

Tugas Kecil 1 IF2211 Strategi Algoritma: Penyelesaian Word Search Puzzle dengan Algoritma Brute Force



Dibuat oleh:

Aditya Prawira N 13520049

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

A. Algoritma

Langkah-langkah yang dilakukan dalam penyelesaian persoalan *word search puzzle* dengan menggunakan algoritma *brute force*:

1. Buat sebuah matriks karakter yang menyimpan puzzle dari file input. Kemudian, buat sebuah list kata yang akan dicari dalam puzzle dari file input.
2. Mulai pencarian kata pertama pada list kata dengan mencocokkan huruf pertama kata yang akan dicari dengan karakter pertama pada matriks, yaitu di posisi koordinat (0,0) (pojok kiri atas).

```

A F H J K I L K K S
G H J I K R T T J S
M H N F S W G S F D

```

Gambar 1 Pencarian kata dimulai dari koordinat (0,0)

3. Jika karakter sama, pencocokan akan lanjut ke arah yang telah ditentukan. Dalam pencarian, ada 8 arah yang mungkin, yaitu ke atas, serong kanan atas, kanan, serong kanan bawah, bawah, serong kiri bawah, kiri, dan serong kiri atas.
4. Jika karakter tidak sama, ada dua kasus yang akan terjadi:
 - a. Pertama, ketika belum mencoba semua arah, maka pencarian akan berganti arah. Setelah itu, pencarian dilanjutkan.
 - b. Kedua, ketika sudah mencoba semua arah, maka akan dilanjutkan ke huruf selanjutnya dalam matriks puzzle. Ada dua kemungkinan, melanjutkan ke huruf kanannya atau ke bawah dan mulai dari kolom pertama.

```

A H J K I L K K    A H J K I L K K    A H J K I L K K
G J I K R T T J    G J I K R T T J    G J I K R T T J
M N F S A G S F → M N F S A G S F → M N F S A G S F
I A F E G T F S    I A F E G T F S    I A F E G T F S
H I K C S E A P    H I K C S E A P    H I K C S E A P

```

Gambar 2 Contoh pencarian kata “KATA” ketika karakter pertama tidak sama sehingga pencarian lanjut ke karakter selanjutnya

```

A H J K I L K K    A H J K I L K K    A H J K I L K K
G J I K R T T J    G J I K R T T J    G J I K R T T J
M N F S A G S F → M N F S A G S F → M N F S A G S F
I A F E G T F S    I A F E G T F S    I A F E G T F S
H I K C S E A P    H I K C S E A P    H I K C S E A P

                                ↓
                                A H J K I L K K
                                G J I K R T T J
                                M N F S A G S F ← M N F S A G S F
                                I A F E G T F S
                                H I K C S E A P

```

Gambar 3 Contoh pencarian kata “KATA” yang mencoba semua arah kemudian menemukan arah yang tepat

5. Jika ditemukan kecocokan kata, artinya semua karakter sudah cocok dengan kata yang dicari dengan arah tertentu, maka akan dikeluarkan output kata yang ditemukan sesuai dengan posisi kata tersebut pada puzzle.

B. Source Program

1. WordSearch.hpp

```
WordSearch.hpp > Match(Matrix, int, int, std::string, int *)
1 #ifndef WORDSEARCH_HPP
2 #define WORDSEARCH_HPP
3
4 #include <string>
5
6 #define ROW 100
7 #define COL 100
8
9 typedef struct Matrix
10 {
11     char content[ROW][COL];
12     int rowEff = 0;
13     int colEff = 0;
14 } Matrix;
15
16 typedef struct ListOfWords
17 {
18     std::string word[50];
19     int idxEff;
20 } ListOfWords;
21
22 /* *** Selektor *** */
23 #define ROWS(M) (M).rowEff
24 #define COLS(M) (M).colEff
25 #define ELMT(M, i, j) (M).content[(i)][(j)]
26 #define IDX_EFF(L) (L).idxEff
27 #define WORD_ELMT(L, i) (L).word[(i)]
28 #define WORD_ELMT_2(L, i, j) (L).word[(i)][(j)]
29
30 /****** MEMBUAT MATRIKS WORD PUZZLE *****/
31 void ReadAndCreateMatrix(Matrix *mat, ListOfWords *list, std::string filename);
32 // Nanti akan read txt files terus ngebuat matriks puzzle
33
34 /****** MEMCOEKAN KATA *****/
35 bool MatchWord(Matrix mat, int *x, int *y, int *arah, std::string word, int *countCompare);
36 // Nanti akan ngecokin kata dan kalo cocok return true
37 // x dan y bakal nyimpen kata pertamanya, arah nilainya 1 - 8, kalo 1 dia ke atas, 2 ke kanan atas, dst
38
39 bool Match1(Matrix mat, int x, int y, std::string word, int *countCompare);
40 // buat arah 1
41 bool Match2(Matrix mat, int x, int y, std::string word, int *countCompare);
42 // buat arah 2
43 bool Match3(Matrix mat, int x, int y, std::string word, int *countCompare);
44 // buat arah 3
45 bool Match4(Matrix mat, int x, int y, std::string word, int *countCompare);
46 // buat arah 4
47 bool Match5(Matrix mat, int x, int y, std::string word, int *countCompare);
48 // buat arah 5
49
50 // buat arah 6
51 bool Match6(Matrix mat, int x, int y, std::string word, int *countCompare);
52 // buat arah 7
53 bool Match7(Matrix mat, int x, int y, std::string word, int *countCompare);
54 // buat arah 8
55
56 bool NumInArray(int x, int y, int wordlength, int arr[], int arry[]);
57 // ngecek (x,y) ada di array atau tidak
58
59 /****** OUTPUT *****/
60 void PrintOutput(Matrix mat, int x, int y, int arah, int wordlength, int countCompare);
61
62 /*
63 nanti ada while loop berdasarkan panjang word, terus fungsi MatchWord itu return true atau false setelah ngecek 8 arah mata angin
64 terus nanti yg traversal di puzzle siapa? kluw awa bingung
65 */
66 #endif
```

2. WordSearch.cpp

```
WordSearch.cpp | WordSearch.hpp | main.cpp
WordSearch.cpp: Humintanyet, int, int [1, int [1]
1 #include "WordSearch.hpp"
2 #include <iostream>
3 #include <iostream>
4 #include <string>
5 #include <ctime>
6 using namespace std;
7
8 /***** MEMBUAT MATRIKS WORD PUZZLE *****/
9 void ReadAndCreateMatrix(Matrix *mat, listofWords *list, string filename) {
10 // KAMUS
11 int row, col, idx, i, j, idxlist;
12 string line;
13 filename += ".txt";
14 ifstream filePuzzle(filename);
15 bool puzzlePart = true;
16
17 // ALGORITMA
18 row = 0;
19 idxlist = 0;
20 while (getline(filePuzzle, line)) {
21 // Buat handle txt bagian puzzle
22 if (puzzlePart && line.length() != 0) {
23 idx = 0;
24 col = 0;
25 while (idx < line.length()) {
26 if (line[idx] != ' ') {
27 MAT(*mat, row, col) = line[idx];
28 col ++ 1;
29 }
30 idx ++ 1;
31 }
32 if (col > COLS(*mat)) {
33 COLS(*mat) = col;
34 }
35 row ++ 1;
36 } else if (puzzlePart == true && line.length() == 0) { // Mengubah boolean puzzlePart
37 puzzlePart = false;
38 } else if (!puzzlePart) { // Handle txt bagian word yg harus ditemukan
39 word_ELM(*list, idxlist) = line;
40 idxlist ++ 1;
41 }
42 }
43 int_EFF(*list) = idxlist; // Memastikan jumlah kata yg harus dicari
44 ROWS(*mat) = row;
45 FilePuzzle.close();
46 }
47 }
```

```
WordSearch.cpp | WordSearch.hpp | main.cpp
WordSearch.cpp: Humintanyet, int, int [1, int [1]
48 /***** MEMBACA KATA *****/
49 bool MatchWord(Matrix mat, int *x, int *y, int *arah, std::string word, int *countCompare) {
50 // KAMUS
51 bool found = false;
52 *arah = 0;
53 *x = 0;
54 *y = 0;
55 int x_cap = COLS(mat) - 1;
56 int y_cap = ROWS(mat) - 1;
57 int compawal = *countCompare;
58
59 // ALGORITMA
60 while (!found && (*y <= y_cap)) {
61 if ((*arah & 8) == 1 -- 1) { // kalo ada modulo nya + 1 - 1, dia arah 1
62 found = Match1(mat, *x, *y, word, countCompare);
63 } else if ((*arah & 8) + 1 == 2) {
64 found = Match2(mat, *x, *y, word, countCompare);
65 } else if ((*arah & 8) + 1 == 3) {
66 found = Match3(mat, *x, *y, word, countCompare);
67 } else if ((*arah & 8) + 1 == 4) {
68 found = Match4(mat, *x, *y, word, countCompare);
69 } else if ((*arah & 8) + 1 == 5) {
70 found = Match5(mat, *x, *y, word, countCompare);
71 } else if ((*arah & 8) + 1 == 6) {
72 found = Match6(mat, *x, *y, word, countCompare);
73 } else if ((*arah & 8) + 1 == 7) {
74 found = Match7(mat, *x, *y, word, countCompare);
75 } else if ((*arah & 8) + 1 == 8) {
76 found = Match8(mat, *x, *y, word, countCompare);
77 } if (found == false) {
78 *x ++ 1;
79 if (*x > x_cap) {
80 *x = 0;
81 *y ++ 1;
82 }
83 }
84 }
85 *arah ++ 1;
86 }
87 if (found) {
88 *arah -- 1;
89 }
90 cout << "Dilakukan perbandingan sebanyak " << *countCompare - compawal << " kali untuk kata di bawah ini!\n";
91 return found;
92 }
```

```

WordSearch.cpp U X WordSearch.cpp U main.cpp U
WordSearch.cpp > NumArray(int, int, int [], int [])
95 bool Match1(Matrix mat, int x, int y, std::string word, int *countCompare) {
96     // KANUS
97     bool match = true;
98     int x1, y1, idx;
99
100     // ALGORITHM
101     idx = 0;
102     x1 = x;
103     y1 = y;
104     while (match && y1 >= 0 && idx < word.length()) {
105         if (ELMH(mat, y1, x1) != word[idx]) {
106             match = false;
107         } else {
108             y1 -= 1; idx += 1;
109         }
110         *countCompare += 1;
111     }
112     if (idx < word.length()) {
113         match = false;
114     }
115     return match;
116 }
117
118 bool Match2(Matrix mat, int x, int y, std::string word, int *countCompare) {
119     // KANUS
120     bool match = true;
121     int x1, y1, idx;
122
123     // ALGORITHM
124     idx = 0;
125     x1 = x;
126     y1 = y;
127     while (match && y1 >= 0 && x1 < COLS(mat) && idx < word.length()) {
128         if (ELMH(mat, y1, x1) != word[idx]) {
129             match = false;
130         } else {
131             x1 += 1; y1 -= 1; idx += 1;
132         }
133         *countCompare += 1;
134     }
135     if (idx < word.length()) {
136         match = false;
137     }
138     return match;
139 }
140

```

```

WordSearch.cpp U X WordSearch.cpp U main.cpp U
WordSearch.cpp > NumArray(int, int, int [], int [])
141 bool Match3(Matrix mat, int x, int y, std::string word, int *countCompare) {
142     // KANUS
143     bool match = true;
144     int x1, y1, idx;
145
146     // ALGORITHM
147     idx = 0;
148     x1 = x;
149     y1 = y;
150     while (match && x1 < COLS(mat) && idx < word.length()) {
151         if (ELMH(mat, y1, x1) != word[idx]) {
152             match = false;
153         } else {
154             x1 += 1; idx += 1;
155         }
156         *countCompare += 1;
157     }
158     if (idx < word.length()) {
159         match = false;
160     }
161     return match;
162 }
163
164 bool Match4(Matrix mat, int x, int y, std::string word, int *countCompare) {
165     // KANUS
166     bool match = true;
167     int x1, y1, idx;
168
169     // ALGORITHM
170     idx = 0;
171     x1 = x;
172     y1 = y;
173     while (match && y1 < ROWS(mat) && x1 < COLS(mat) && idx < word.length()) {
174         if (ELMH(mat, y1, x1) != word[idx]) {
175             match = false;
176         } else {
177             x1 += 1; y1 += 1; idx += 1;
178         }
179         *countCompare += 1;
180     }
181     if (idx < word.length()) {
182         match = false;
183     }
184     return match;
185 }
186

```

```

WordSearch.cpp U X WordSearch.cpp U main.cpp U
WordSearch.cpp > NumArray(int, int, int [], int [])
187 }
188
189 bool Match5(Matrix mat, int x, int y, std::string word, int *countCompare) {
190     // KANUS
191     bool match = true;
192     int x1, y1, idx;
193
194     // ALGORITHM
195     idx = 0;
196     x1 = x;
197     y1 = y;
198     while (match && y1 < ROWS(mat) && idx < word.length()) {
199         if (ELMH(mat, y1, x1) != word[idx]) {
200             match = false;
201         } else {
202             y1 += 1; idx += 1;
203         }
204         *countCompare += 1;
205     }
206     if (idx < word.length()) {
207         match = false;
208     }
209     return match;
210 }
211
212 bool Match6(Matrix mat, int x, int y, std::string word, int *countCompare) {
213     // KANUS
214     bool match = true;
215     int x1, y1, idx;
216
217     // ALGORITHM
218     idx = 0;
219     x1 = x;
220     y1 = y;
221     while (match && y1 < ROWS(mat) && x1 >= 0 && idx < word.length()) {
222         if (ELMH(mat, y1, x1) != word[idx]) {
223             match = false;
224         } else {
225             x1 -= 1; y1 += 1; idx += 1;
226         }
227         *countCompare += 1;
228     }
229     if (idx < word.length()) {
230         match = false;
231     }
232     return match;
233 }
234

```


3. main.cpp

```
WordSearch.cpp U WordSearch.hpp U main.cpp X
main.cpp > main()
1 #include "WordSearch.hpp"
2 #include <iostream>
3 #include <iostream>
4 #include <chrono>
5 using namespace std;
6 using namespace std::chrono;
7
8 int main()
9 // KAMUS
10 int i, j, idx, x, y, arah, countCompare;
11 string fname;
12 listofwords list;
13 Matrix mat;
14
15 // ALGORITMA
16 countCompare = 0;
17 cout << "Masukkan nama file: ";
18 cin >> fname;
19 ReadAndCreateMatrix(&mat, &list, "../test/" + fname);
20
21 idx = 0;
22
23 uint64_t awal = duration_cast<milliseconds>(system_clock::now().time_since_epoch()).count();
24 while (idx < list.size()) {
25     x = 0; y = 0; arah = 0;
26     if (MatchWord(mat, &x, &y, &arah, WORD_LENGTH(list, idx), &countCompare)) {
27         cout << "DITEMUKAN KATA: " << WORD_LENGTH(list, idx) << " \n";
28         PrintOutput(mat, x, y, arah, WORD_LENGTH(list, idx).length(), countCompare);
29     }
30     idx++;
31 }
32 uint64_t akhir = duration_cast<milliseconds>(system_clock::now().time_since_epoch()).count();
33 cout << "Terjadi perbandingan total sebanyak " << countCompare << " kali\n";
34 cout << "Diperlukan waktu sebanyak 0." << akhir - awal << " detik\n";
35
36 return 0;
37 }
```

C. Input dan Output Program

1. Contoh 1 (ukuran 15 x 15)

Input:

```
puzzlem1.txt
1 A K P H O E N I X A M D I R L
2 R R L A Q V Q H S I N R O C U
3 T U G D D A N C S N Q Y R M C
4 A Y S N N I H G W A W O E S O
5 M B P U A V A A V T G E N N C
6 U V M A L A Y Z D W K S I A H
7 S P I W L T O W Y A S N P R I
8 M J C N O O A T B J R R M A N
9 I A W X H R P N N O W X A M O
10 G I A V O J S J H V X D C R G
11 U A N O C N A G A E L E E S A
12 P M X S N O E G S V E V L P V
13 J L W I Y L Y S Y O A R G G A
14 V M O C B R U Y B D A C P E B
15 F D J P U S T B U C K E Y E L
16
17 ANCONA
18 CORNISH
19 MARANS
20 ASEEL
21 HOLLAND
22 PHOENIX
23 BUCKEYE
24 JAVA
25 SULTAN
26 CAMPINE
27 LEGHORN
28 SUMATRA
29 COCHIN
30 MALAY
31 SUSSEX
```

Output:

```
Masukkan nama file: puzzlem1
Dilakukan perbandingan sebanyak 1419 kali untuk kata di bawah ini
DITEMUKAN KATA: ANCONA !
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- A N O C N A - - - - -
- - - - -
- - - - -
- - - - -
```

```
Dilakukan perbandingan sebanyak 243 kali untuk kata di bawah ini
DITEMUKAN KATA: CORNISH !
- - - - -
- - - - - H S I N R O C -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
```

```
Dilakukan perbandingan sebanyak 1115 kali untuk kata di bawah ini
DITEMUKAN KATA: MARANS !
- - - - -
- - - - -
- - - - -
- - - - - S -
- - - - - N -
- - - - - A -
- - - - - R -
- - - - - A -
- - - - - M -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
```


LEESA

D
N
A
L
L
O
H

PHOENIX

BUCKEYE

A diagram showing a grid of dashed lines representing a coordinate system. The horizontal axis is labeled 'J' and the vertical axis is labeled 'A'. The origin is labeled 'V'. The label 'A' is also placed near the vertical axis.

S
U
L
T
A
N

ENIPMAC

A 10x10 grid of dots. The letters are placed at the following intersections (row, column) starting from the top-left:

- N at (4, 8)
- R at (4, 7)
- O at (4, 6)
- H at (5, 5)
- G at (6, 4)
- E at (7, 3)
- L at (8, 2)

A
R
T
A
M
U
S

[illegible]

MALAY

A 10x10 grid of dashed lines. The letters are placed at the following intersections (row, column): X at (5, 8), E at (5, 7), S at (6, 6), S at (7, 5), U at (8, 4), and S at (9, 3).

Terjadi perbandingan total sebanyak 15060 kali
Diperlukan waktu sebanyak 0.684 detik

Input:

```

1  R P M J F Z X T V R R C F P I
2  T E Z D Z G Q M E K Y I U A X
3  F N M Q E Q N V O R Q B S V N
4  M I B E I P E Z E R R U I I W
5  O T J W H F O L C G A C O L B
6  T O J Q Y P T S A P V L N I U
7  K L C A K U S R I C V E Z O D
8  X L H X C W M A Y T K W N N Q
9  F I T M S E N J L R E P E E B
10 S U P X N F K S L B P D J P K
11 V G J T N D E F L E C T I O N
12 B Q R N H K L O F E L T N E G
13 J H A Z I N E S S C D Y V U Z
14 X E T A L U G E R D B Z F W X
15 M I G O I T D Q L N K K P L W
16
17 BEEPER
18 BLASPHEMER
19 CUBIC
20 CUTLERY
21 DEFLECTION
22 DEPOSITED
23 FUSION
24 GARMENT
25 GENTLEFOLK
26 GUILLOTINE
27 HAYFEVER
28 HAZINESS
29 MORAL
30 PAVILION
31 REGULATE

```

Output:

```
Masukkan nama file: puzzlem2
Dilakukan perbandingan sebanyak 1109 kali untuk kata di bawah ini
DITEMUKAN KATA: BEEPER !
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - - R E P E E B
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Dilakukan perbandingan sebanyak 1205 kali untuk kata di bawah ini
DITEMUKAN KATA: BLASPHEMER !
R - - - - -
- E - - - - -
- - M - - - - -
- - - E - - - - -
- - - - H - - - - -
- - - - - P - - - - -
- - - - - S - - - - -
- - - - - A - - - - -
- - - - - L - - - - -
- - - - - B - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Dilakukan perbandingan sebanyak 587 kali untuk kata di bawah ini
DITEMUKAN KATA: CUBIC !
- - - - - C - - - - -
- - - - - I - - - - -
- - - - - B - - - - -
- - - - - U - - - - -
- - - - - C - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Dilakukan perbandingan sebanyak 920 kali untuk kata di bawah ini
DITEMUKAN KATA: CUTLERY !
- - - - -
- - - - - Y - - - - -
- - - - - R - - - - -
- - - - - E - - - - -
- - - - - L - - - - -
- - - - - T - - - - -
- - - - - U - - - - -
- - - - - C - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
```



DEFLECTION

D
 E
 P
 O
 S
 I
 T
 E
 D

F
U
S
I
O
N

— K L O F E L T N E G —

ENITOLLIG

R
 E
 V
 E
 F
 Y
 A
 H

HAZINESS

Dilakukan perbandingan sebanyak 192 kali untuk kata di bawah ini
DITEMUKAN KATA: MORAL !

M
O
R
A
L

Dilakukan perbandingan sebanyak 123 kali untuk kata di bawah ini
DITEMUKAN KATA: PAVILION !

P
A
V
I
L
I
I
O
N

Dilakukan perbandingan sebanyak 1711 kali untuk kata di bawah ini
DITEMUKAN KATA: REGULATE !

ETALUGER

Terjadi perbandingan total sebanyak 13045 kali
Diperlukan waktu sebanyak 0.678 detik

3. Contoh 3 (ukuran 15 x 15)

Input:

```
puzzlem3.txt
1  M D Z E E D F Q S S D R A D L
2  P G E N C I B A F U T E R I A
3  B R U B N N C R L J W C O S Y
4  H J E A I K E A I O V O L H G
5  E R L S I R C R H N T V F O L
6  S P G N C I C X E F Y E R N L
7  U J G Y L R N S L H O R A E P
8  M C T B M Y I S N F D U L S P
9  A I I A I X X B J I I A U T G
10 V B S S E N T R E N I R G V U
11 B Y M N B D Q V V D D L E W O
12 S I B R A I D W D A V O R X L
13 V L X Z Q J E I R L E S R R X
14 B Y Z L W W K N H H O P I N D
15 P E R J U R Y K B P D B V U Q
16
17 ADHERENCE
18 AMUSE
19 BIBLICAL
20 BRAID
21 BRINY
22 DISHONEST
23 FINAL
24 INERTNESS
25 INSCRIBED
26 IRREGULAR
27 JUNE
28 PERJURY
29 PRESCRIBED
30 RECOVER
31 SACKING
```

Output:

DISHONEST

-	-	-	-	F	-	-	-	-	-
-	-	-	I	-	-	-	-	-	-
-	-	N	-	-	-	-	-	-	-
-	A	-	-	-	-	-	-	-	-
-	L	-	-	-	-	-	-	-	-

— S S E N T R E N I —

A 10x10 grid of dots. The letters are placed at the following intersections (row, column) starting from the top-left:

- D: (1, 1)
- E: (1, 2)
- B: (2, 3)
- I: (3, 4)
- R: (4, 5)
- C: (5, 6)
- S: (6, 7)
- N: (7, 8)
- I: (8, 9)

[illegible]

A 10x10 grid of dots. The letters J, U, N, and E are placed in the top-left corner, forming the word 'JUNE' diagonally. The grid is composed of 10 rows and 10 columns of dots.

PERJURY —

```

Dilakukan perbandingan sebanyak 136 kali untuk kata di bawah ini
DITEMUKAN KATA: PRESCRIBED !
- - - - -
P - - - - -
- R - - - - -
- - E - - - - -
- - - S - - - - -
- - - - C - - - - -
- - - - - R - - - - -
- - - - - I - - - - -
- - - - - B - - - - -
- - - - - E - - - - -
- - - - - D - - - - -
- - - - -
- - - - -
- - - - -
Dilakukan perbandingan sebanyak 101 kali untuk kata di bawah ini
DITEMUKAN KATA: RECOVER !
- - - - - R - - - - -
- - - - - E - - - - -
- - - - - C - - - - -
- - - - - O - - - - -
- - - - - V - - - - -
- - - - - E - - - - -
- - - - - R - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Dilakukan perbandingan sebanyak 79 kali untuk kata di bawah ini
DITEMUKAN KATA: SACKING !
- - - - - S - - - - -
- - - - - A - - - - -
- - - - - C - - - - -
- - - - - K - - - - -
- - - - - I - - - - -
- - - - - N - - - - -
- - - - - G - - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
- - - - -
Terjadi perbandingan total sebanyak 11722 kali
Diperlukan waktu sebanyak 0.649 detik

```

4. Contoh 4 (ukuran 22 x 20)

Input:

```

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

```

Output:

[illegible]

[illegible]

A

T -



SEE L

Terjadi perbandingan total sebanyak 167
Diperlukan waktu sebanyak 0,820 detik

Input:

```

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

```

Output:

```
Masukkan nama file: puzzleM2
Dilakukan perbandingan sebanyak 3026 kali untuk kata di bawah ini
DITEMUKAN KATA: ATTILA !
```

ALL ITA

Dilakukan perbandingan sebanyak 3089 kali untuk kata di bawah ini
DITEMUKAN KATA: BEEF !

A 20x20 grid of dots. At the bottom right, there are three labels: 'B' at the 17th column and 3rd row from the bottom, 'E' at the 18th column and 4th row from the bottom, and 'F' at the 19th column and 5th row from the bottom.

[illegible][illegible][illegible]

Dilakukan perbandingan sebanyak 111 kali untuk kata di bawah ini
DITEMUKAN KATA: DISCHARGE !

- - - - - D - - - - -
- - - - - I - - - - -
- - - - - S - - - - -
- - - - - C - - - - -
- - - - - H - - - - -
- - - - - A - - - - -
- - - - - R - - - - -
- - - - - G - - - - -
- - - - - E - - - - -

Dilakukan perbandingan sebanyak 48 kali untuk kata di bawah ini
DITEMUKAN KATA: GROUP !

- - - - G - - - - -
- - - R - - - - -
- - O - - - - -
- U - - - - -
P - - - - -

Dilakukan perbandingan sebanyak 3524 kali untuk kata di bawah ini
DITEMUKAN KATA: JUDICIALLY !

Y
L
L
A
I
C
I
D
U
J

Dilakukan perbandingan sebanyak 1104 kali untuk kata di bawah ini
DITEMUKAN KATA: MACAROOON !

MACAROON

Terjadi perbandingan total sebanyak 22304 kali
Diperlukan waktu sebanyak 0,574 detik

6. Contoh 6 (ukuran 22 x 20)

Input:

```

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

```

Output:

```
Masukkan nama file: puzzleM3
Dilakukan perbandingan sebanyak 871 kali untuk kata di bawah ini
DITEMUKAN KATA: ASPECT !
```

B

Dilakukan perbandingan sebanyak 2216 kali untuk kata di bawah ini
DITEMUKAN KATA: EXPLICABLE !



EXPLICABLE — — — — —

Dilakukan perbandingan sebanyak 1570 kali untuk kata di bawah ini
DITEMUKAN KATA: FOREGROUND !

F
O
R
E
G
R
O
U
N
D

H
U
T
C
H

I
 N
 E
 L
 E
 G
 A
 N
 T

— — — — — R E P P A N D I K — — — — —

```

1  BASVAPHKHUEGHWPUTOKXZFKWZOZPDRU
2  QIMPZEXTPYZMDXPDWLTOJRWVVLNYAVDM
3  QKAAAIMRQGNWWMZGEJDRQWEHTUXVJXP
4  GSLVHVHCGBIWPWIOFMTKKCAZVQIGCCGH
5  NTLAIFFSAGNHFMDDLRSDASSEMBLYFXM
6  BVTJLMSOCJYDLWDAGCIJRPAUEZSNRE
7  TXAQUIFQKXRVFPNJMLXNEMBDPANDSK
8  KLLCECCSMMSMFVSNOWOYTTIARNIQAJ CZ
9  YKKNHGHUHDJSSSTQHLIQNGBZGTM TVVAG
10 ZHQNQVYVVSPPHSQALNYZAVBJSUZKNXOG
11 COHBRPGIPVREGTCJEGSFXAFOTQLWDS
12 KCXOGZUPZBSGUXKSLJZAEQEYUZDNQV
13 AXGTCUWCBIPNFWWPBIOIUPUOLPTENR
14 CDYYLANEFWAGESLVAFHITHAVDEBQOSA
15 HCTKGVRSSBDKCLOXUERAIVFLAAEHKVP
16 RIZPPYXNNYQSDDBQARVGLLKADHTTNXA
17 BOIHTYJGGOZKMOEUCHOJRCAYFCTRJ
18 OPIQPOGOTXYEQWLLLDHGGGBIPMBQBCZ
19 CWLGNWTWZLUEIXVGNABOTUCYNFUDEH
20 YWTFIBJYMKIGJLNPYNBKXD FVHEOEUE
21 YNQQAQKNZHCUC C ALVQURVCTQQT XQZQK
22 BMEQEWMLRLAEALGSSBBHWKIOINIIDVISJF
23 SVVIEQZPLSCBORJBKRYKEAXRTEDCO
24 DJKJDXROTW MWZGATGRMPAFWCNVDRLD
25 JEVGGVWSMDNVEDGLTYQFHYAOYFKUDCRV
26 KZEGNWHIIAASGVEJQAQHEHYBERIEIGN
27 KXPLLLCAGAEIXZFC CCKHTNLBIQPTKO
28 MNBDTVZIFMMBDFQRSCZGVVWSFIYVYAB
29 NWBXSVGVFVBBNQDXHEBTEGEVUNBFMDG
30 RCVIPSRBYEMZBSIFKRIYFGGFGPQC DXHB
31 YWTFIBJYMKIGJLNPYNBKXD FVHEOEUE
32 YNQQAQKNZHCUC C ALVQURVCTQQT XQZQK
33
34 ADA
35 FORTRAN
36 PHP
37 ALGOL
38 JAVA
39 PYTHON
40 ASSEMBLY
41 LABVIEW
42 SMALLTALK
43 BASIC

```

Output:

```
Masukkan nama file: puzzel1
Dilakukan perbandingan sebanyak 3767 kali untuk kata di bawah ini
DITEMUKAN KATA: ADA !
```

A
D
A

Dilakukan perbandingan sebanyak 175 kali untuk kata di bawah ini
DITEMUKAN KATA: FORTRAN !

F
 O
 R
 T
 R
 A
 N

A
V
A
J

NOHT

[illegible]

```

Dilakukan perbandingan sebanyak 4427 kali untuk kata di bawah ini
DITEMUKAN KATA: LABVIEW !

```

Dilakukan perbandingan sebanyak 31 kali untuk kata di bawah ini
DITEMUKAN KATA: SMALLTALK !

-- S --
-- M --
-- A --
-- L --
-- L --
-- T --
-- A --
-- L --
-- K --

Dilakukan perbandingan sebanyak 804 kali untuk kata di bawah ini
DITEMUKAN KATA: BASIC !

-- B --
-- A --
-- S --
-- I --
-- C --

Terjadi perbandingan total sebanyak 24914 kali
Diperlukan waktu sebanyak 1,408 detik

8. Contoh 8 (ukuran 32 x 30)

Input:

```

# puzzle12.txt
1  LFSRVTCPBWEXCCSSMAZIUURVWFDDUSS
2  LQESLWXLXIWEKEYFWALEALBZBANYTU
3  AWXJRXVJXJARWSCHEGSTKDIQSBESSBPR
4  GRWHPAAQOTYJSKSTQAFXOEWEPEXCYGFU
5  IPOZGZSTAOFLFSTIREOOLPBYADHARAT
6  AMQXZWLNBETELGEUSENZAZAKVQIMLC
7  PTVJBLMGACMDCANYXPIEDOORGCMYMR
8  LFJJIELSAFMVOBTTANJJIJGOXZALDERA
9  PNHBPIKKEZEBULQGBDLKPNGSTANFVVJ
10 TUTYYFUAXYYCQKXWINMJFUWGAPSTEW
11 QOCOWNTLHVZEXDWCIBDLJPJUVPLAPSFV
12 TFVPXNFIDGIIYETGOTCOJZJENWXPCC
13 KKYUIQCPMNTJUPUWEDAMRJTTEURATKWPW
14 KSRLEQRKLHCPWEUBYACIUUDDAOIAROLA
15 ACAYASGFPKEHUVSJMWNCYGAMRBPAPWY
16 AMHZASZQJITRVNZLDRCLIMKIWNTQJZ
17 XKSUPUTXQXQIUAGVFKQGEJWVAHJJPQ
18 HRUOFNRIWULMDVCFRPIWVEHPPUQOLD
19 YMRIATLASUQCLMITWWYVEQMJJQTBPCZ
20 FEDIMJVVYKMXSFFMVWUTUNIYATBFEFWZ
21 EVRDPAAKEVALWACPGEFYMEFZQHIJZO
22 XEYXISDYLPJMHUXADZLQTHZCYORPJ
23 HYNKDJVRUBSAITMVFRLBHXZPVNSXSZK
24 ZVOJQIEIXUMXDOBKGHBWBMRJITIUQPZD
25 JEOXFCTXAUUPQSAEVGAMEMNIDZWDT
26 JWRXRDDOQOGEPRZHSAFUOYUANMQVNNCF
27 MTWGJFQRRMCIWYUQXUKMOYAUVEIIPPCF
28 LGHBZXWHDWCWYUQXSAVZNLRNHSIQBH
29 YEPBFIQDSIDWICALYZSMRJCFMACHG
30 PUUFLPQCEPYADNQLKNKQOVLTFWXYFL
31 HRUOFNRIWULMDVCFRPIWVEHPPUQOLD
32 YMRIATLASUQCLMITWWYVEQMJJQTBPCZ
33
34 ACRUX
35 ADHARA
36 ELDEBARAN
37 ALNAIR
38 ANITLAM
39 ALTAIR
40 ANTARES
41 ARCTURUS
42 BELLATRIX
43 BETELGEUSE

```

Output:

```

Masukkan nama file: puzzleL2
Dilakukan perbandingan sebanyak 3885 kali untuk kata di bawah ini
DITEMUKAN KATA: ACRUX !

```

— A D H A R A —

Dilakukan perbandingan sebanyak 2079 kali untuk kata di bawah ini
DITEMUKAN KATA: ARCTURUS !

S
 U
 R
 U
 T
 C
 R
 A

Dilakukan perbandingan sebanyak 2030 kali untuk kata di bawah ini
DITEMUKAN KATA: BELLATRIX !

BETELGEUSE

Terjadi perbandingan total sebanyak 22008 kali
Diperlukan waktu sebanyak 1,416 detik

Input:

	puzzleL3.txt
1	X R J B Y K F U D R B U K Q X M N F F M Q Z O S U P O N A C
2	D X W L S D K D Q C X J R X X J W W E B L V Z W K T U R Q B
3	L Z R Y T X S H P K F C E Y T K R H S E H J V D R X G R H F U
4	X S L B G U L B T H F O V T M R D L I N T K A Z C G I K U F
5	C G Q K H J C T A K G T P P D T E O U J L A I M D W W K Y C
6	N C U N G F N E I V T Y J W Z Z G G M Q N Z N L I S M H V L
7	Z W J H R K L S N R E I S X T K Y Z D X R P U L V A I V I
8	J J J A D K F H C D W P K X P Q Y Z T F S O R I E U X R V Q
9	L X D I C V F P U Z Z G N P B H C Q C T R I R A P O C P T H
10	G A H I H Z I E S O J W R H B E L C Y U R K W I O Q Z C P W
11	H E P X W Y F R S Y K O V A B L O Z X O M T G L J D W V A
12	I A A D F P B X R W C J U T M G G O N T O F E M L Y E B L P
13	A U J R K L T X P Y I R R H Q U H M Z T J B J W U Y Z V R F
14	W N G C F S M L O P W D J L T D B H C Q J A H X I D W R Q
15	C X T G P R W N E S Y K N Z R L T S B U L U G E R S V K H X U S
16	F A R C K S I G J G F W J J Y U P L E J U S G A U S I B O E
17	Q M O R S C Q R V P I Q Z V K A M N Y H N P E T R Y H K P L
18	R O Y S Q D L B C H W R Z D C H S I Q J J V Z C Y W H W G H
19	W S H O W C G Y Z G N Y E E U L J W M O A Z N V R E K A W A
20	P S V X T O R N G B V F K C Y A T D U O W N R D G O R X G N
21	S B H D K X E G Y O J M F X C E M W M F W S Z Z V H B Y B M
22	U V Z I L C V M W H I S Y V Y O L Z J P B A S F T D T R T I
23	T Q Z T J R C S Z J C I G E Y F G R K F C L H A B F D I X B
24	I V M N D Z R D M I A P L A C I D U S S C E M D U H E C A G O
25	L X D H Z C H A G H K Z T L H B I E U S M L D P R Z E Y Q K
26	N R M N T F L W N Y Z L B G B D D I U F G E G E A E F I A O
27	F U Y Z I I X U K W H K L J B P Y U H S K W X Y H K H N O K
28	V D B K P Y S J M N P U D Y T N I V U Z E X F V S O Q J S
29	O U C K R H E I H L X D F D M N S R D Q X N L Z N K Z P H Y
30	T Z C O O Y W B D N A C S E M M D B T H B H W B L G G E M D
31	L X D I C V F P U Z Z G N P B H C Q C T R I R A P O C P T H
32	G A H I H Z I E S O J W R H B E L C Y U R K W I O Q Z C P W
33	
34	CANOPUS
35	ELNATH
36	FOMALHAUT
37	HADAR
38	MIAPLACIDUS
39	MIMOSA
40	POLLUX
41	PROCYON
42	REGULUS
43	RIGEL

Output:

```
Masukkan nama file: puzzel3
Dilakukan perbandingan sebanyak 247 kali untuk kata di bawah ini
DITEMUKAN KATA: CANOPUS !
```

----- SUPONAC

Dilakukan perbandingan sebanyak 1943 kali untuk kata di bawah ini
DITEMUKAN KATA: ELNATH !

THE
H
T
A
N
L
E

[illegible]

[illegible]

```

Dilakukan perbandingan sebanyak 4073 kali untuk kata di bawah ini
DITEMUKAN KATA: MIMOSA !

```

```

Dilakukan perbandingan sebanyak 2199 kali untuk kata di bawah ini
DITEMUKAN KATA: POLLUX !

```

```

Dilakukan perbandingan sebanyak 2104 kali untuk kata di bawah ini
DITEMUKAN KATA: PROCYON !

```

Dilakukan perbandingan sebanyak 3760 kali untuk kata di bawah ini
DITEMUKAN KATA: REGULUS !

S U L U G E R

Dilakukan perbandingan sebanyak 4449 kali untuk kata di bawah ini
DITEMUKAN KATA: RIGEL !

L
E
G
I
R

Terjadi perbandingan total sebanyak 32601 kali
Diperlukan waktu sebanyak 1,584 detik

D. Alamat Kode Program

Program dapat diunduh dari alamat berikut:

https://drive.google.com/drive/folders/1YXChzEVX2KxA_SzcPVB8KXv_RDj65t1?usp=sharing

atau

https://github.com/Adityapnn811/Tucil1Stima_13520049

E. Tabel Penilaian

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