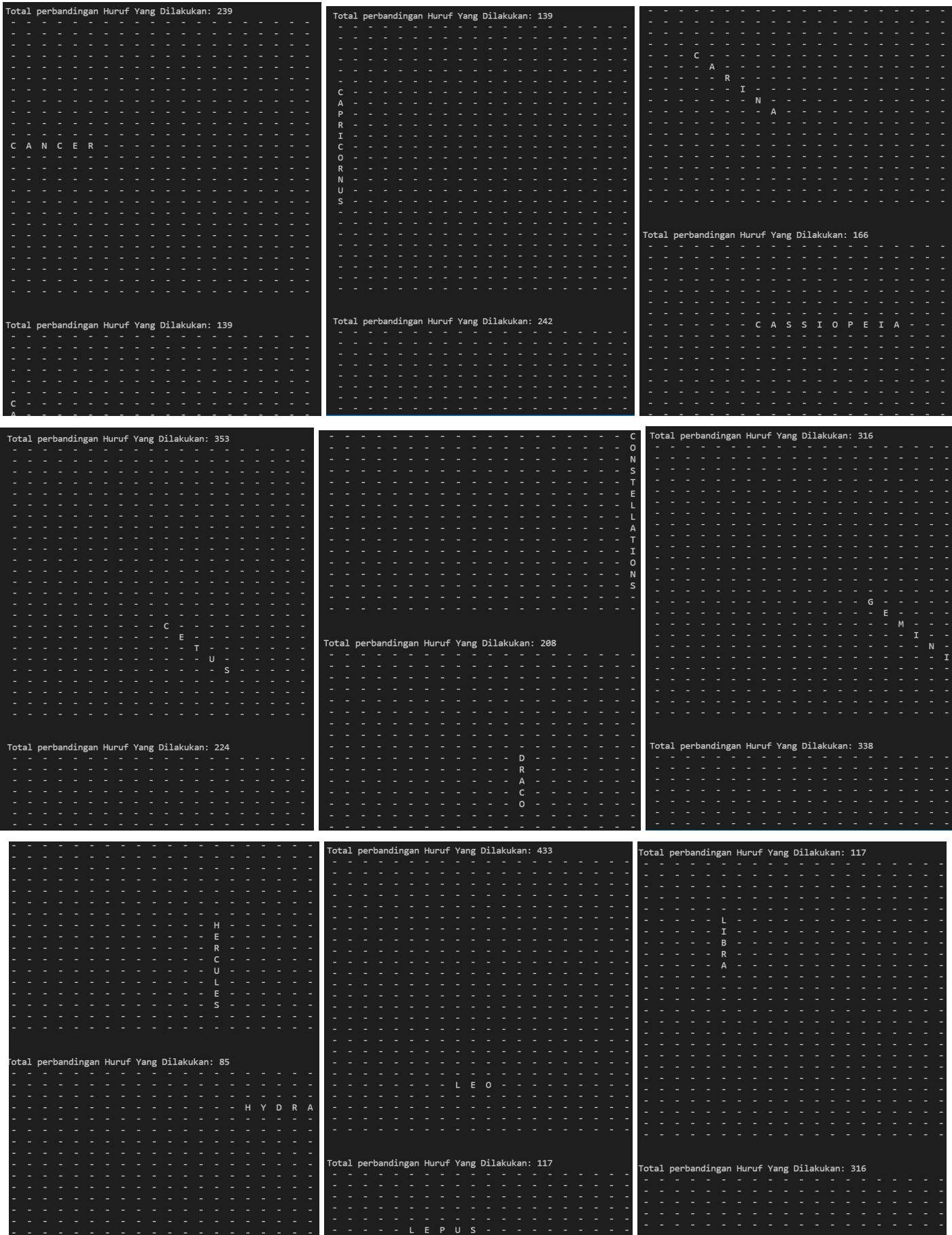
Total perbandingan Huruf Yang Dilakukan: 43

- - - - - A Q U A R I U S - - - - -

Total perbandingan Huruf Yang Dilakukan: 484

- - - - - A R I E S - - - - -

Total perbandingan Huruf Yang Dilakukan: 239



```
Total perbandingan Huruf Yang Dilakukan: 205
L - Y - R - A
```

```
Total perbandingan Huruf Yang Dilakukan: 325
P - I - S - C - E - S
```

```
S A G I T T A R I U S - - -
Total perbandingan Huruf Yang Dilakukan: 526
S C O R P I O U S
```

```
Total perbandingan Huruf Yang Dilakukan: 127
O - R - I - O - N
```

```
S E R P E N T I S
Total perbandingan Huruf Yang Dilakukan: 274
S T A R S
```

```
Total perbandingan Huruf Yang Dilakukan: 407
S C U L P T O R - - -
T A U R U S - - -
Total perbandingan Huruf Yang Dilakukan: 326
```

```
Total perbandingan Huruf Yang Dilakukan: 326
V - I - R - G - O
```

Waktu Eksekusi Program : 1.273000 s

```
.....Program selesai.....  
.....Silakan mencoba kembali di lain waktu.....
```

```
PS E:\STEI\ITB SEMESTER 4 (22 SKS) IF\IF2211 - Strategi Algoritma (3 SKS)\Tugas Kecil\loadData\src> █
```

## 5. Test Case 5 - Medium



```
Masukkan nama file (dalam format.txt): .../test/tes5.txt
jumlah baris: 22
jumlah kolom: 20
```

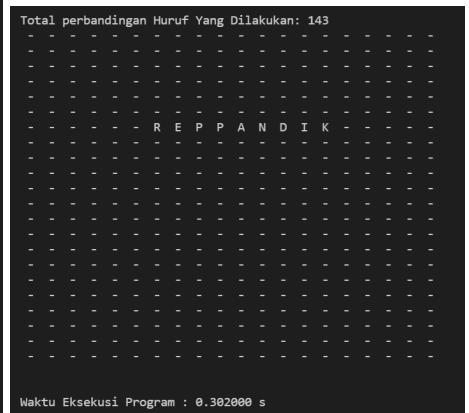
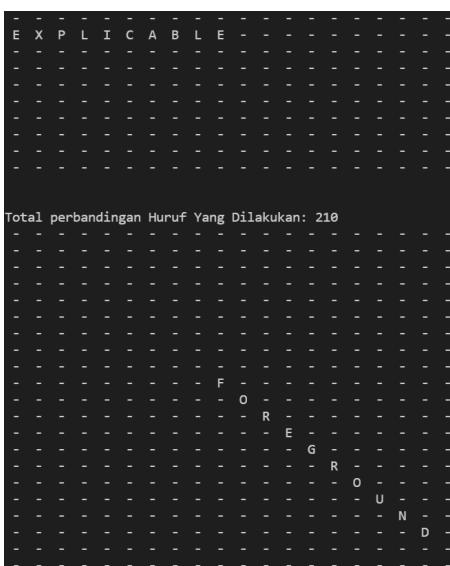
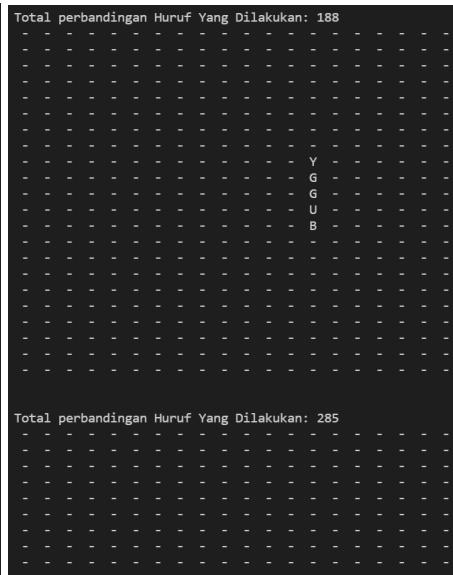
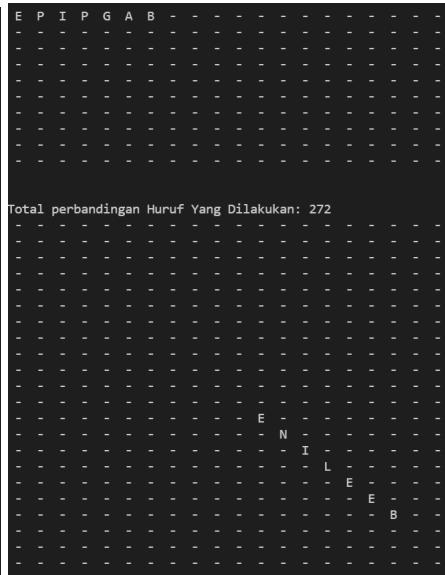
Berikut adalah word puzzle yang akan digunakan:

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

Jumlah Kata yang Akan Dicari: 10

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 261



## 6. Test Case 6 - Medium

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

SNDD  
EKKDDL  
LWKFNFN  
WKFKSPP  
CLKNF EW  
DMDSL LW  
LWSNKE  
JALANLALAL  
SMDMMEM

```
Masukkan nama file (dalam format.txt): ../../test/tes6.txt
jumlah baris: 20
jumlah kolom: 20
Berikut adalah word puzzle yang akan digunakan:
```

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

Jumlah Kata yang Akan Dicari: 10

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 156

K I M I N O

Total perbandingan Huruf Yang Dilakukan: 248

S N D D

Total perbandingan Huruf Yang Dilakukan: 223

E  
K  
D  
D  
I

```
Total perbandingan Huruf Yang Dilakukan: 143  
- - - - - - - - - - - - - - - L  
- - - - - - - - - - - - - W -  
- - - - - - - - - - - - K -  
- - - - - - - - - - - F -  
- - - - - - - - - - - N -  
- - - - - - - - - - - E -
```

Total nonbonding: 11.015; VDW: 11.015; Total: 32

- - - - -

-----

-----

W

- - - | - - - - - - - L - - - - -

Total perbandingan Huruf Yang Dilakukan: 166

-----

- - - - - J A L A N L A L A

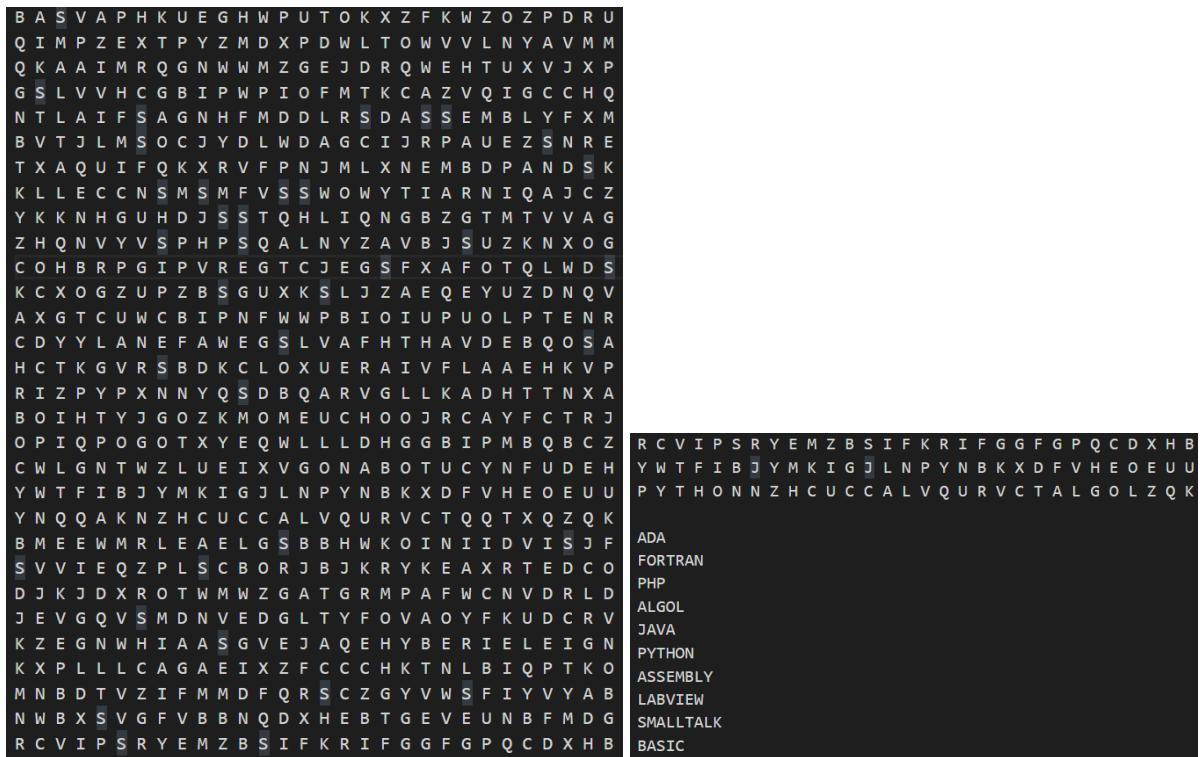
Total perbandingan Huruf Yang Dilakukan: 14

F  
N  
K  
L  
C

S M D

Waktu Eksekusi Program : 0.363000 s

## 7. Test Case 7 - Large



```
Masukkan nama file (dalam format.txt): ../test/tes7.txt
jumlah baris: 32
jumlah kolom: 30
Berikut adalah word puzzle yang akan digunakan:
```

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

Total perbandingan Huruf Yang Dilakukan: 289

F  
O  
R  
T  
R  
A  
N

Jumlah Kata yang Akan Dicari: 10

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 462

A  
D  
A

Total perbandingan Huruf Yang Dilakukan: 669

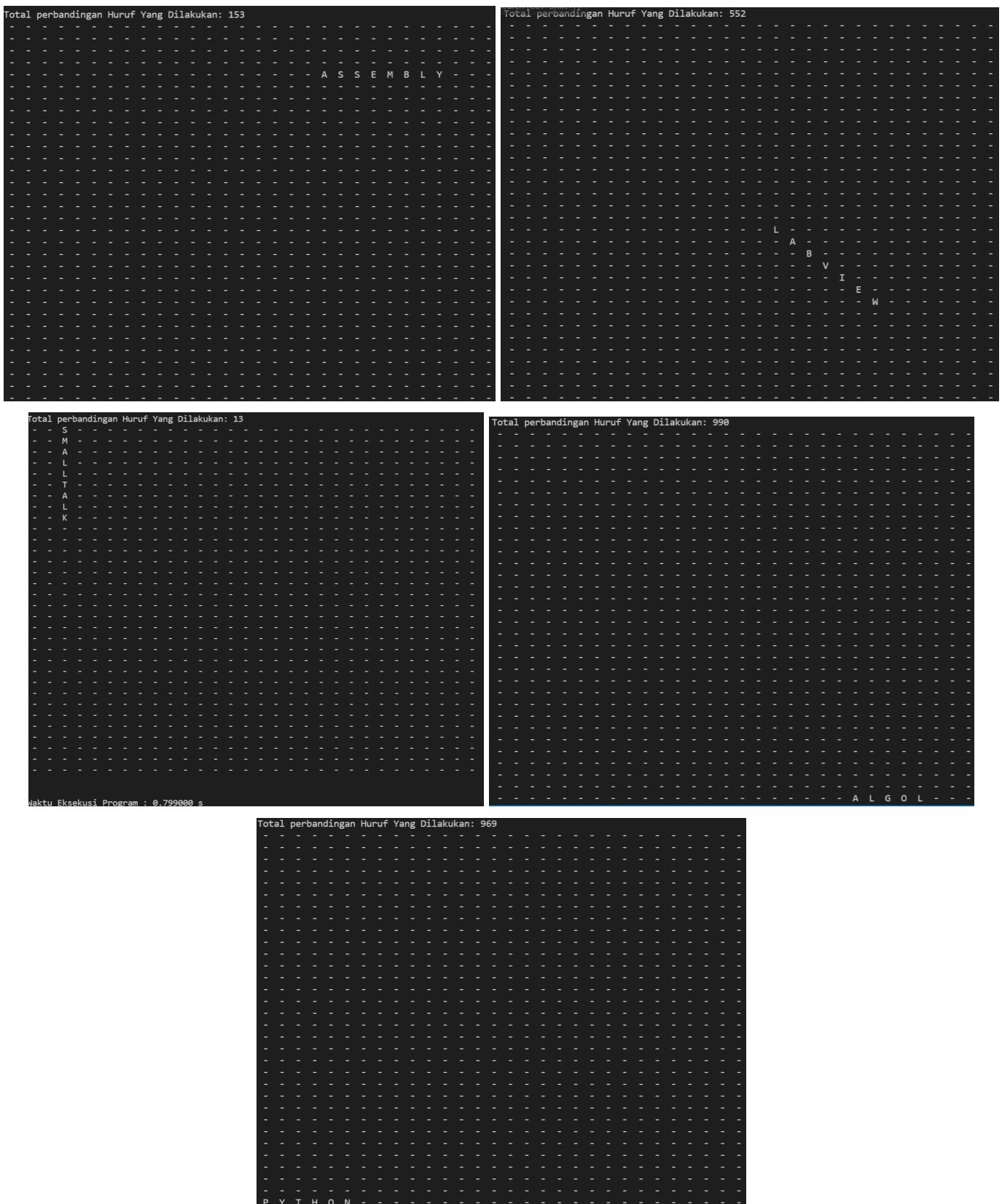
L  
O  
G  
L  
A

Total perbandingan Huruf Yang Dilakukan: 292

P H P

Total perbandingan Huruf Yang Dilakukan: 71

A  
V  
A  
J



#### 8. Test Case 8 - Large

CVLUXGELLBZZRDSWENDTIEMZGZQXIUYS  
PEQGYNLIOAYVXKRIHMZWNTNHVBMRAYIE  
XMISTACQLCALLIMJVZAIZKDUZAOSTSDZ  
CDHSTUKAGLOIRUSTAMETSLSGNYBOOPME  
SEJYFIAOKCALPYULRDIAHKUOHCYCSYN  
UBGFIKZMIYULKBFLSYGDWLKLORWCNZQ  
BKTUXHDUKKEXBFEIAPYNSPWOBRZESESYG  
JYGOEKASTIFCHMMVUFARIQXONHBSDEDTF  
WCFEBBEXLODKFFSMUEALIBROICWWSYXP  
WVKIXUFJJQPAAEWBJNUAAXEALNJMYYSG  
DNSEMYAKYBDECMTUOOMMLNXFBHAQMTU  
LOYMFRVYZHEATKANXRREAMKJHNMZIRSX  
AXKUOCKVAIMHIZRRQOPFKHNOAMYVPEYA  
GSQKRWRRLSMSDOCOSIKHZCMCKNGUHUFXN  
TZEARIXEnderJJCJBKNBXAGHTEZDMLNXA  
EPIWZSROAZRIPNXNNBEHAVNINASXEZES  
YSQOWETZKILUAFOXYSMOGUHQFCENIDVIQH  
OSOTCKRONIILARJAGJYXLFPWZPVEWRYF  
YRCROJMZMSGJGJONOZKOSQOGFUKIVPSL  
YDAHIMEARAIAKLIWSKWRZAHKMSISHKUPA  
RNKYVKLJXCPKCTHMEVZHAUNEFPAJBAZR  
ATANAKALUIOXVYWPQVAHAROYLUBAWGUE  
LLBEBYSHIONSDCZEDOJRJLNQCARAMKLOP  
MQZDWLAVAQZFPQIOSUISEIUIJHUMMTMLO  
GTLVMUPBGAKZZXXFFRTUYMZSOWYYORPT  
QATIQUEURBKJNLEGSIHIOTVGXCURDDAG  
SZPAQIUCXHFCCOCOENGULYIQUOKNBVQMS  
MRNCLGXFBAMURNBHKTGNGURAONMDH  
BIEXQIZZPGWRJFRIIXLPAAGONWAKYU  
BSAZJDADVVORUSHIABRQOBILHCBOADYCY

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

AKIROSE  
AME  
ANYA  
AQUA  
AYAME  
AZKI  
BAELS  
BOTAN  
CALLI  
CERES  
CHLOE  
CHOCO  
COCO  
FLARE  
FUBUKI  
GURA  
HAACHAMA  
INA  
IOFI  
IROHA  
IRYS  
KANATA  
KIARA  
KORONE  
KOYORI

Masukkan nama file (dalam format.txt): ..../test/tes8.txt

jumlah baris: 34

jumlah kolom: 32

Berikut adalah word puzzle yang akan digunakan:

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

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

Jumlah Kata yang Akan Dicari: 25

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 584

Total perbandingan Huruf Yang Dilakukan: 120

A M E

Total perbandingan Huruf Yang Dilakukan: 961

A Y N A

Total perbandingan Huruf Yang Dilakukan: 343

A  
Y  
A  
M  
E

Total perbandingan Huruf Yang Dilakukan: 144

A  
Z  
K  
I

Total perbandingan Huruf Yang Dilakukan: 82

C A L L I

Total perbandingan Huruf Yang Dilakukan: 881

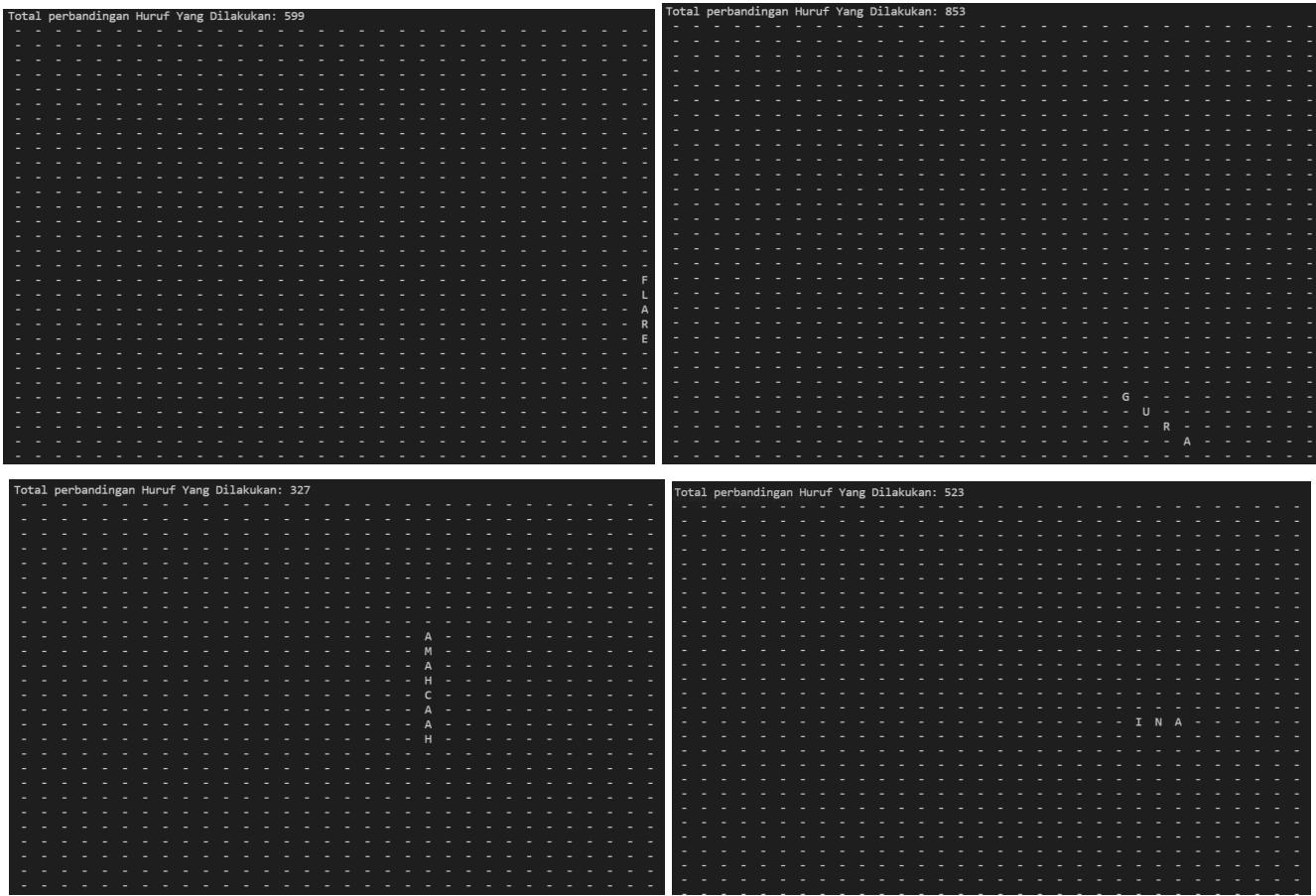
C H L O E

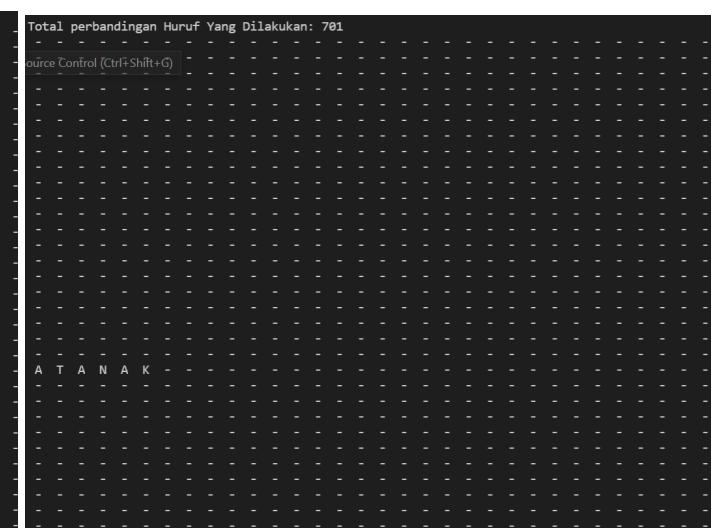
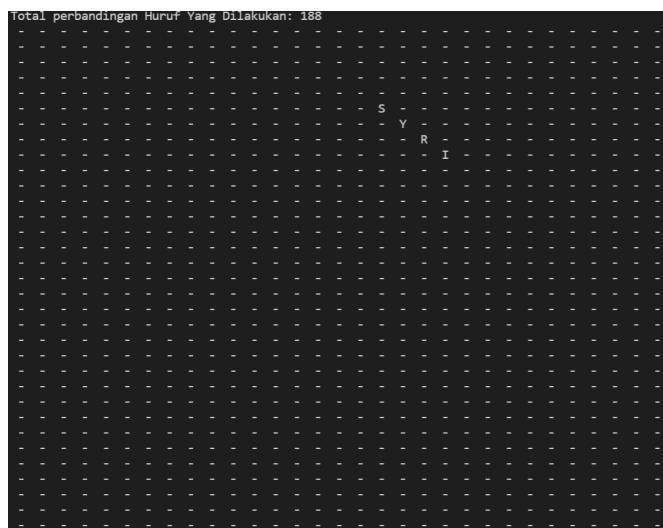
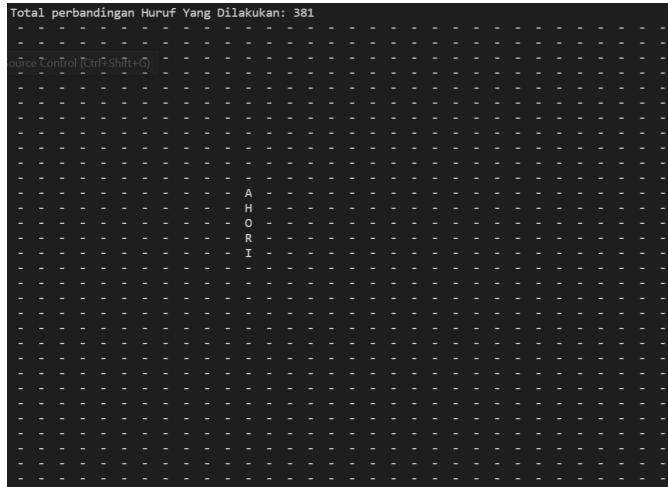
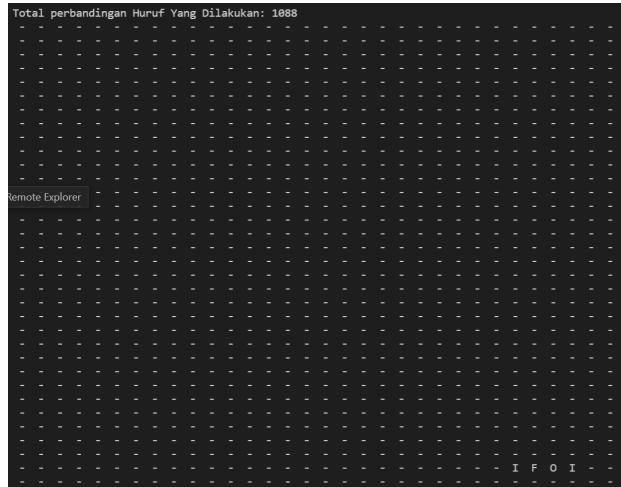
Total perbandingan Huruf Yang Dilakukan: 757

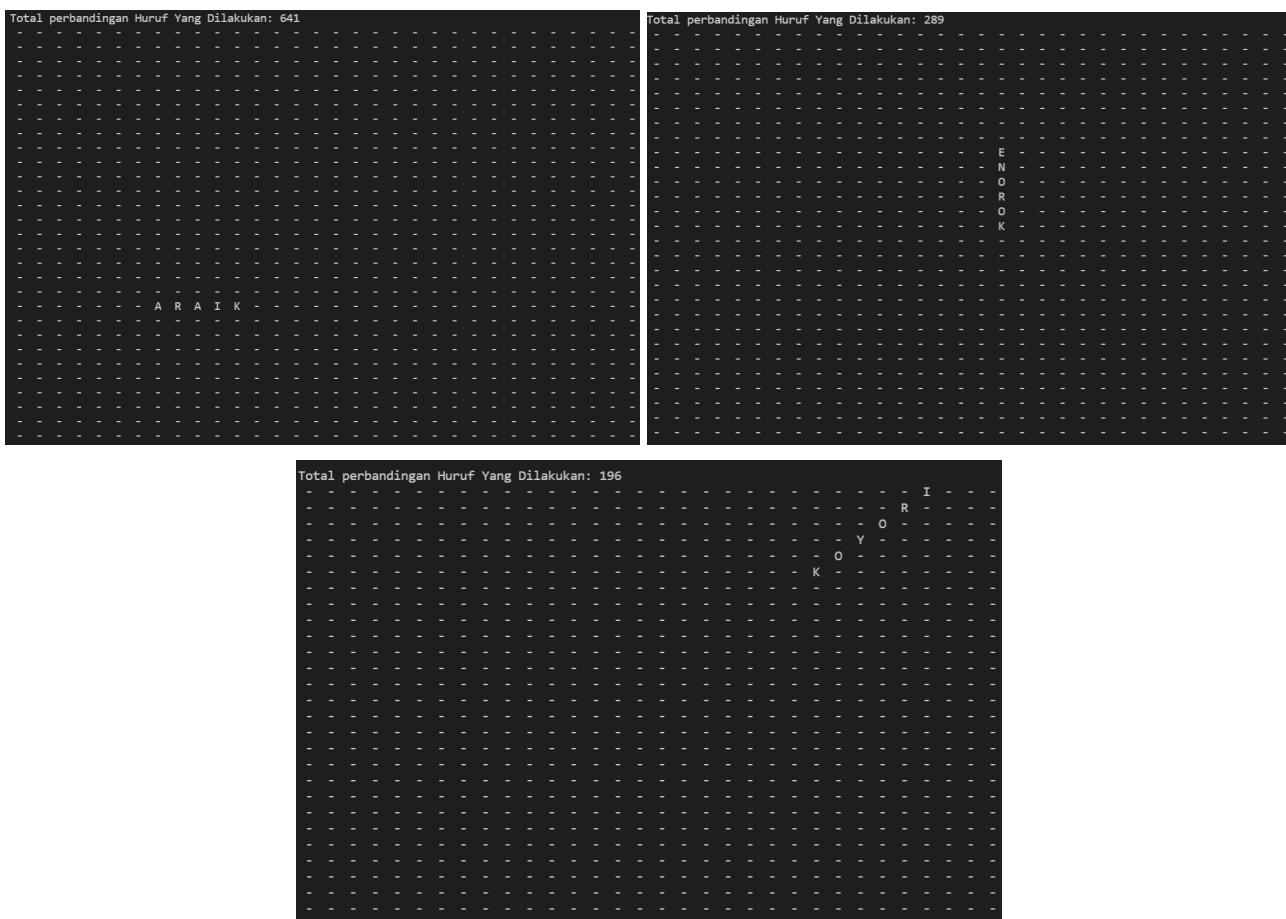
C H O C O

Total perbandingan Huruf Yang Dilakukan: 876

C O C O







### 9. Test Case 9 - Large

L F S R V T C P B W E X C C S S M A Z I U R V W F D D U S S	H R U O F N R I W U L M D V C F R P I M V E H P P U Q O L D
T L Q E S L W W L X I W E K Y F W A L E A L B Z B A N Y T U	Y M R I A T L A S U Q C L M I T W W Y V E Q M J Q T B P Z C
A W X J R X V J A R W S C H E G S T K D I Q S B E S S B P R	F E D I M J V Y K M X S F F M V W T U N I Y A T B F E F W Z
G R W H P A O Q T Y J S K S T Q A F X O E W P E X C Y G F U	E V R D P A A K E V A L W A C P G E F Y M E F Z Q H I J Z O
I P O Z G S T A O F L F S T I R E O O L P B Y A D H A R A T	X E Y X I S D Y L P M J M H U X A D Z L Q T H Z C Y O R P J
A M Q X Z W L N B E T E L G E U S E N Z A Z A K V Q I M L C	H Y N K D J V R U B S A I M V F R B L X H Z P V S H X S Z K
P T V J B L M G A C M D C A N Y X P I E D O O R G C Y M Y R	Z V O J E I X U M X D O B K G H W B X M R J I T I U P Q Z D
L F J I E L S A F M V O B T T A N J I J G O X Z A L D E R A	J E O X F C P T X A U P P Q S A E V G A M E M N I D Z W D T
P N H B P I K K Z E B U L G Q B D L K P N G S T A N F V V J	J W R X R D O Q O G E P R Z H S A F U U U A N M Q Y N W N F
T U T Y Y F U A X Y Y C Q K X W I N M J F U W G A P S T E W	M T W G J F Q R R M I V T O W X U K M O Y A U E V I P P C V
Q O C O W T L H V Z E X D W C I B D L P J U V P L A P S F V	L G H B Z X W H D W C W Y U Q S A V Z N L N R N H S I Q B H
T F V P X N F I D G I I Y E T G Q T C O J Z J E N W X P C C	Y E P B F I Q D S I D W I C A L Y Z S M R J C F M A X C H G
K K Y U I C P M N T J U P U U D A M R J T E U R A T K W P W	P U U F L P Q C E P Y A D N Q L Q N K Q O V L T W F T Y F L
K S R L E Q R K L H C P W E B Y A C I U U D A O I A R O L A	ACRUX ADHARA ALDEBARAN ALNAIR ALNILAM ALTAIR ANTARES ARCTURUS BELLATRIX BETELGEUSE
A C A Y A S G F P K E H U V S J M W N C Y G A M R B P A W Y	
A M H Z A S Z Q I I R V N Z O L D R C L I M K I W N T Q J Z	
X K S P U P T X Q U X Q I U A G W F K Q G E J V H A H J P Q	
H R U O F N R I W U L M D V C F R P I M V E H P P U Q O L D	
Y M R I A T L A S U Q C L M I T W W Y V E Q M J Q T B P Z C	

Masukkan nama file (dalam format.txt): .../test/tes9.txt  
jumlah baris: 32  
jumlah kolom: 30  
Berikut adalah word puzzle yang akan digunakan:

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

Total perbandingan Huruf Yang Dilakukan: 972

A C R U X

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

Jumlah Kata yang Akan Dicari: 10

Berikut Proses Pencarian Setiap Kata

Total perbandingan Huruf Yang Dilakukan: 155

A D H A R A

A - -  
L - -  
D - -  
E - -  
B - -  
A - -  
R - -  
A - -  
N - -

Total perbandingan Huruf Yang Dilakukan: 311

A - -  
L - -  
N - -  
A - -  
I - -  
R - -

Total perbandingan Huruf Yang Dilakukan: 953

A L N I L A M

Total perbandingan Huruf Yang Dilakukan: 568

Total perbandingan Huruf Yang Dilakukan: 11

Total perbandingan Huruf Yang Dilakukan: 39

- - - - - - - - - - - - - I - - - -

S U R U T C R  
- - - - - - - - R  
- - - - - - - - T  
- - - - - - - - A  
- - - - - L - - -  
- - - - L - - - -  
- - E - - - - - -  
- - B - - - - - -

Waktu Eksekusi Program : 0.689000 s

.....Program selesai.....  
.....Silakan mencoba kembali di lain waktu.  
Ingin melanjutkan program? (Y/N): N

## **BAB IV**

### **Lampiran**

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

**Alamat github :** <https://github.com/febryola/tucil1stima>