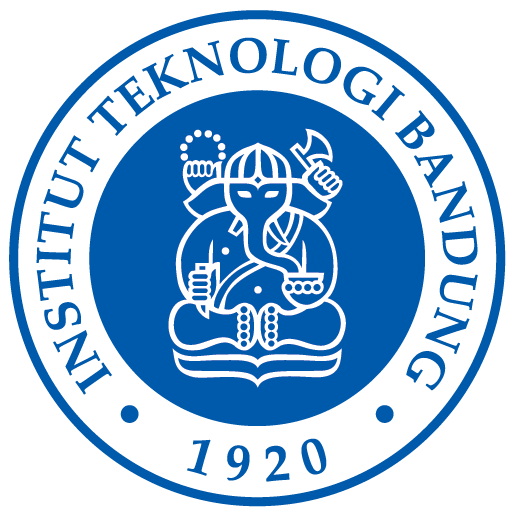
Tugas Kecil 1 IF2211 Strategi Algoritma: Penyelesaian Word Search Puzzle dengan Algoritma Brute Force



Dibuat oleh:

Aditya Prawira N 13520049

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2022

# Algoritma

Langkah-langkah yang dilakukan dalam penyelesaian persoalan *word search puzzle* dengan menggunakan algorima *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 mencocokan 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)

1. 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.
2. Jika karakter tidak sama, ada dua kasus yang akan terjadi:
   1. Pertama, ketika belum mencoba semua arah, maka pencarian akan berganti arah. Setelah itu, pencarian dilanjutkan.
   2. 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 A H J K I L K K

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

I A F E G T F S I A F E G T F S

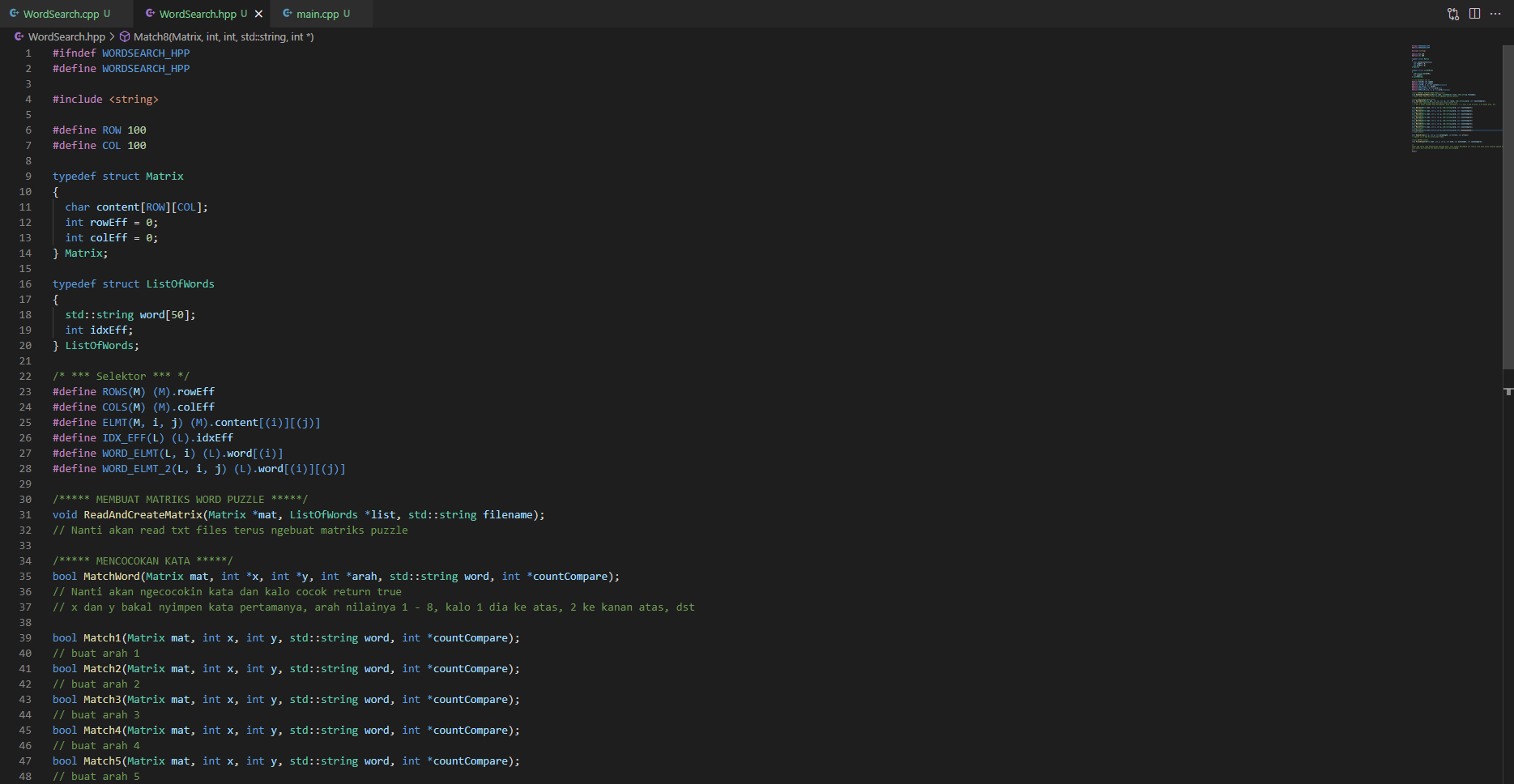
H I K C S E A P H I K C S E A P

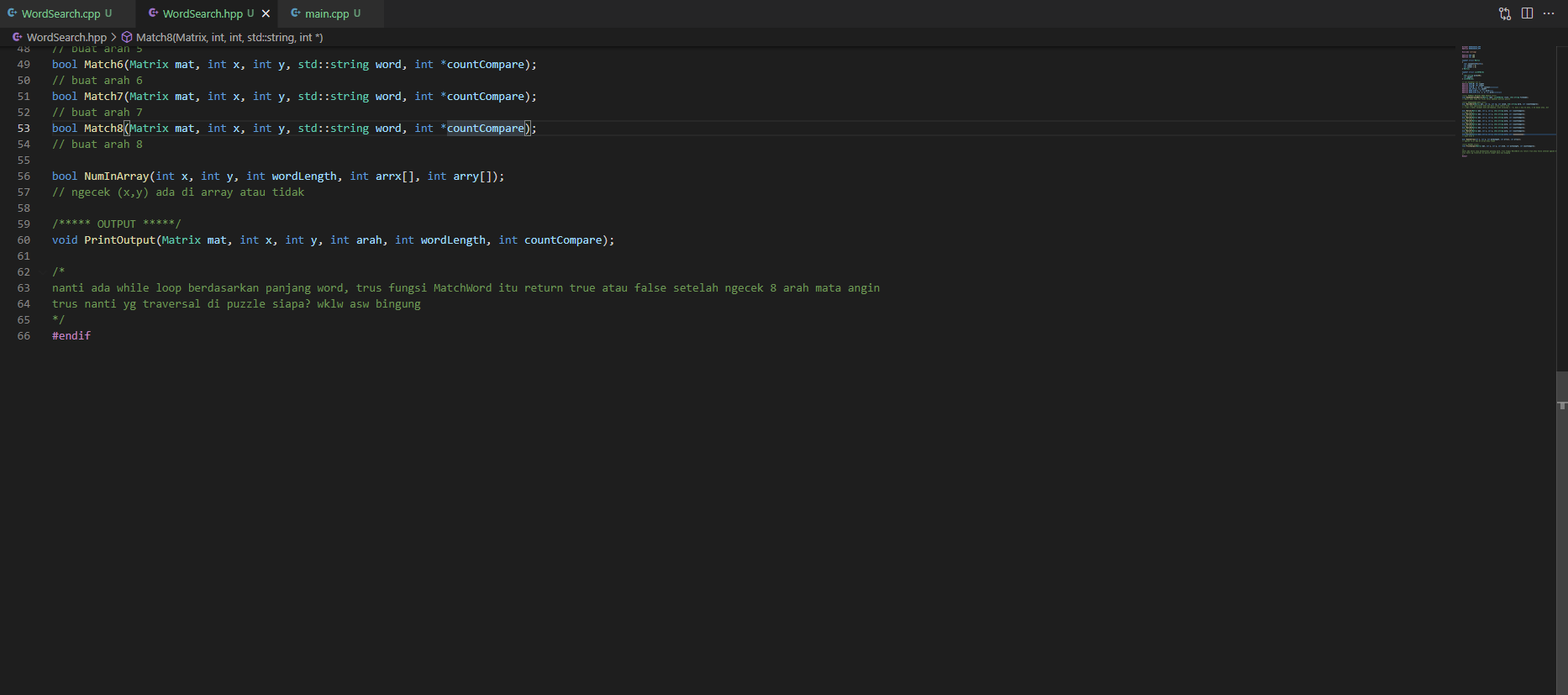
Gambar 3 Contoh pencarian kata “KATA” yang mencoba semua arah kemudian menemukan arah yang tepat

1. 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.

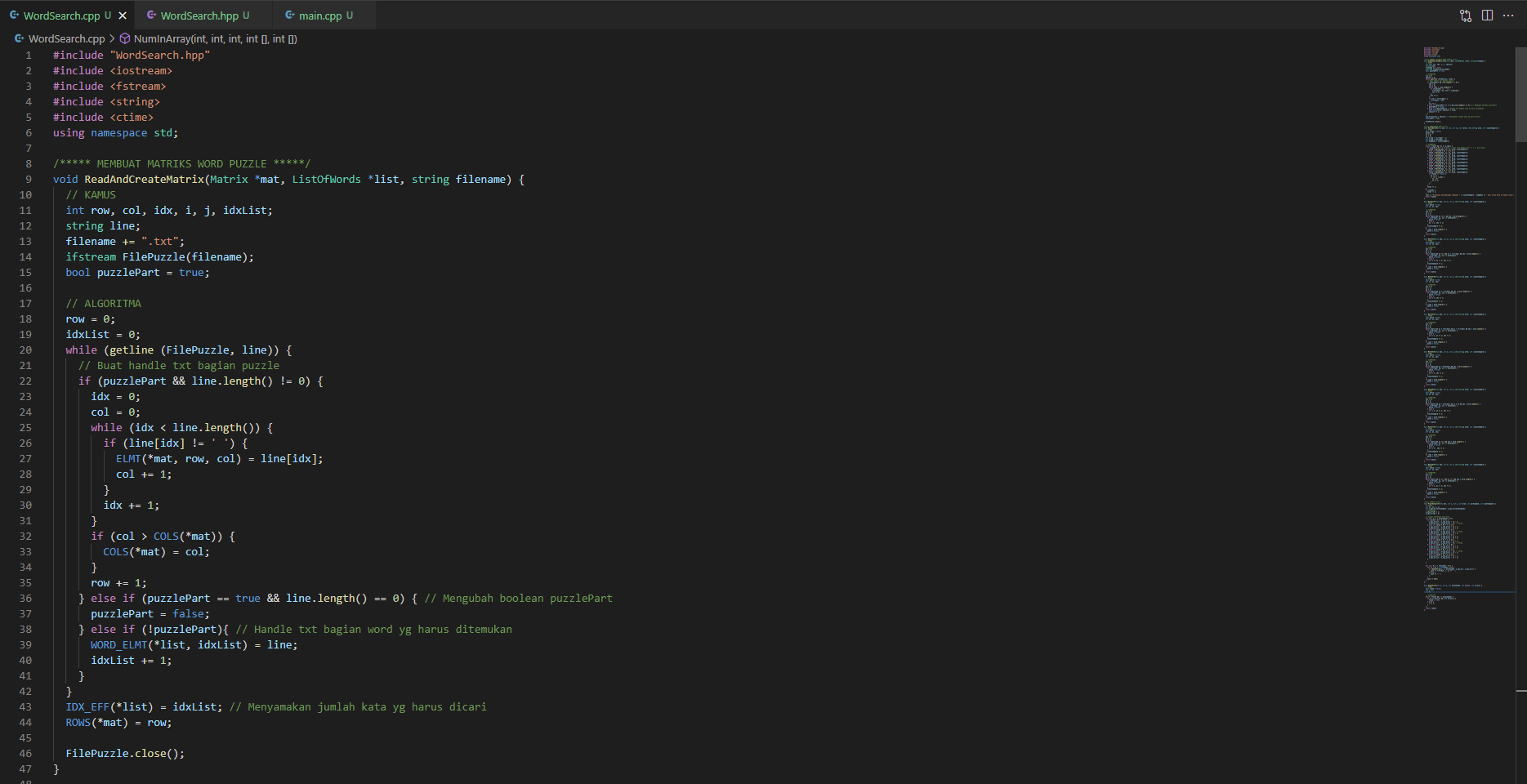
# Source Program

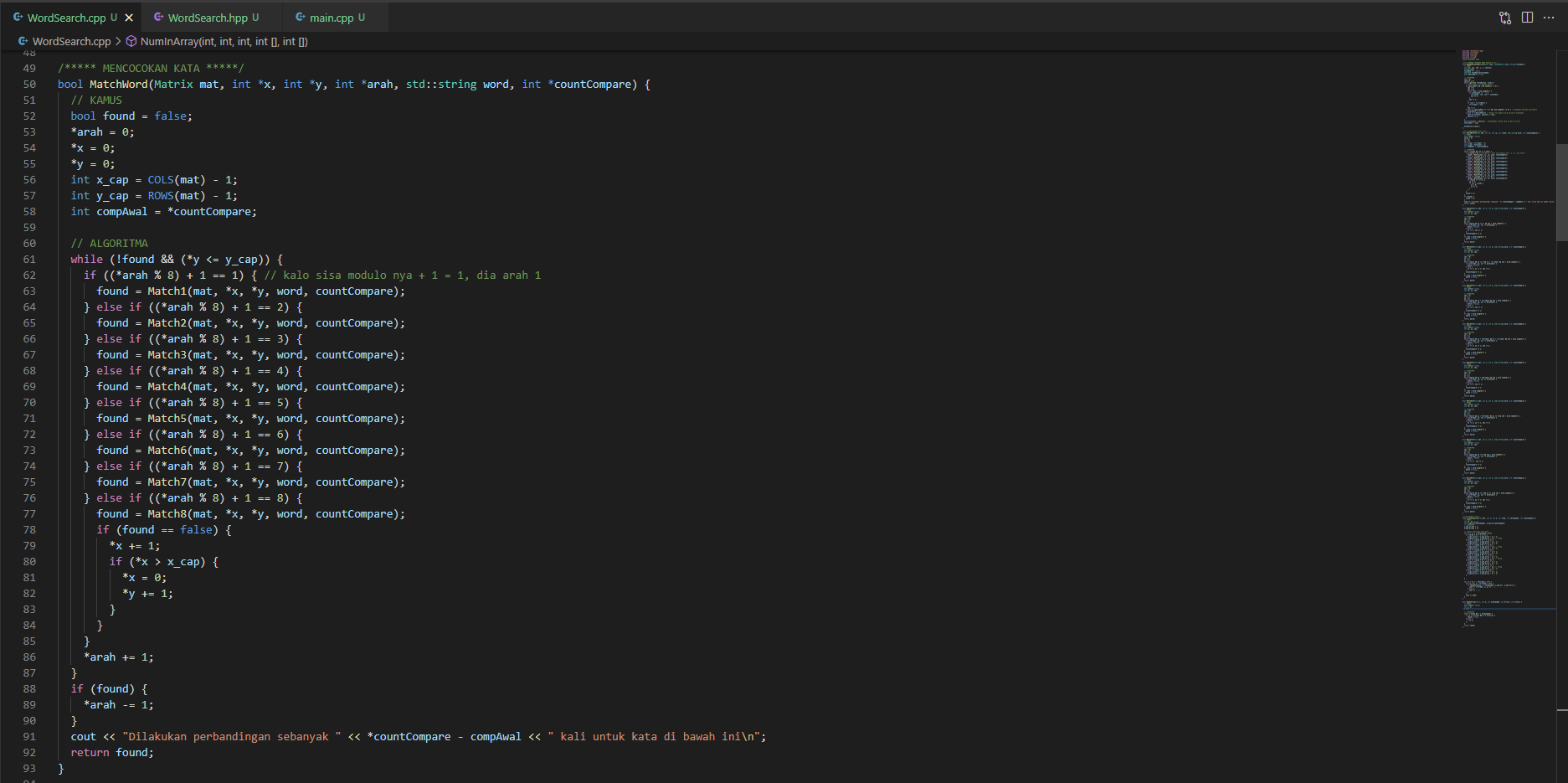
1. WordSearch.hpp

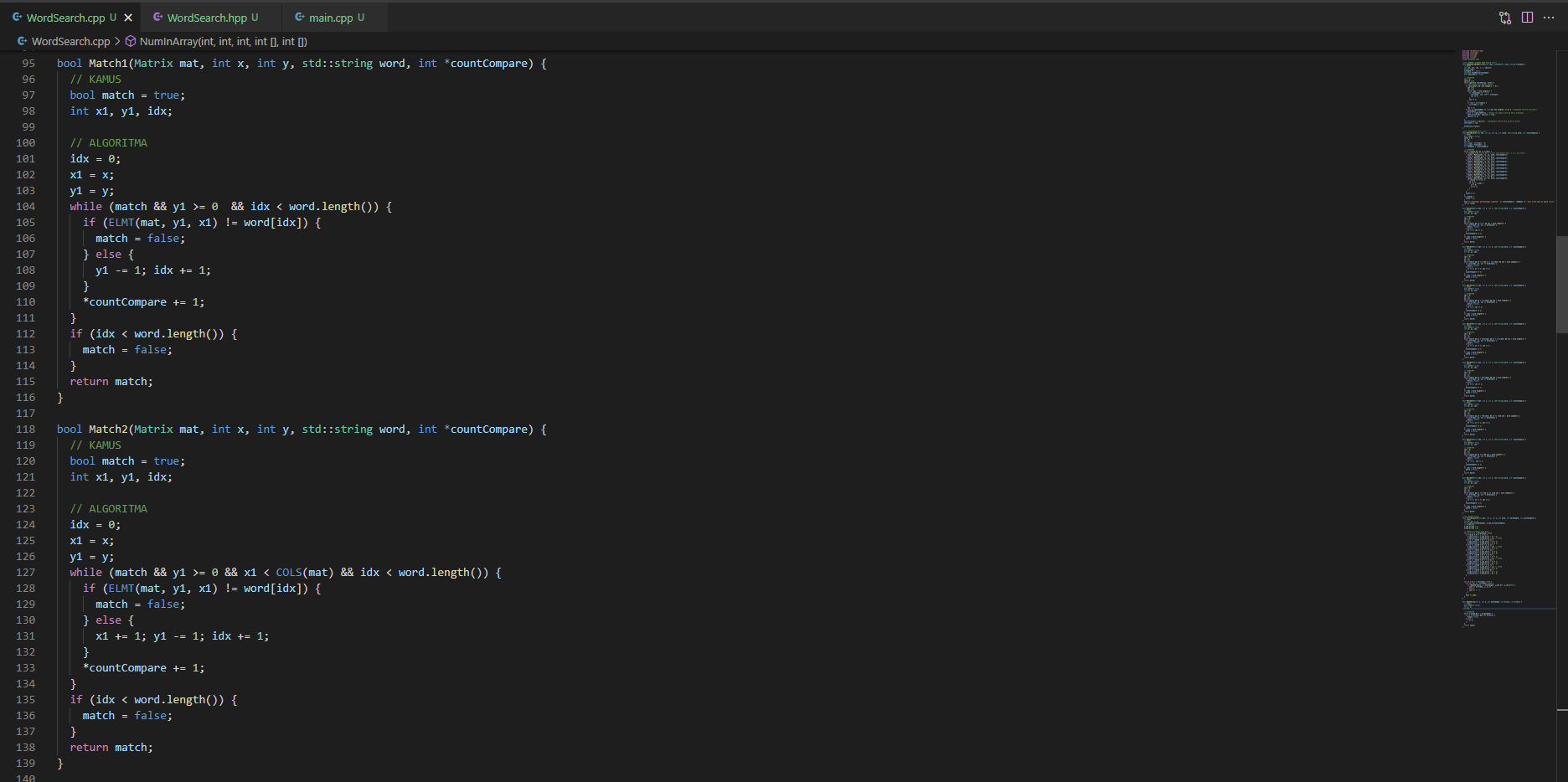


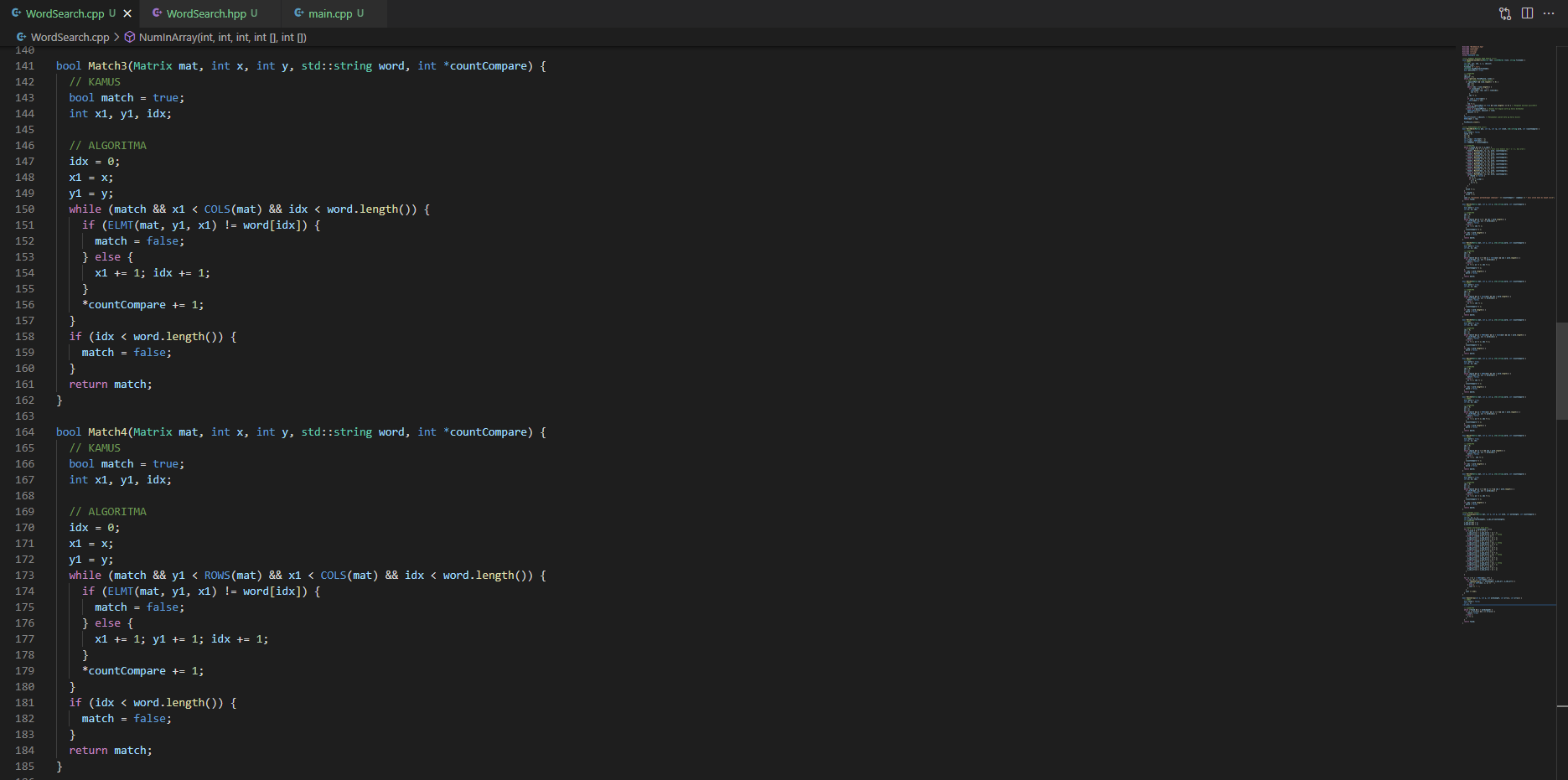


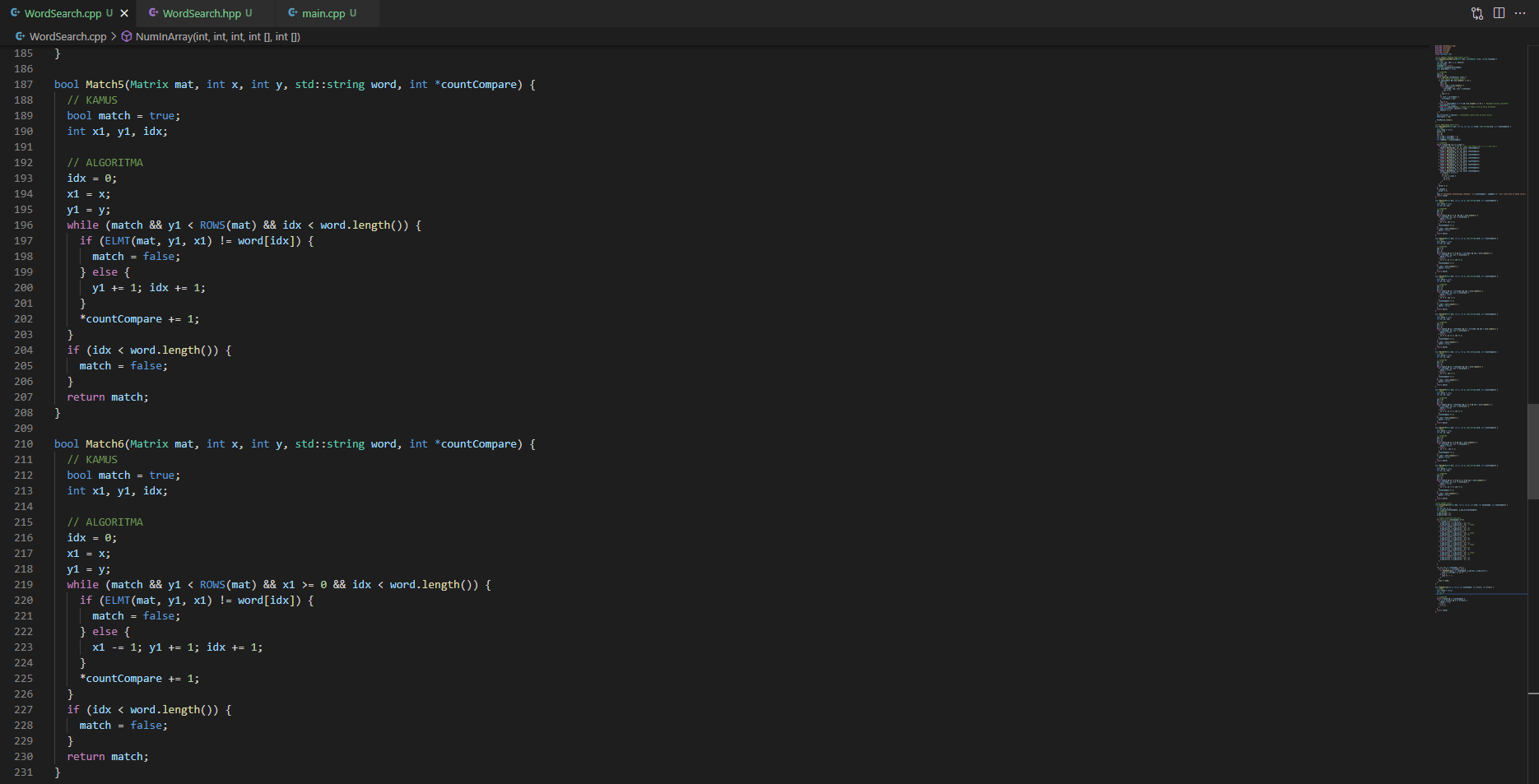
1. WordSearch.cpp

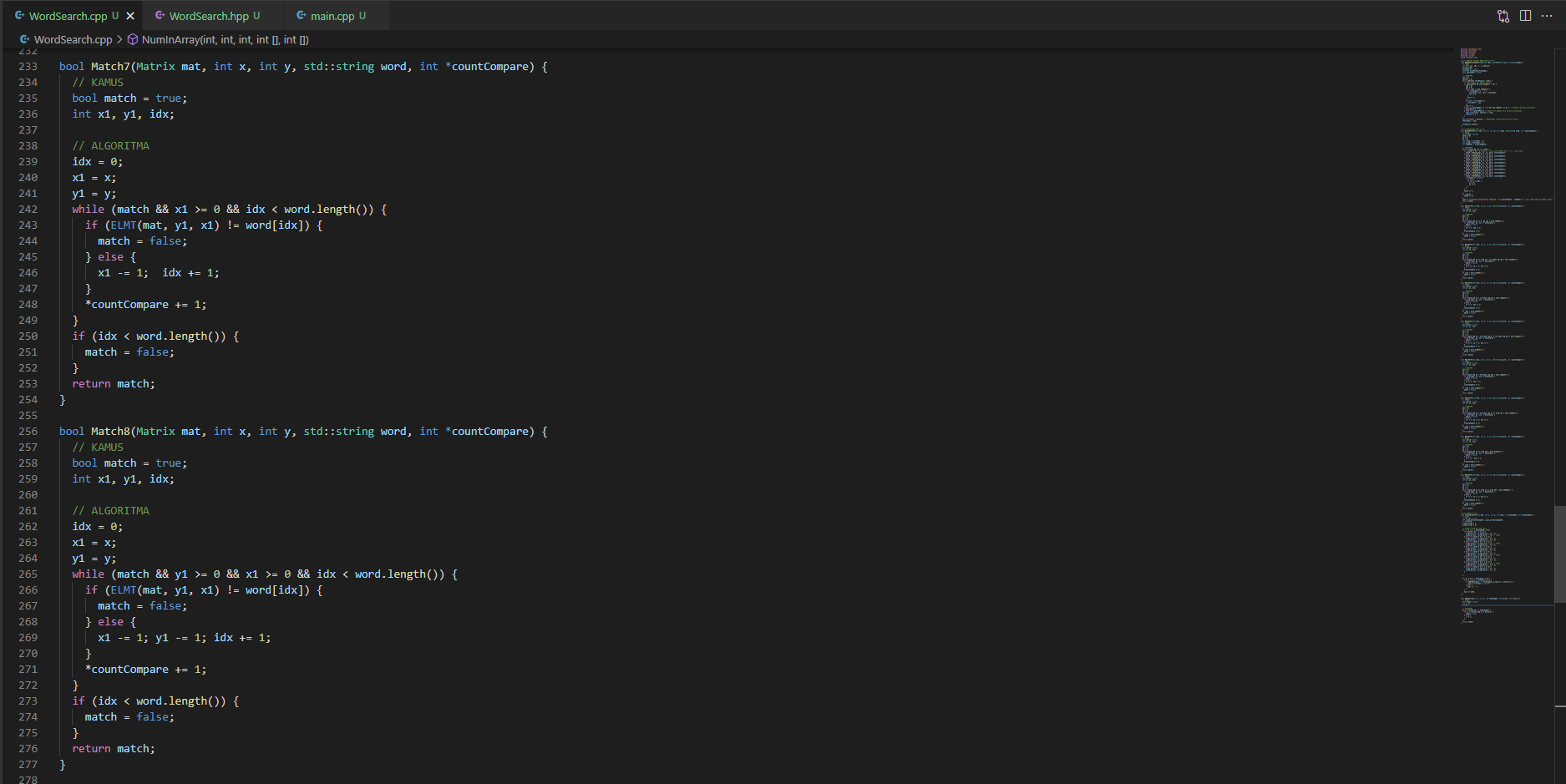


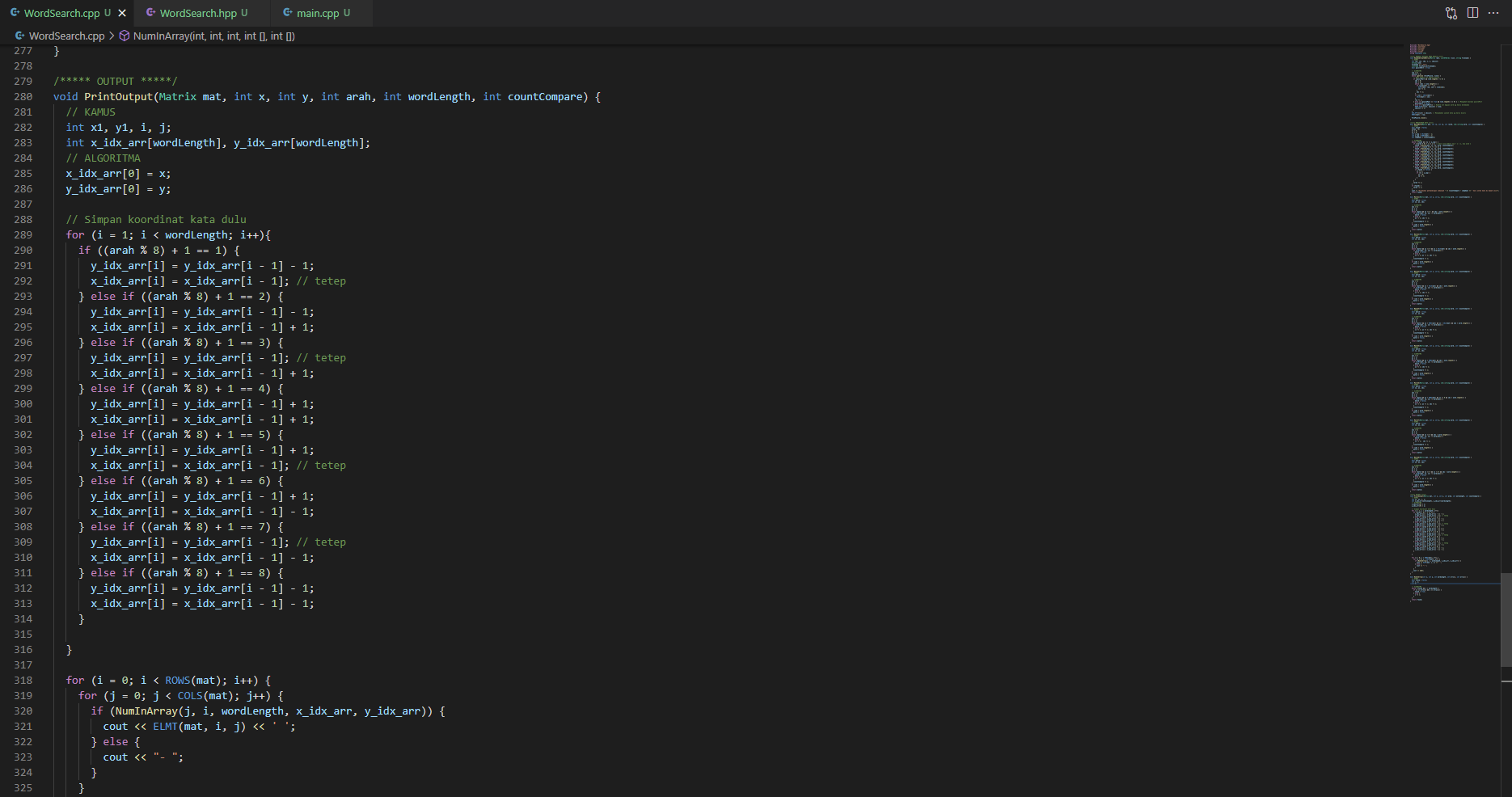


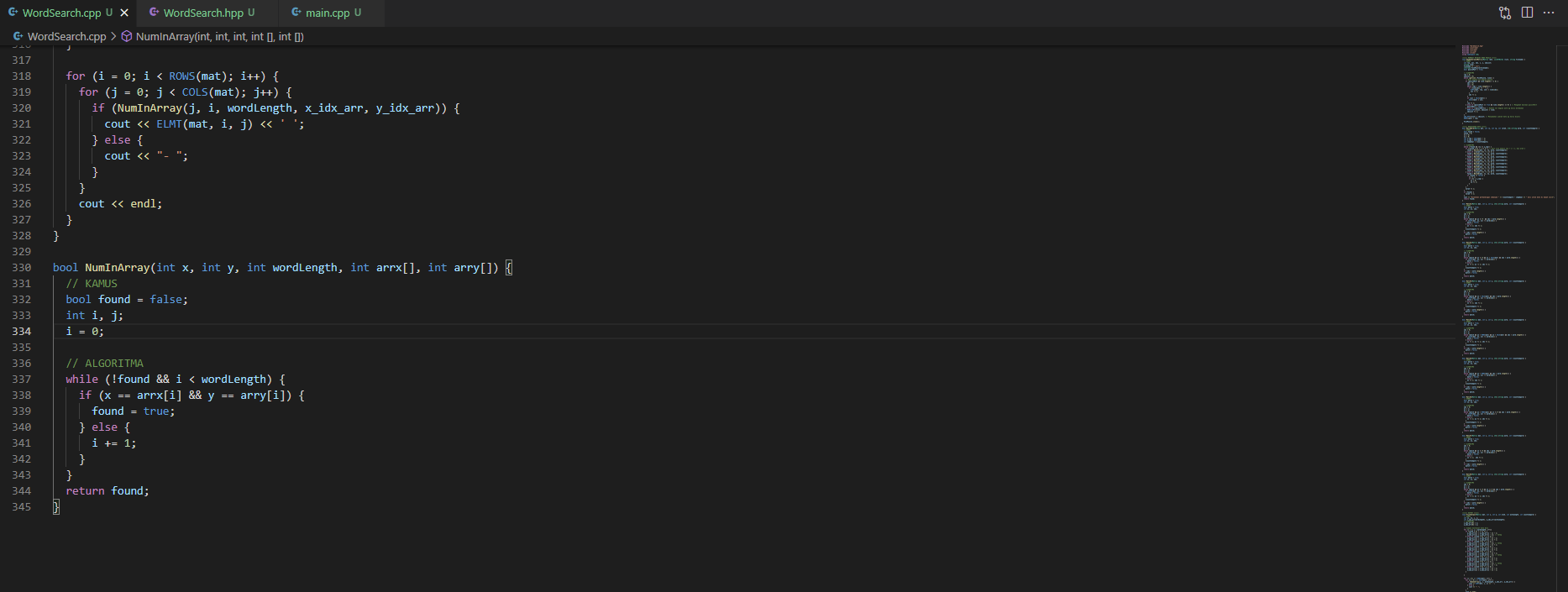




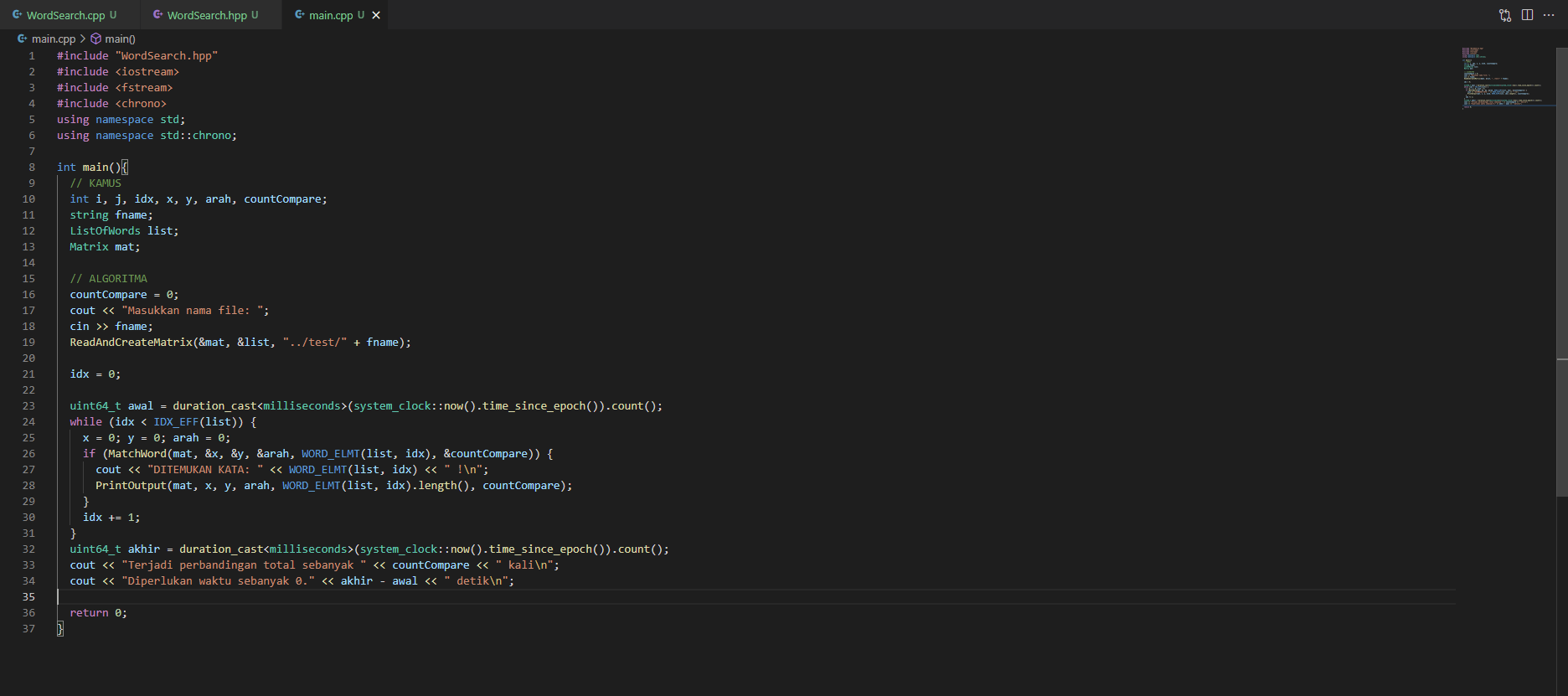








1. main.cpp



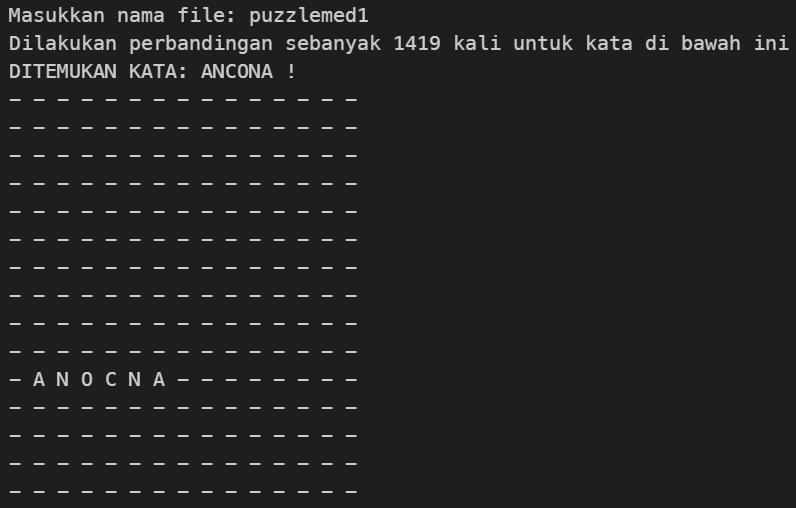
# Input dan Output Program

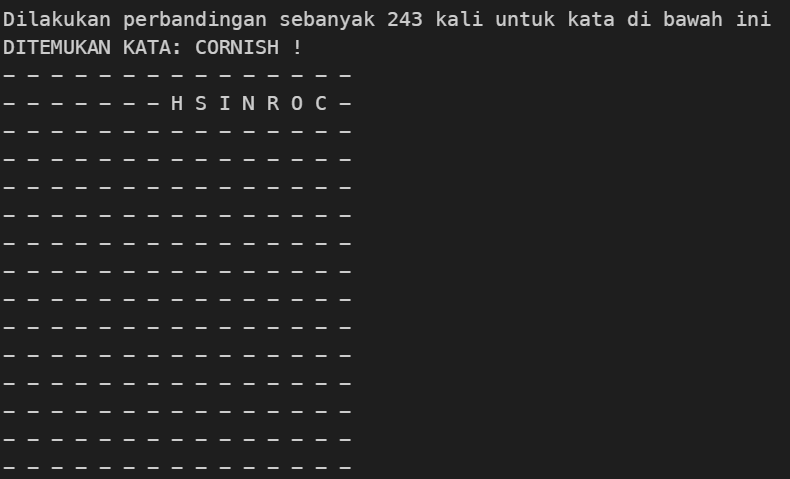
1. Contoh 1 (ukuran 15 x 15)

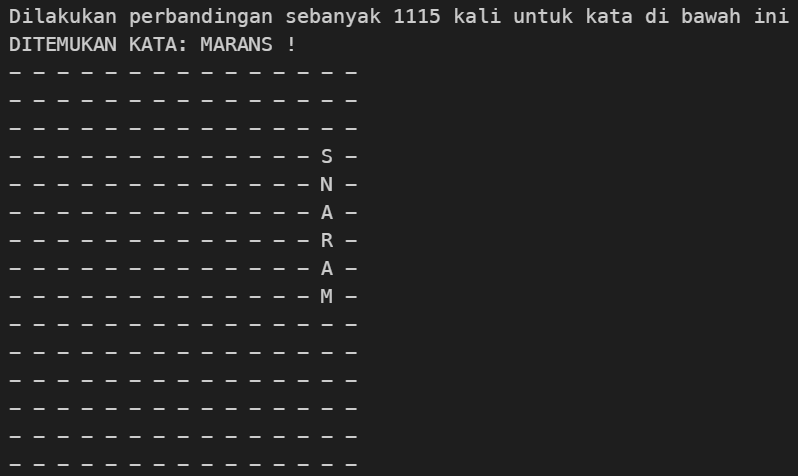
Input:

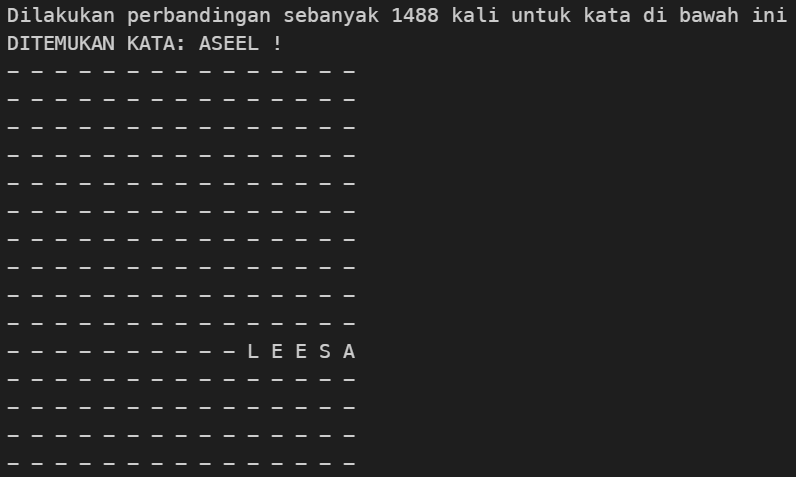


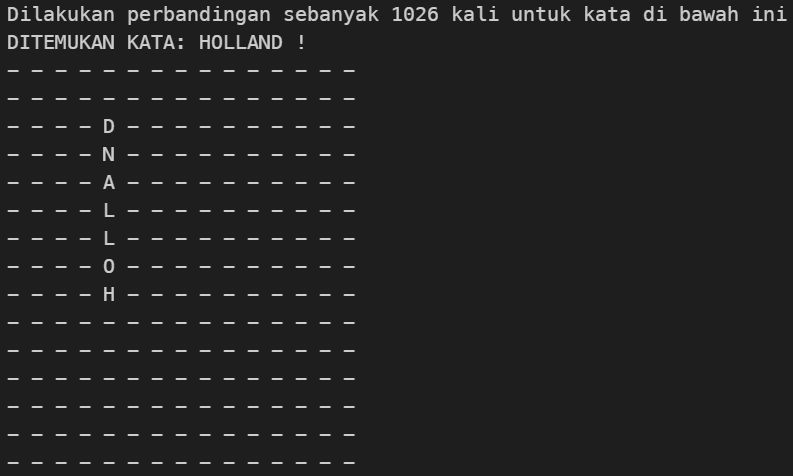
Output:

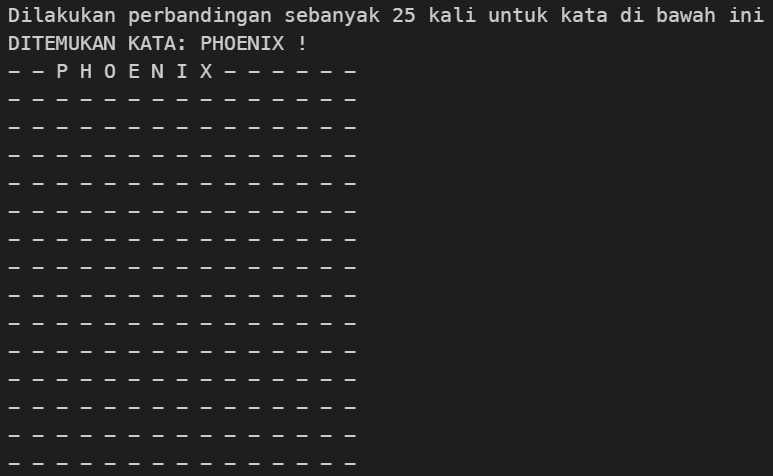


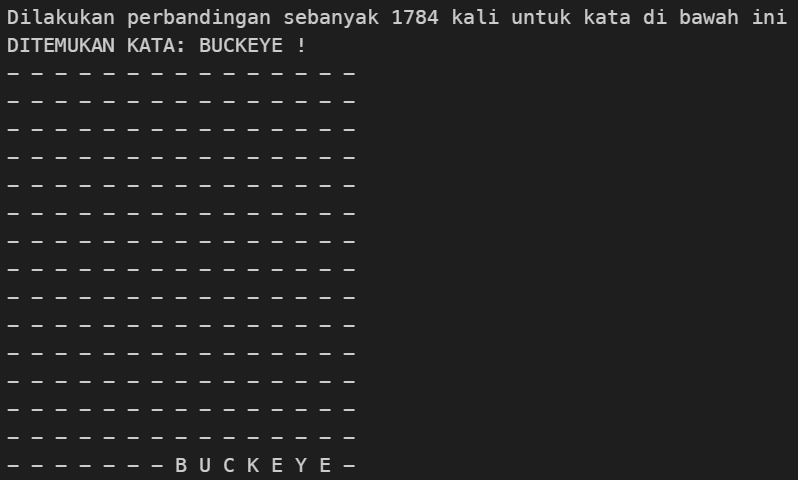


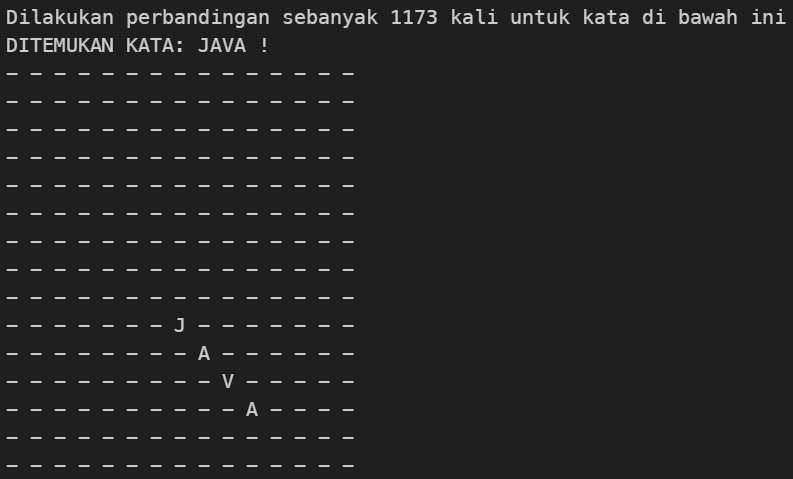


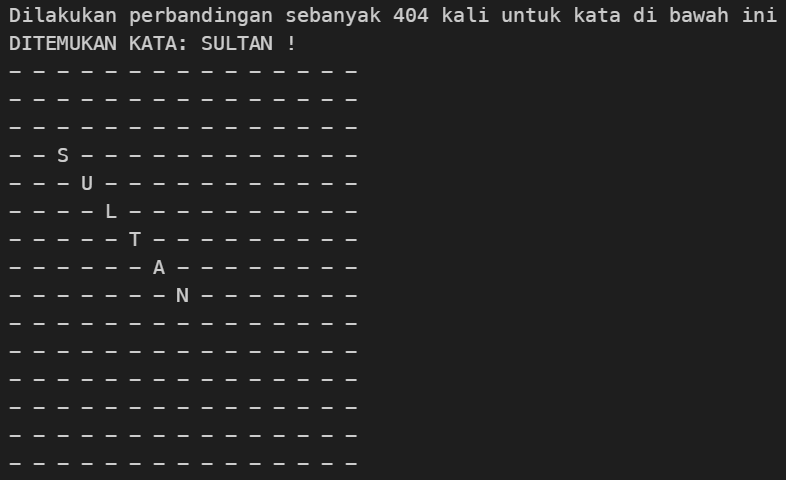


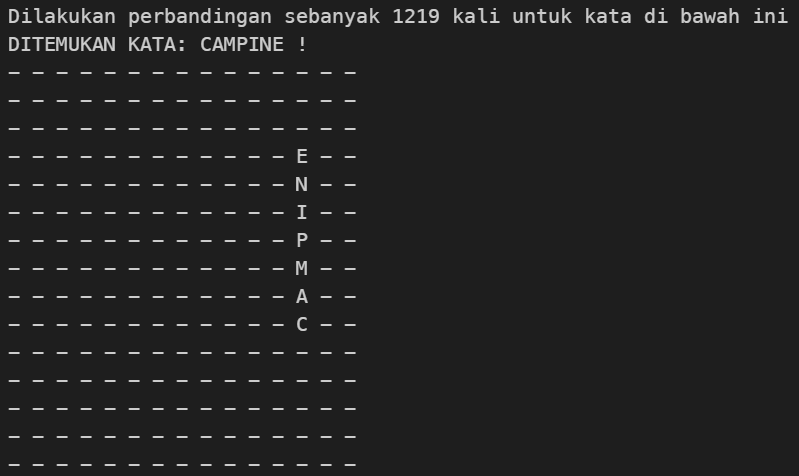


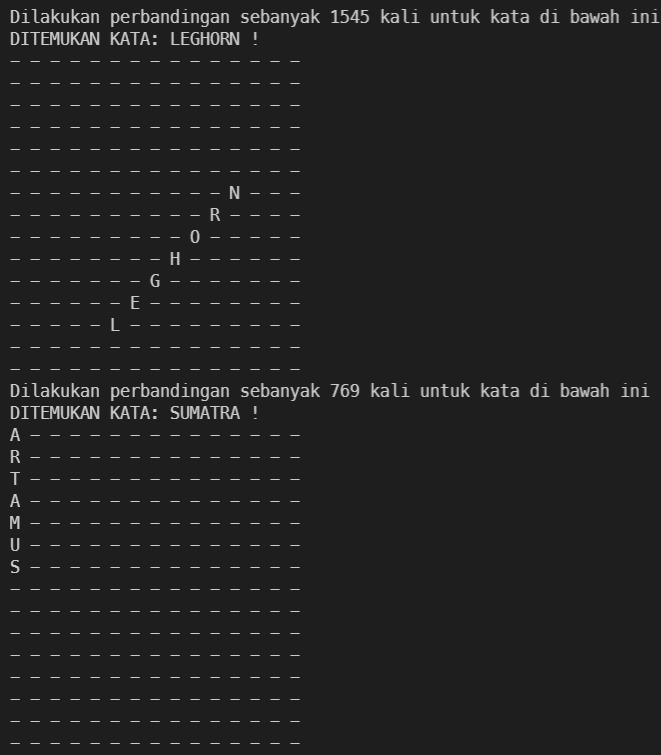




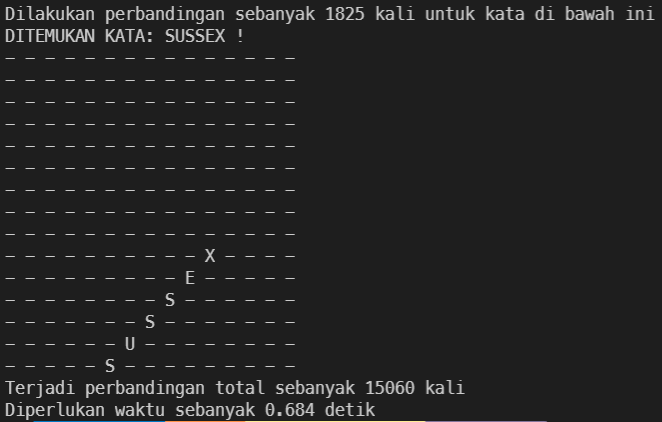






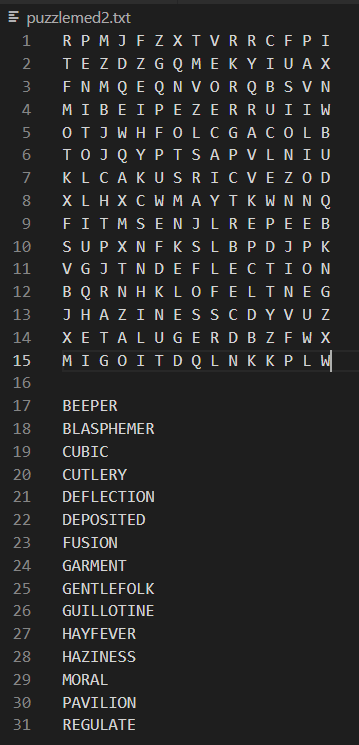






1. Contoh 2 (ukuran 15 x 15)

Input:

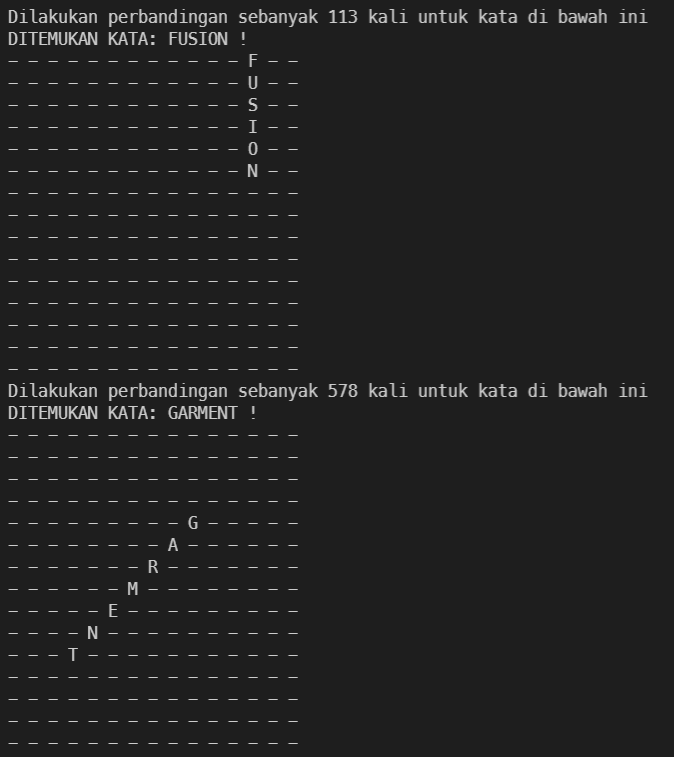


Output:





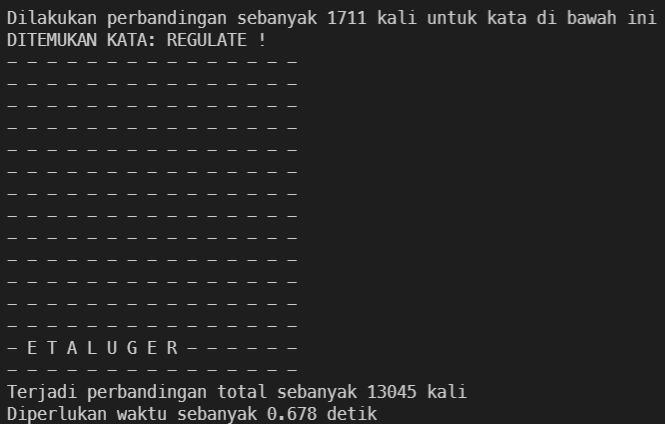






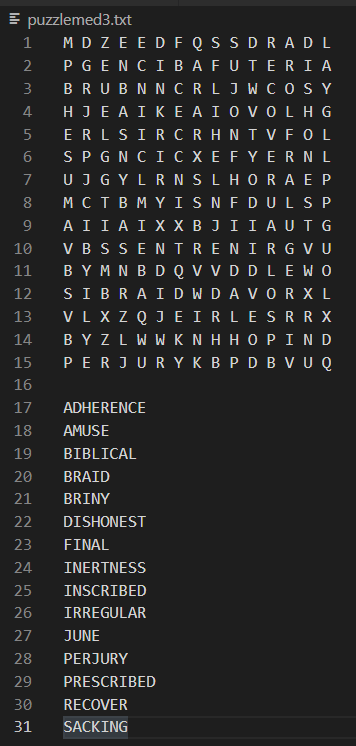




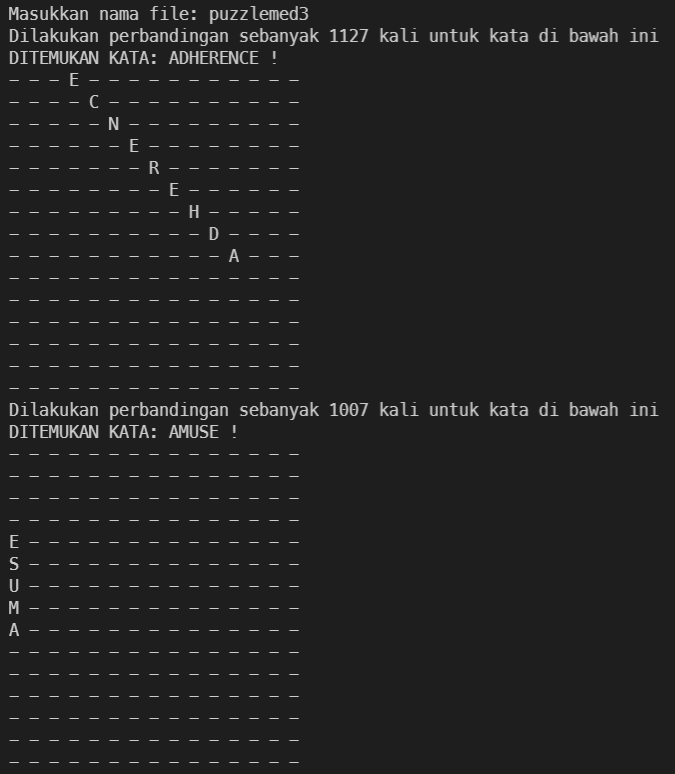


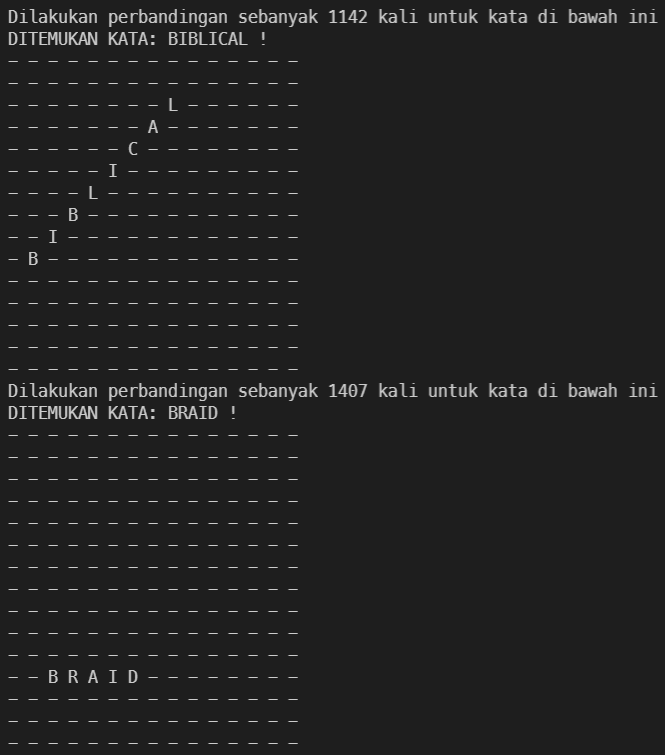
1. Contoh 3 (ukuran 15 x 15)

Input:



Output:



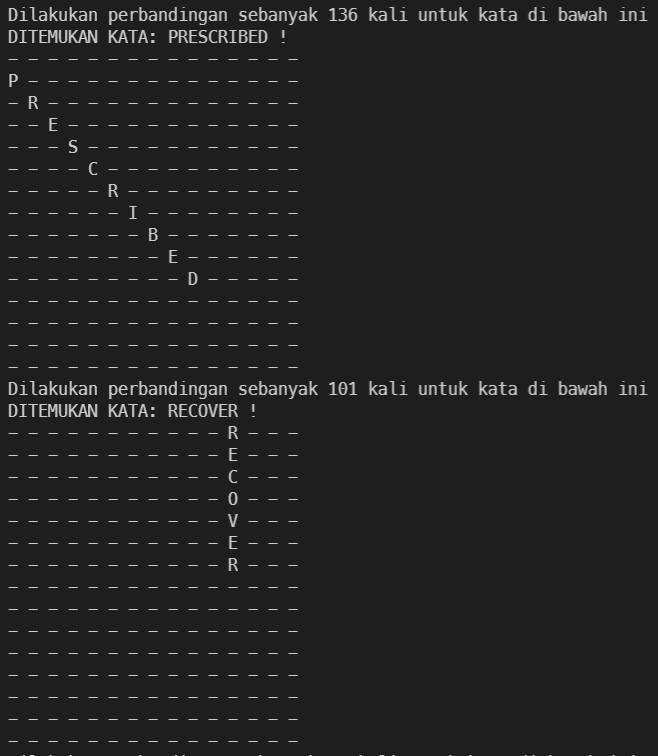


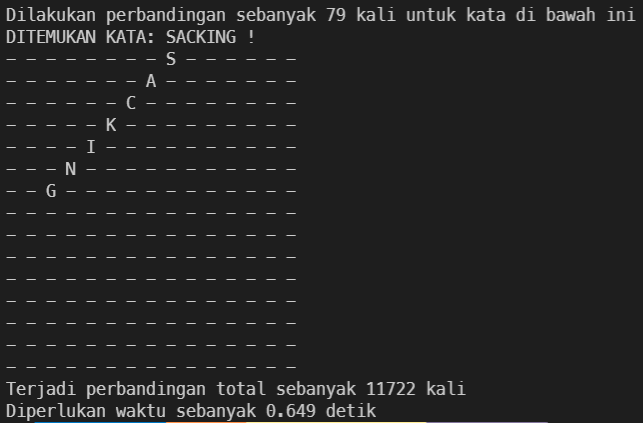










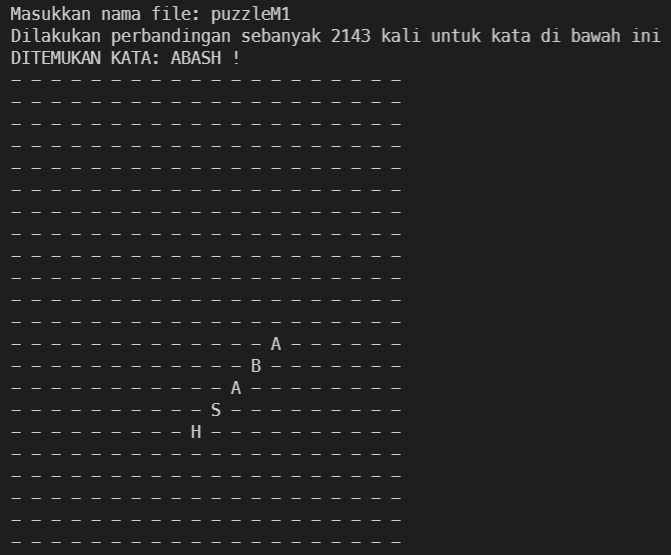


1. Contoh 4 (ukuran 22 x 20)

Input:



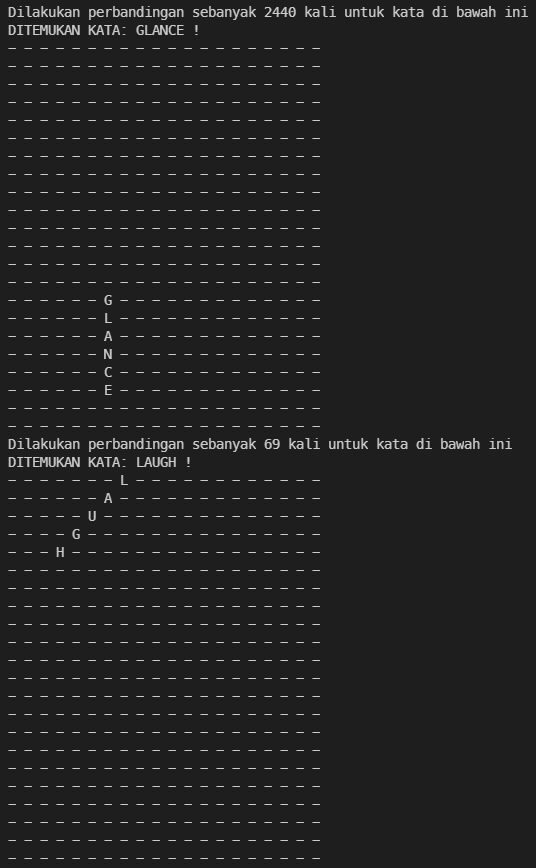
Output:













1. Contoh 5 (ukuran 22 x 20)

Input:



Output:





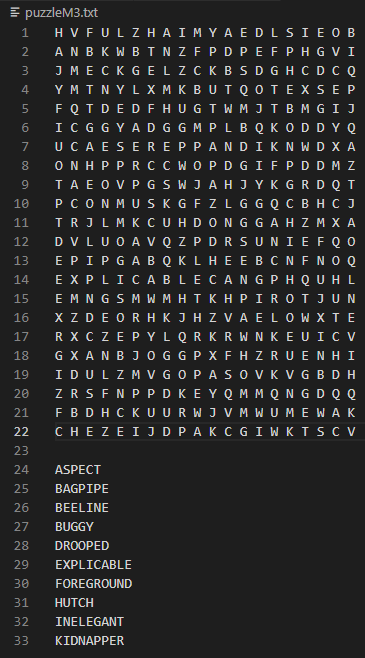




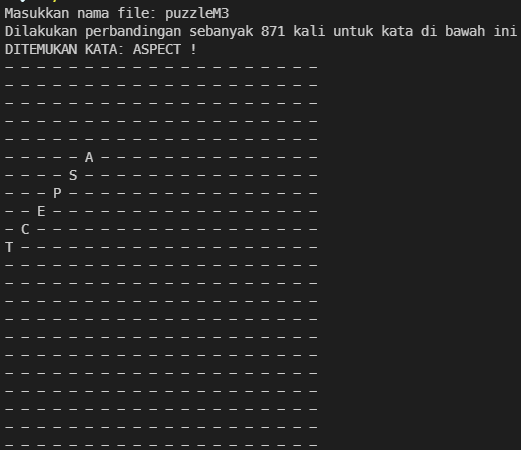


1. Contoh 6 (ukuran 22 x 20)

Input:



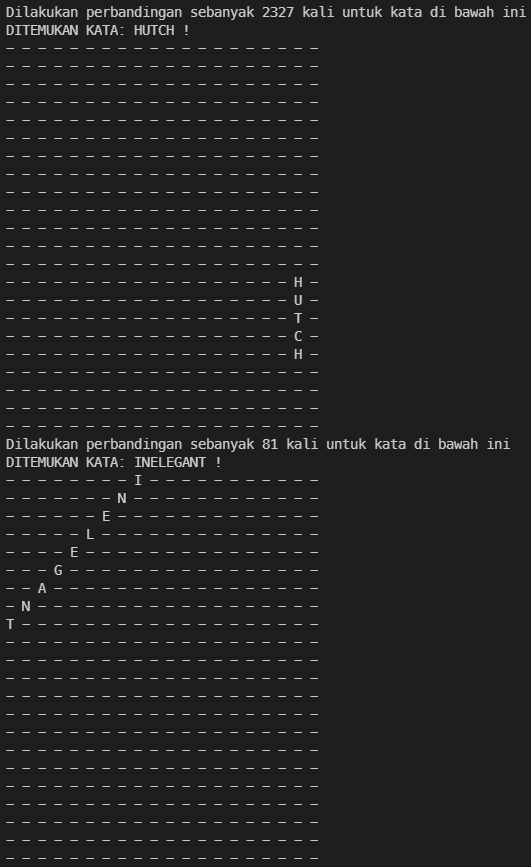
Output:







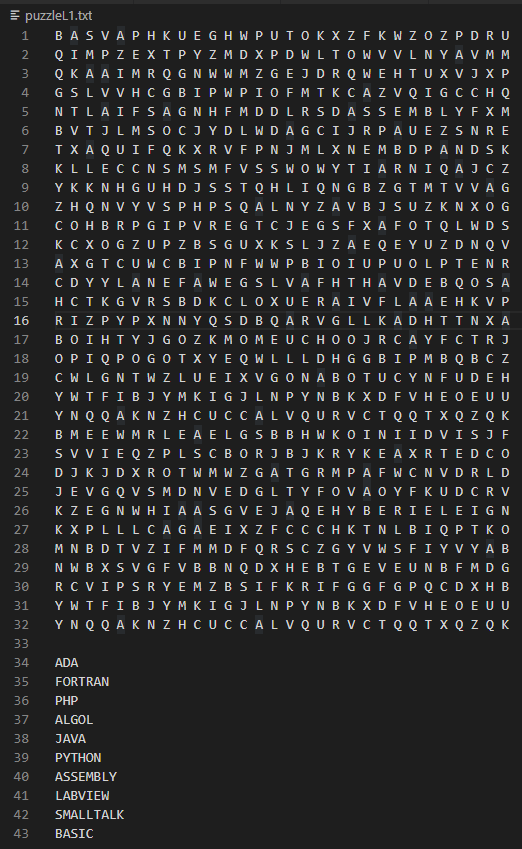






1. Contoh 7 (ukuran 32 x 30)

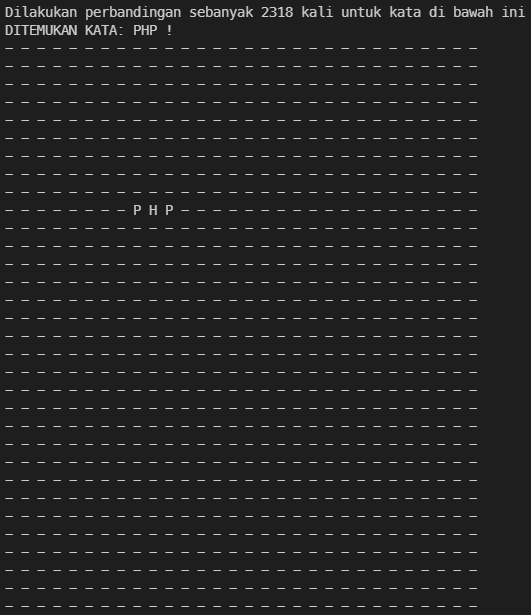
Input:



Output:

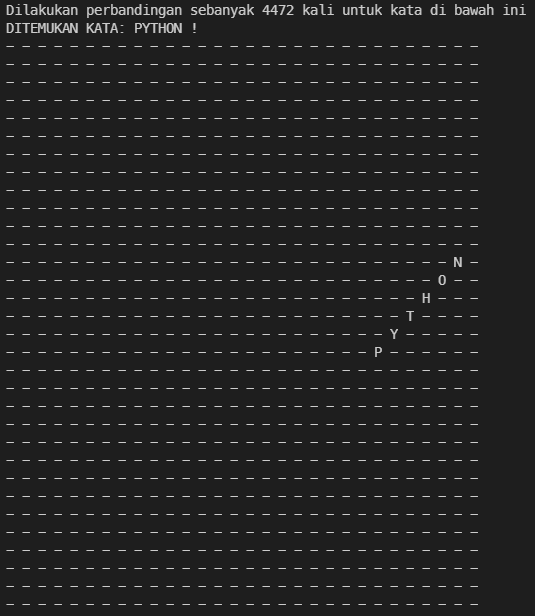


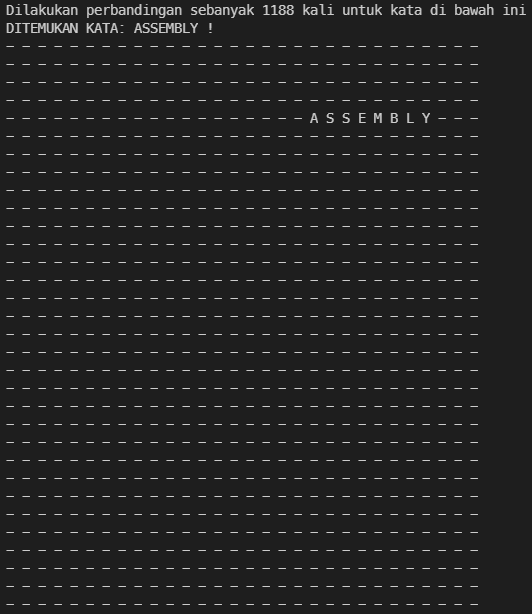


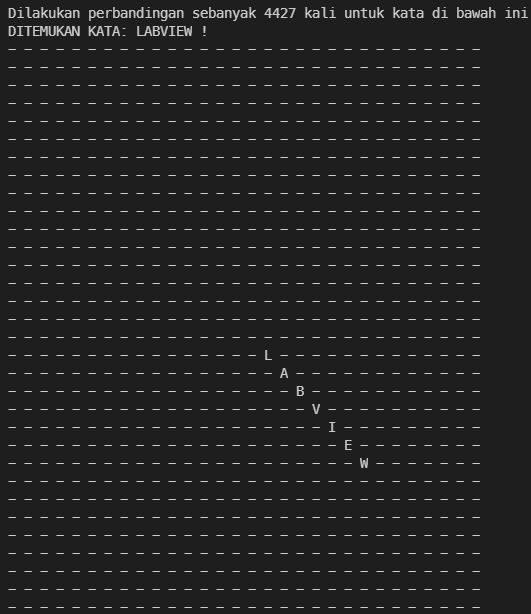


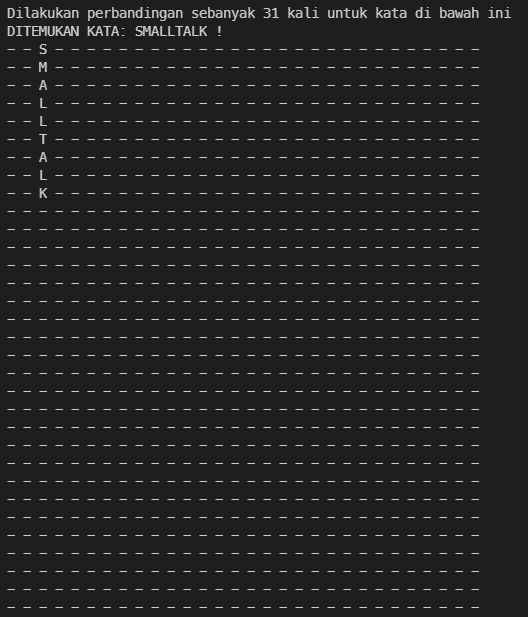


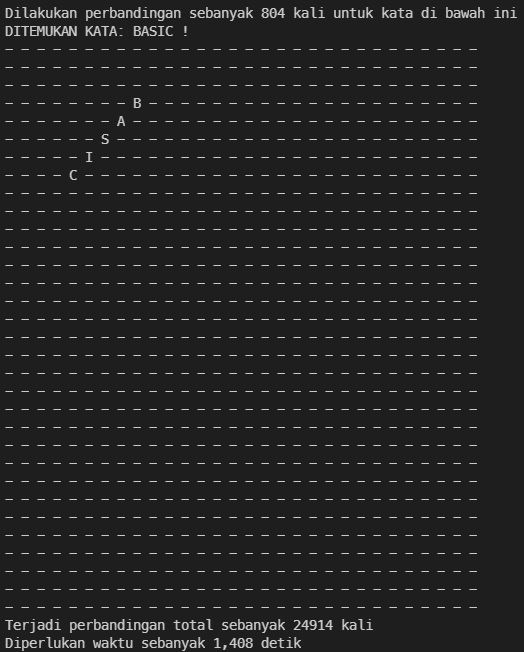






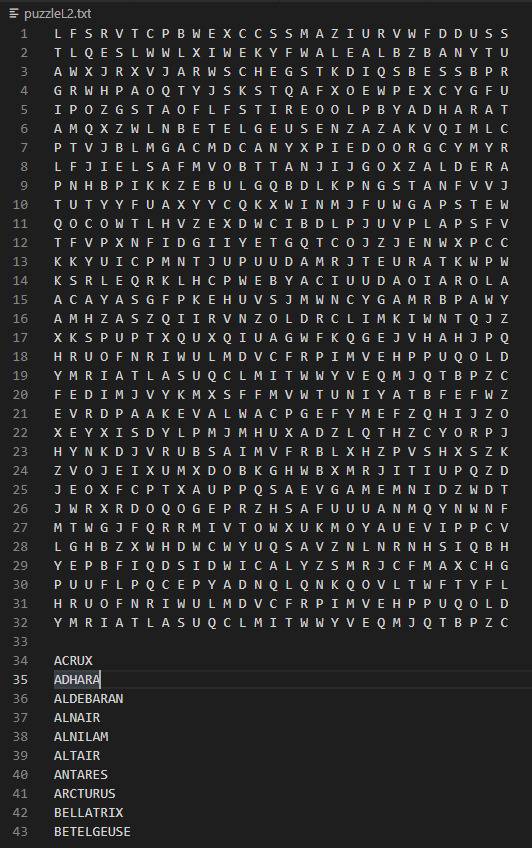






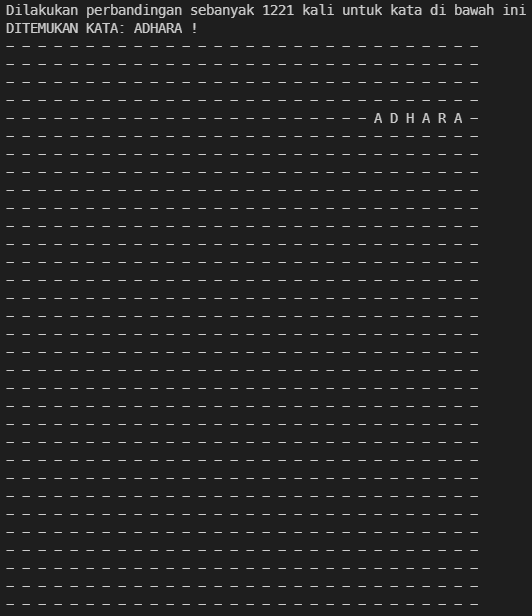
1. Contoh 8 ( ukuran 32 x 30 )

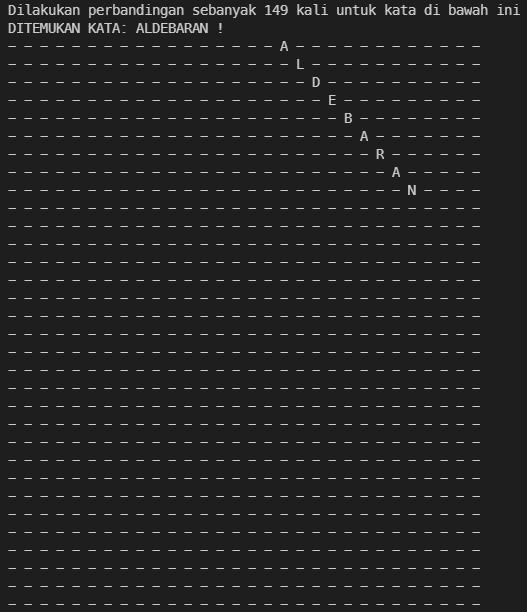
Input:

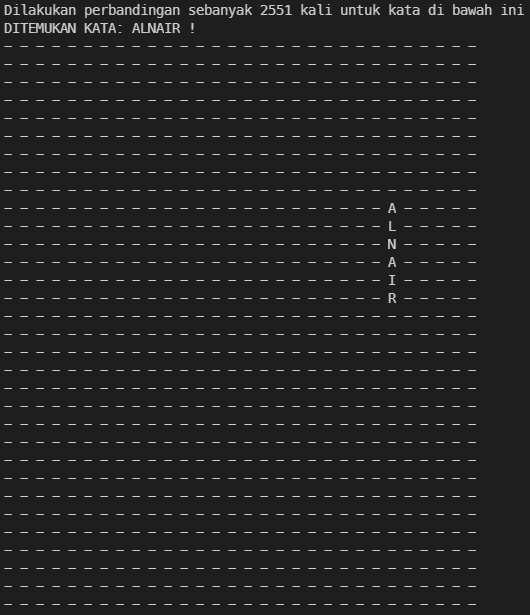


Output:

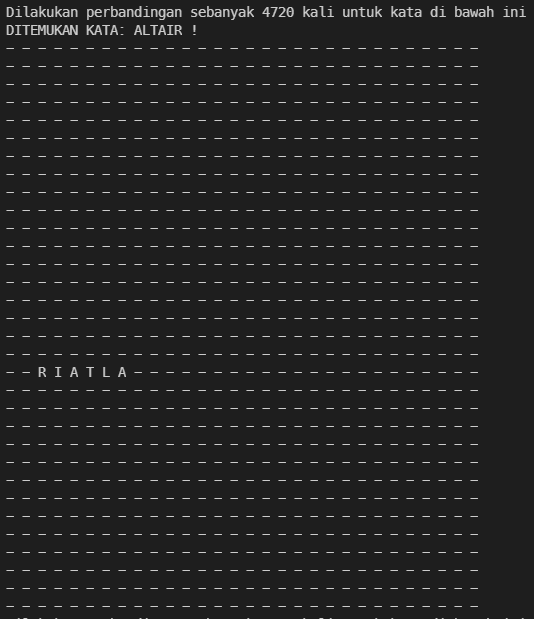


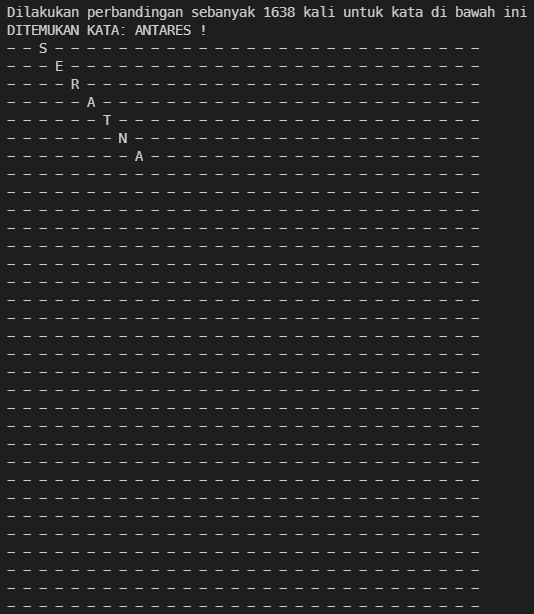


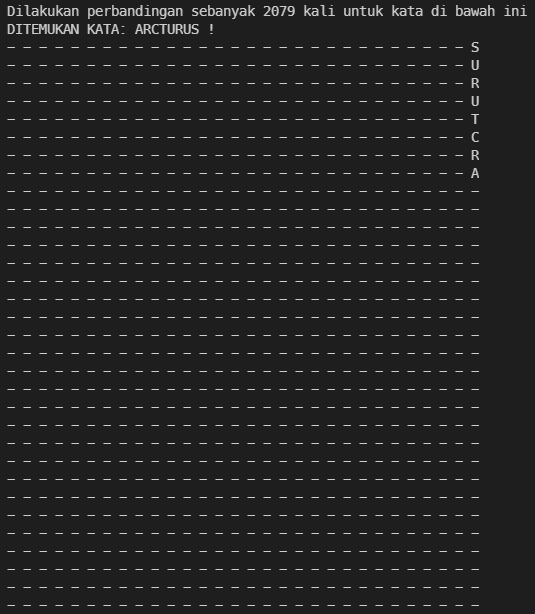


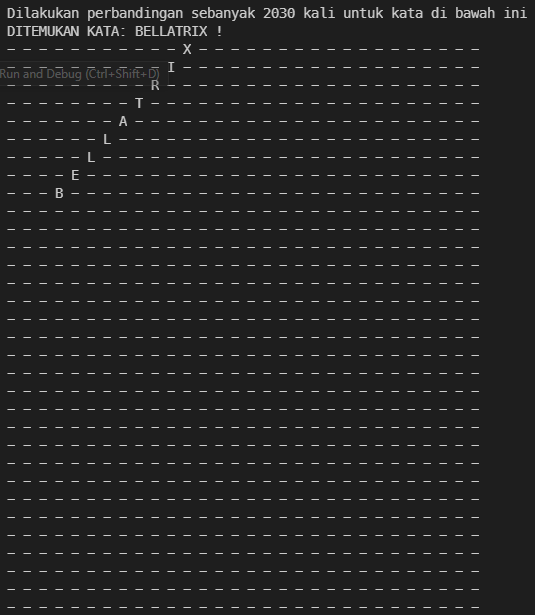


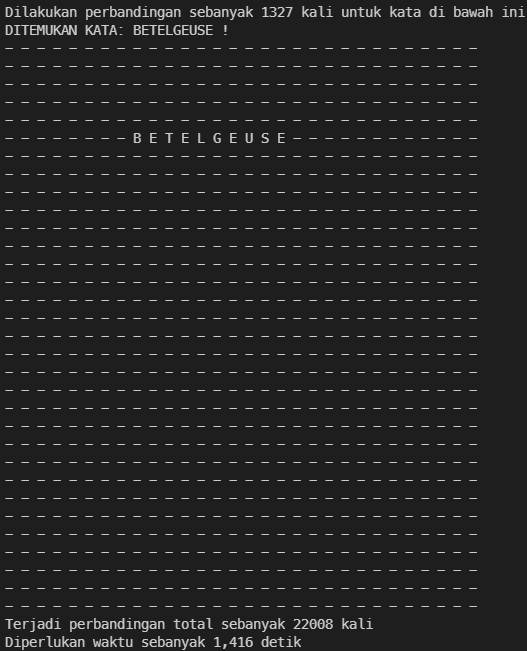






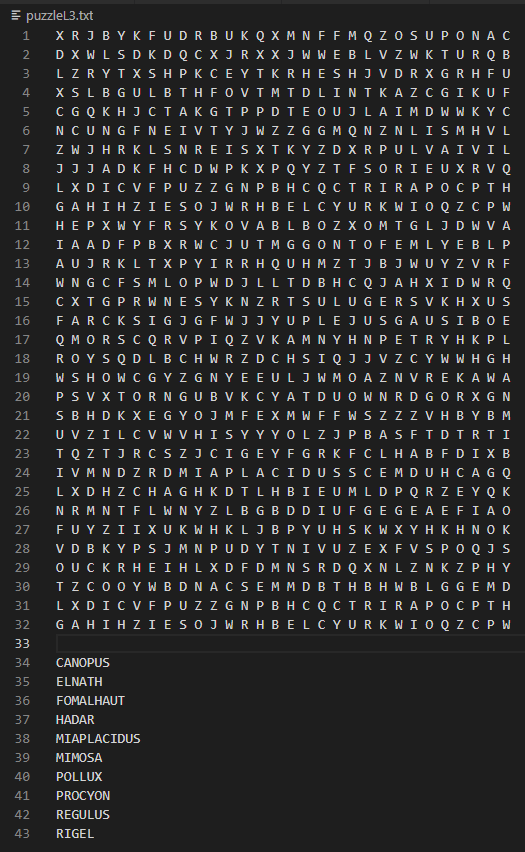




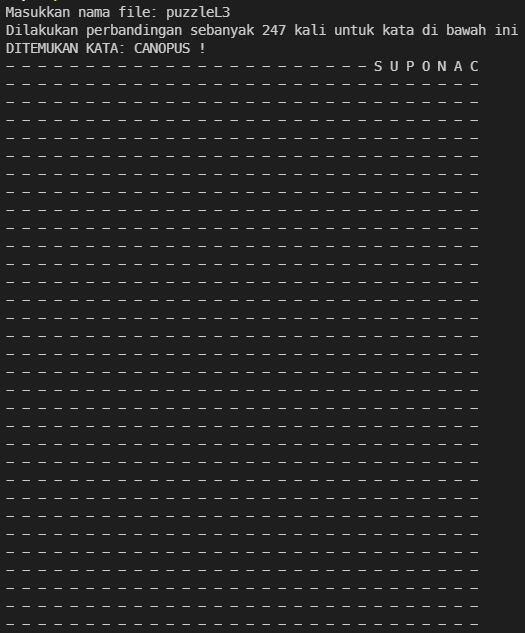


1. Contoh 9 (ukuran 32 x 30)

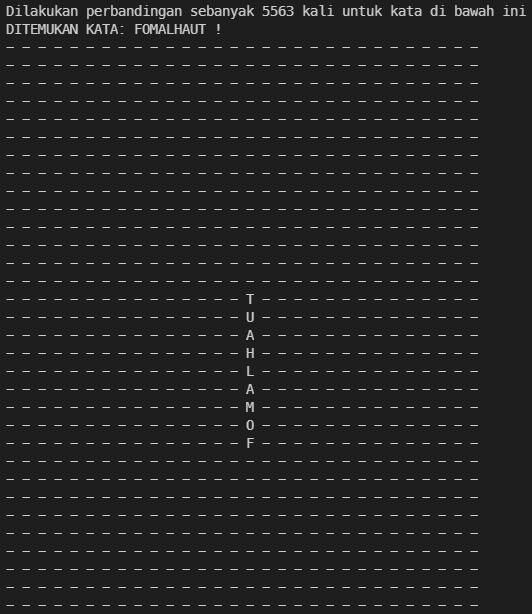
Input:



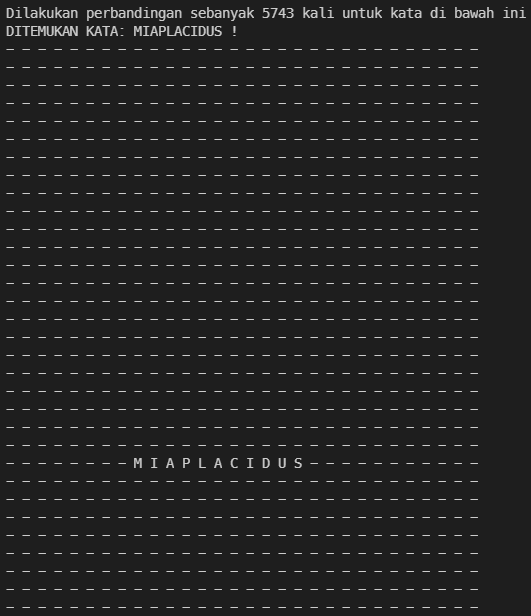
Output:

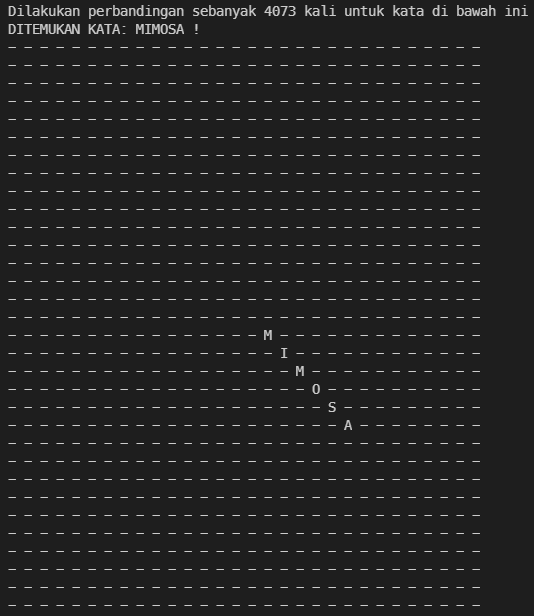


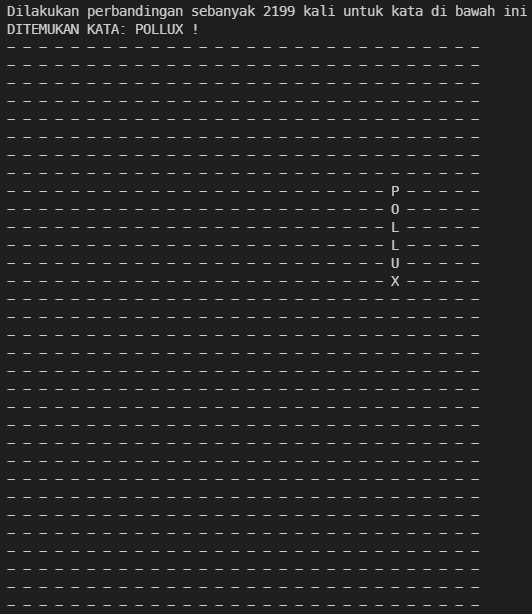


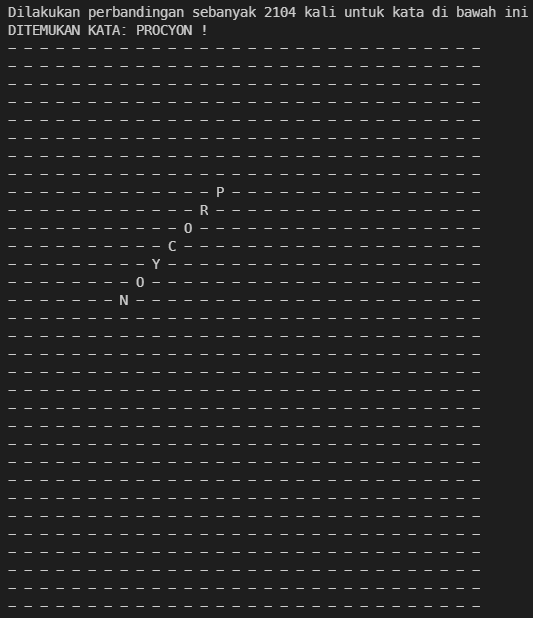




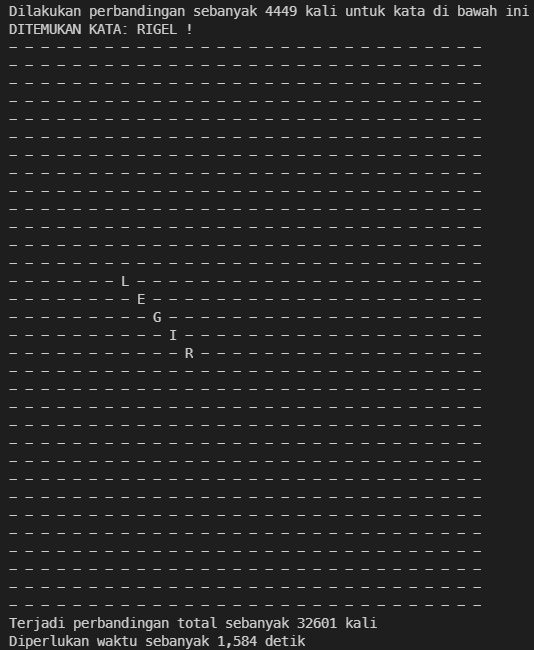












# Alamat Kode Program

Program dapat diunduh dari alamat berikut:

# Tabel Penilaian

|  |  |  |
| --- | --- | --- |
| Poin | Ya | Tidak |
| 1. Program berhasil dikompilasi tanpa kesalahan (no syntax error). | ✓ |  |
| 1. Program berhasil *running.* | ✓ |  |
| 1. Program dapat membaca file masukan dan menuliskan luaran. | ✓ |  |
| 1. Program berhasil menemukan semua kata di dalam puzzle. | ✓ |  |