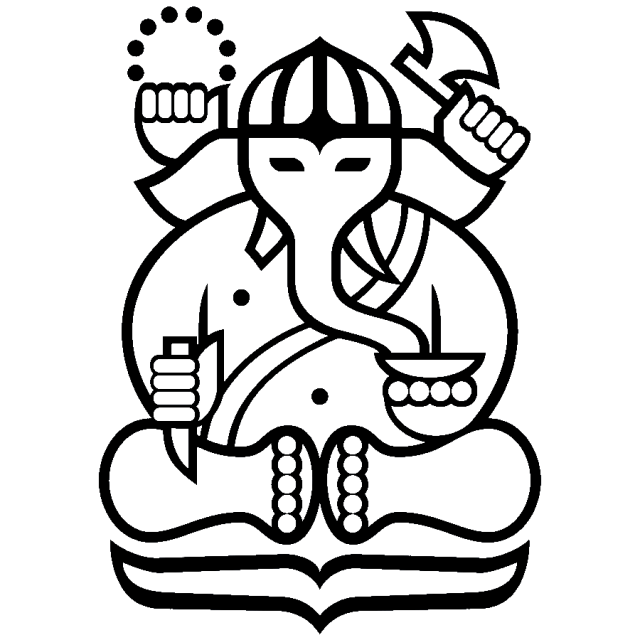
**LAPORAN TUGAS KECIL 1**

**IF2211 STRATEGI ALGORITMA**

Penyelesaian game iq puzzle dengan algoritma *bruteforce*



Disusun Oleh:

Aryo Bama Wiratama (13523088)

**Program Studi Teknik Informatika**

**Sekolah Teknik Elektro dan Informatika**

**Institut Teknologi Bandung**

**Tahun 2025**

1. **Algoritma Bruteforce**

Program ini menggunakan algoritma *bruteforce* untuk menemukan solusi dalam gim iq puzzler. Algoritma *bruteforce* adalah algoritma yang memiliki pendekatan yang *straighforward* dalam memecahkan masalah. Oleh karena itu, algoritma ini cenderung simpel dan mudah dimengerti. Akan tetapi, algoritma ini tidak efisien dan cenderung membutuhkan waktu komputasi yang lama.

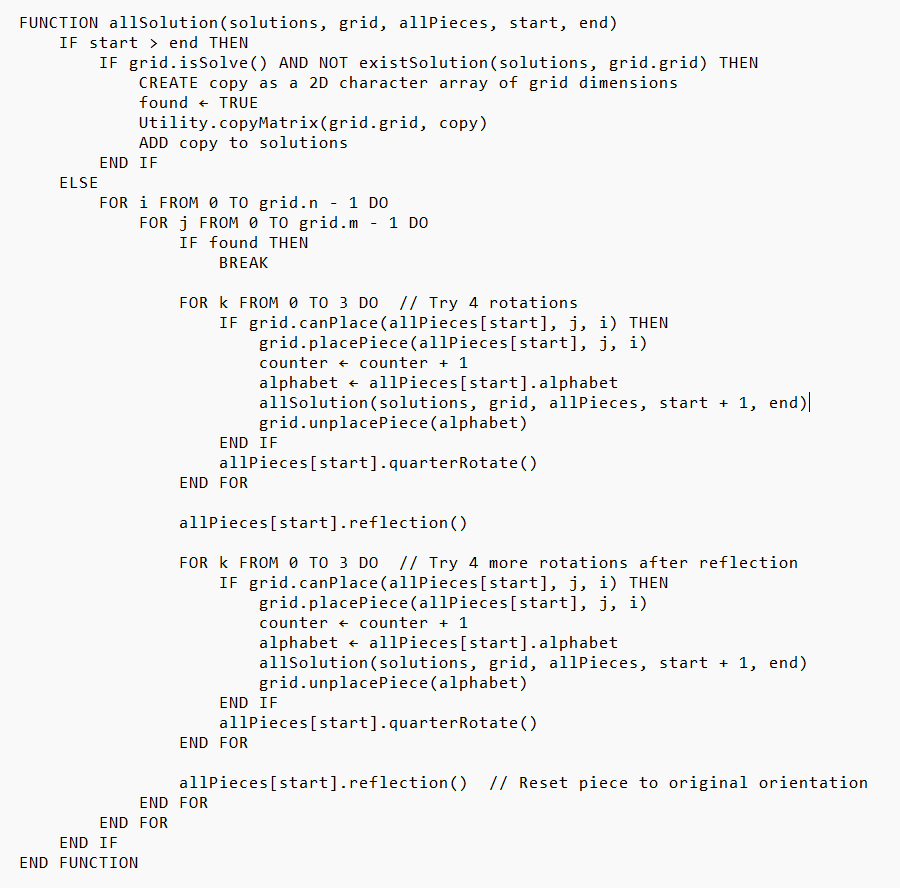
Berikut penjelasan algoritma bruteforce dalam mencari solusi pada gim iq puzzler:

1. Misalkan terdapat P buah piece dan papan berukuran n x m
2. Susunlah piece tersebut. Penyusunan piece tidak berdasarkan syarat apapun.
3. Ambil satu piece kemudian letakkan pada papan mulai dari titik O (0,0).
4. Kemudian ambil piece selanjutnya. Coba letakkan piece dalam papan dimulai dari titik O. Apabila suatu titik telah ditempati oleh piece lain, maju ke titik selanjutnya.
5. Lakukan hal no.4 hingga mencapai piece terakhir.
6. Apabila dalam proses no 4 dan 5 terdapat piece yang tidak menemukan tempat kosong pada papan, coba lakukan semua gerakan rotasi dan refleksi.
7. Apabila langkah no 6 juga tidak berhasil, lakukan backtracking dengan cara melakukan rotasi dan refleksi pada piece sebelumnya. Kemudian letakkan kembali piece yang gagal ditempatkan.
8. Ulangi langkah 1-7 hingga semua kemungkinan telah dicoba.

Jika terdapat satu solusi yang ditemukan, pencarian tidak usah dilanjutkan.

Dalam kasus terburuk, algoritma ini memiliki kompleksitas waktu T((n x m x 8)P) atau O((n x m)P). Piece ke-I memiliki kemungkinan penempatan sebanyak m x n dan di setiap penempatan terdapat 8 gerakan sehingga setiap piece memilik 8 x n x m kemungkinan. Apabila terdapat P piece, berdasakan aturan perkalian, total kemungkinan dapat dhitung dari perkalian (m x n x 8) sebanyak P kali atau dapat ditulis ((n x m x 8)P.

Berikut adalah pseudocode dari algoritma bruteforce yang digunakan.



1. **Source Code**

Projek ini ditulis dalam bahasa java menggunakan library sebagai berikut:

1. java.util
2. java.io

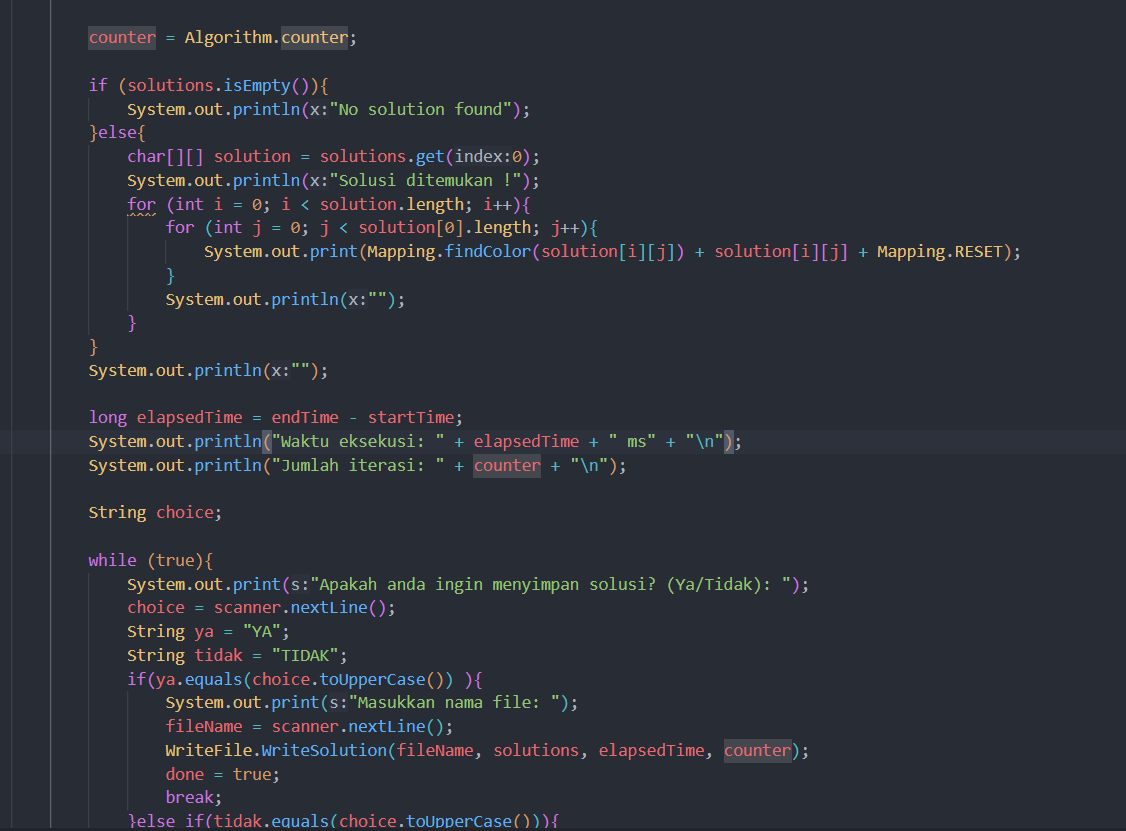
Projek ini terbagi menjadi beberapa file, yaitu:

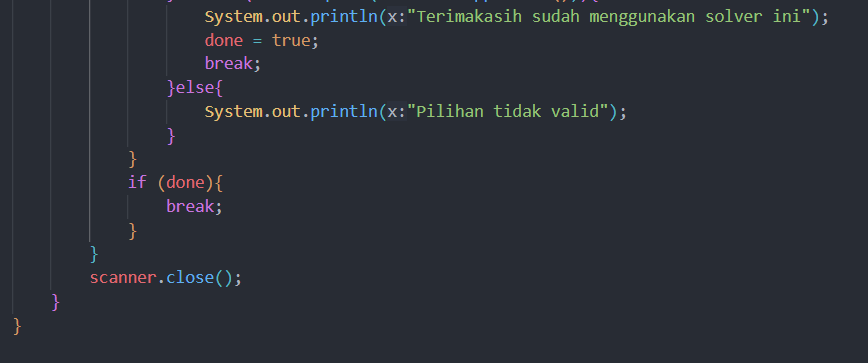
1. Main.java
2. Point.java
3. Piece.java
4. Grid.java
5. Algorithm.java
6. ReadFile.java
7. WriteFile.java
8. Mapping.java
9. Utility.java

Berikut source codenya:

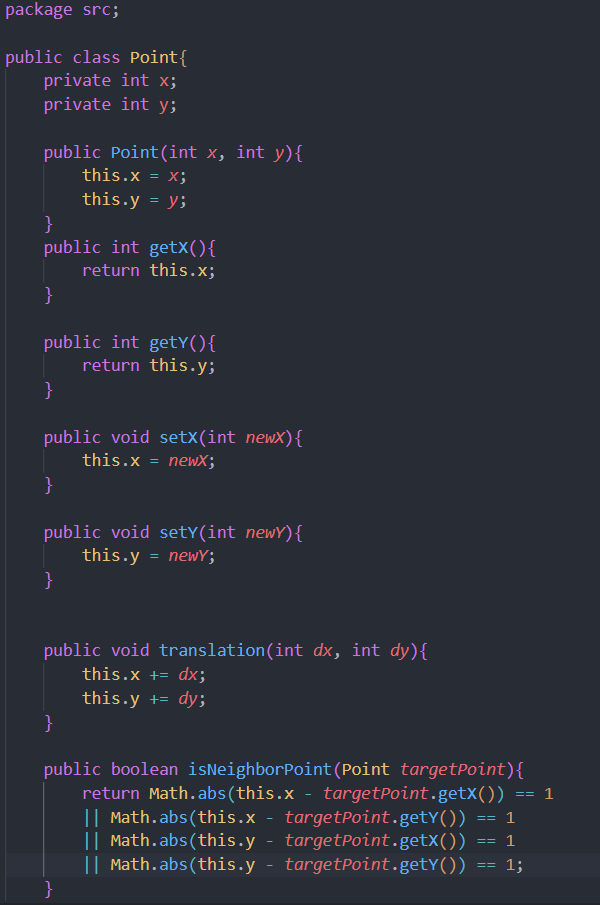
1. Main.java



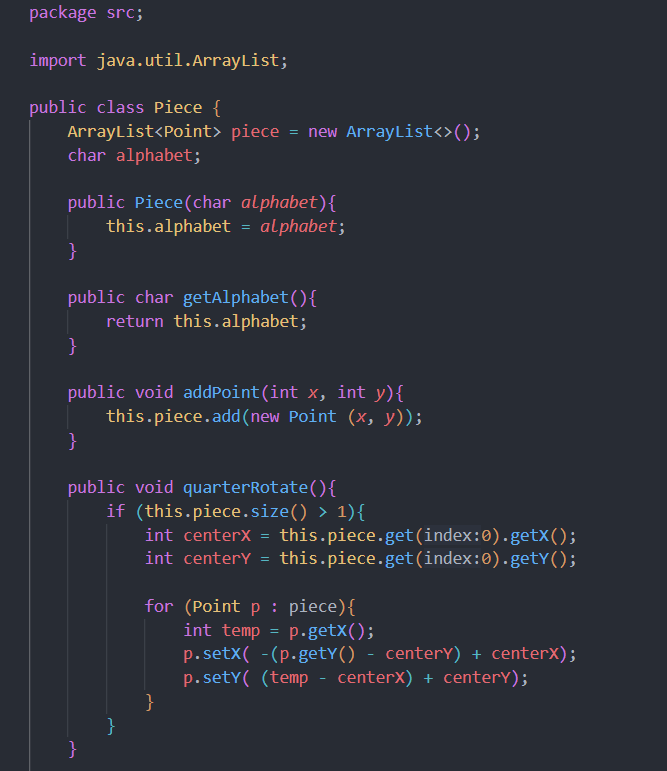


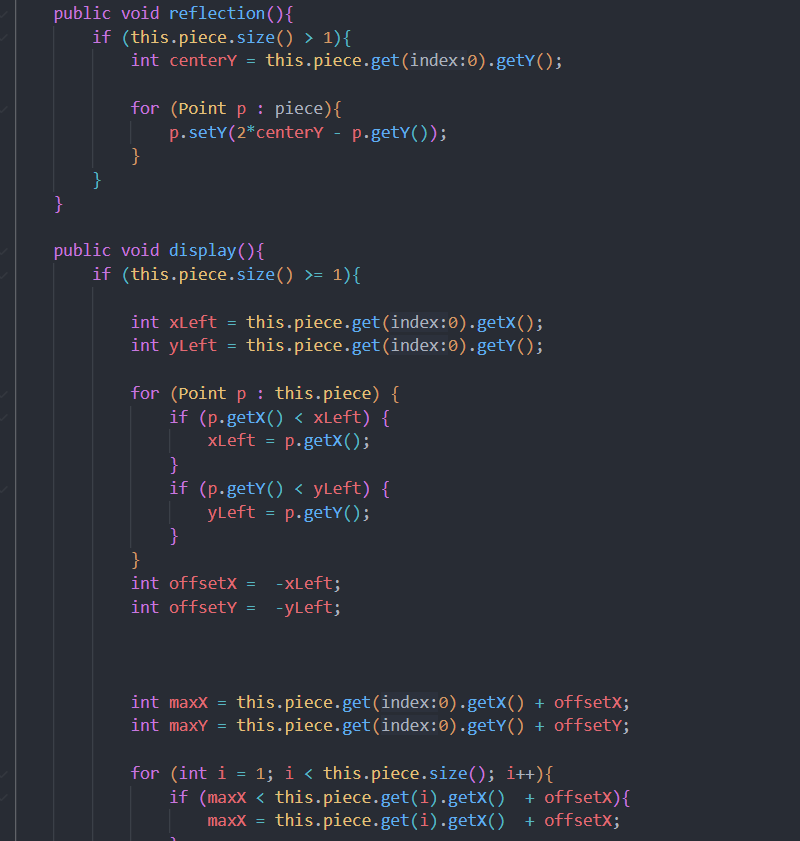


1. Point.java

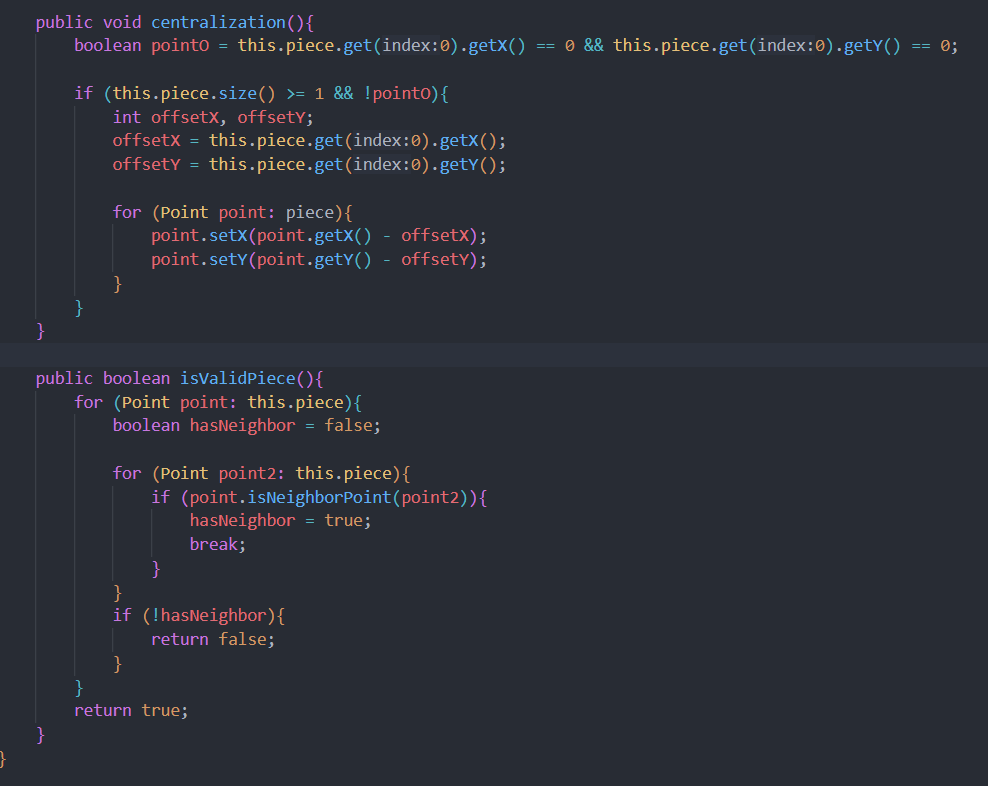


1. Piece.java



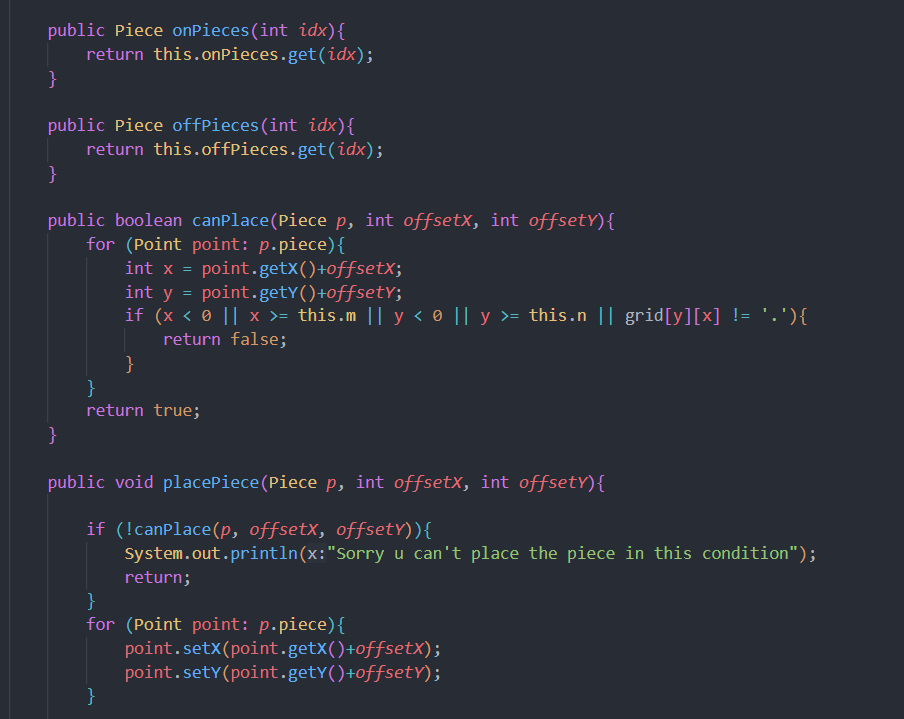


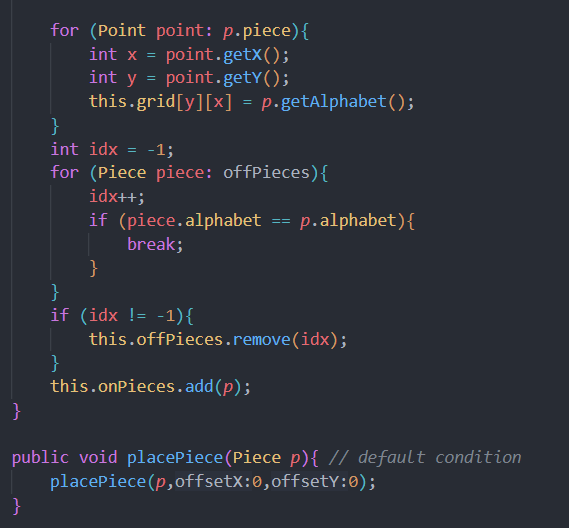




1. Grid.java

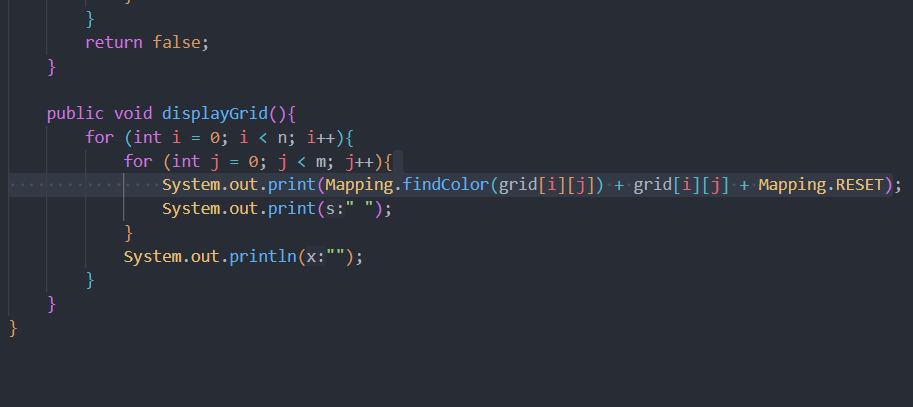




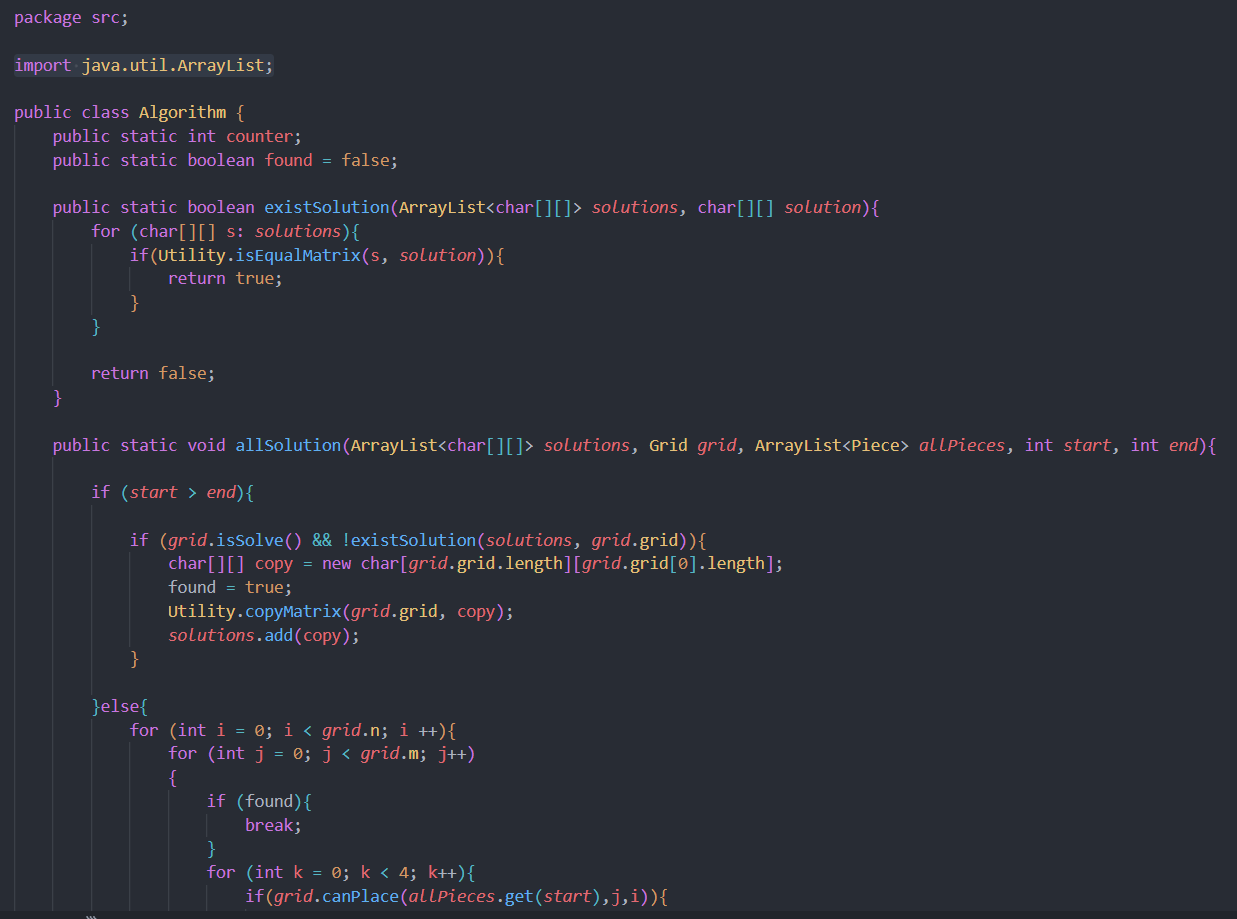


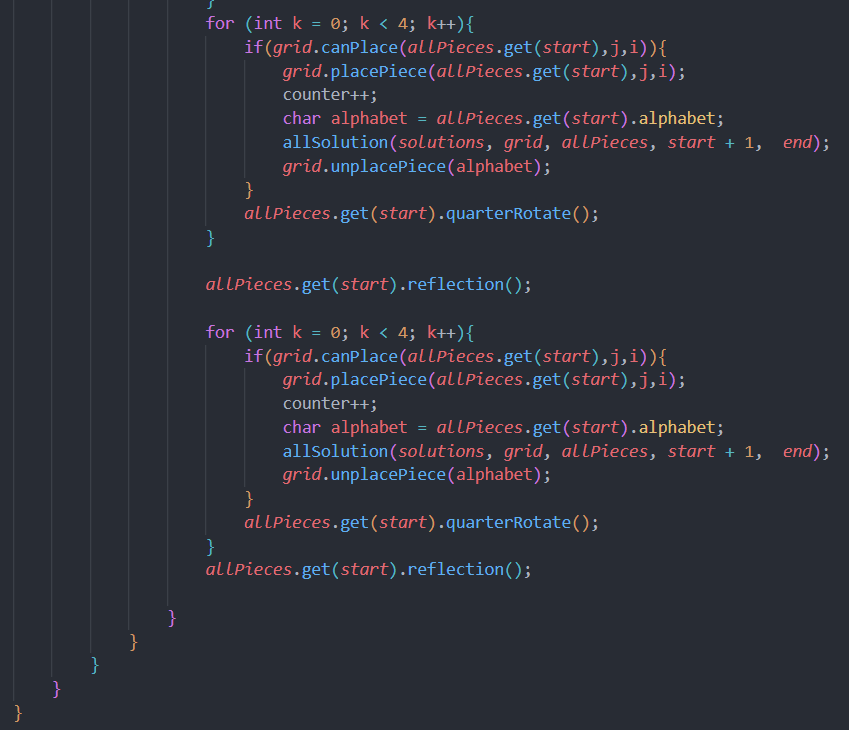






1. Algorithm.java



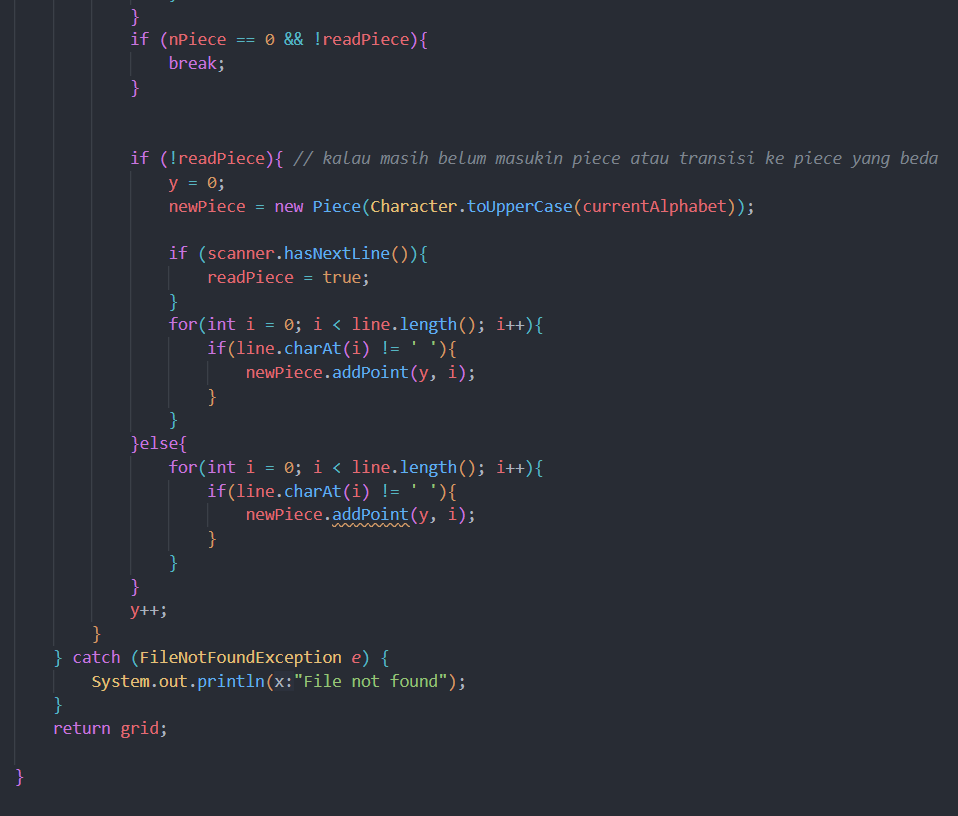


1. ReadFile.java

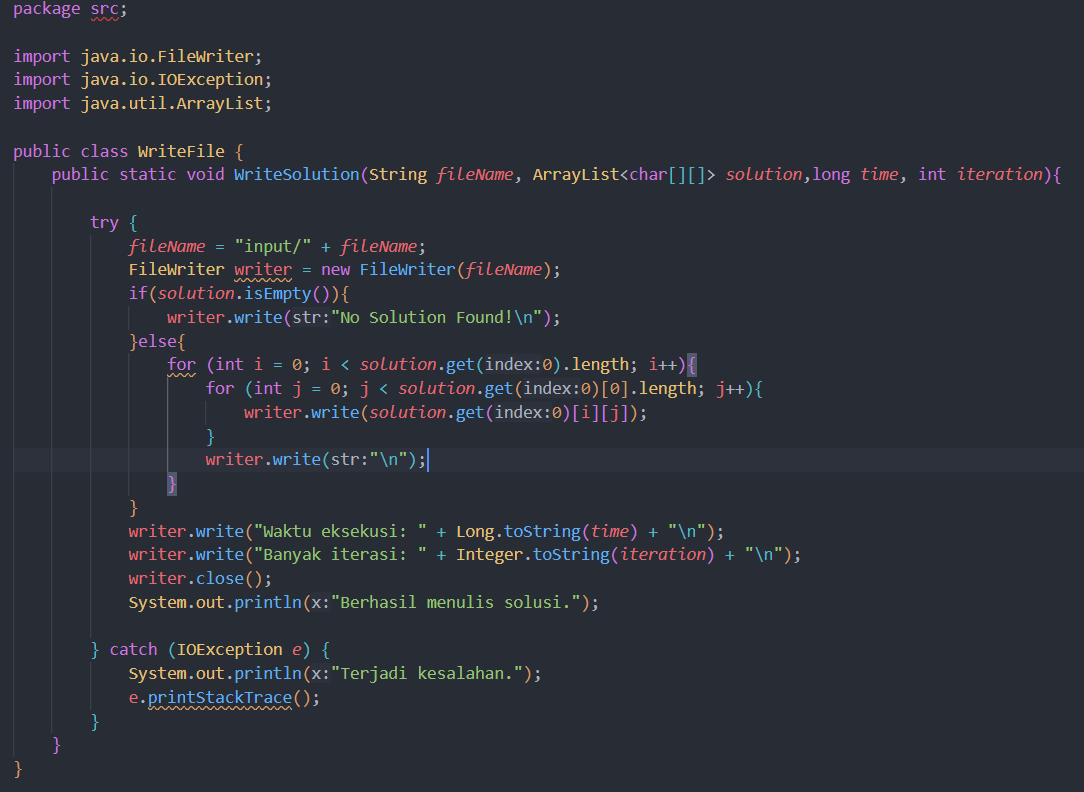




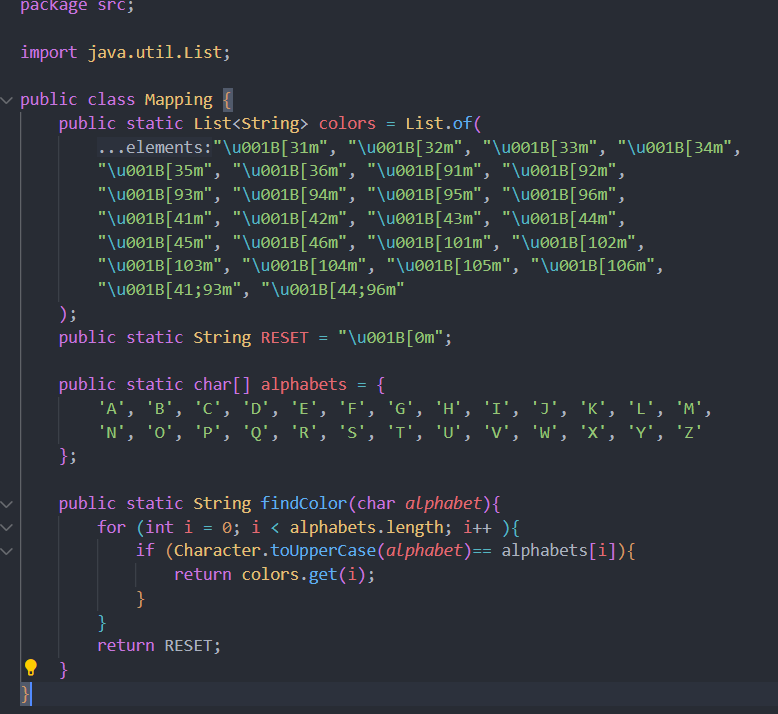




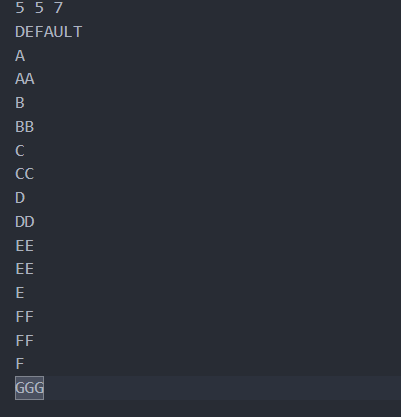
1. WriteFile.java

import java.util.Scanner;

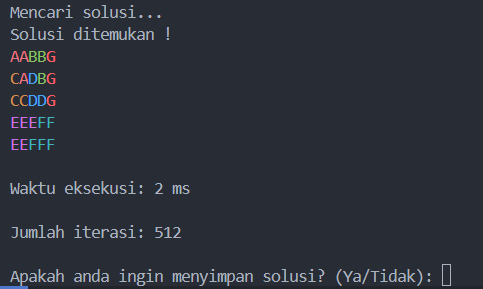
1. Mapping.java



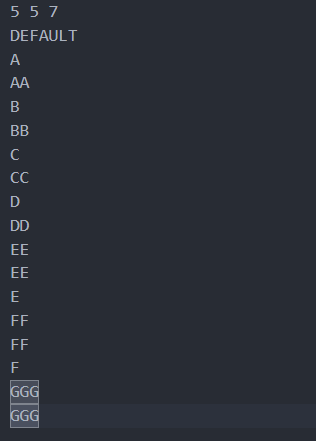
1. **Testing**
2. Input:



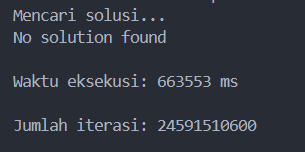
Output:



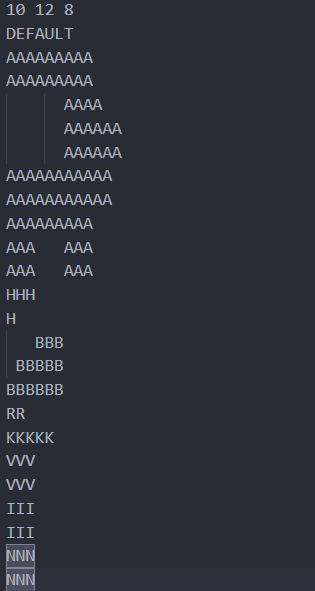
1. Input:



Output:



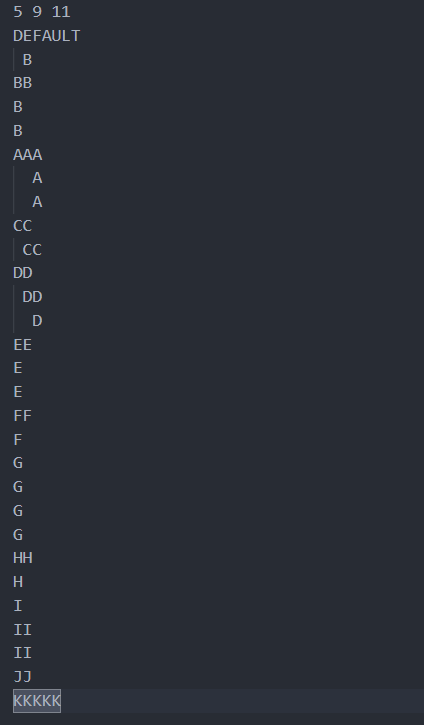
1. Input:



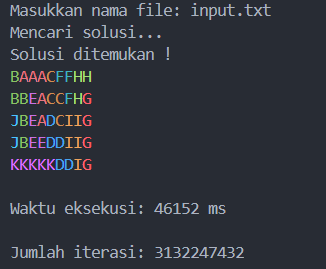
Output:



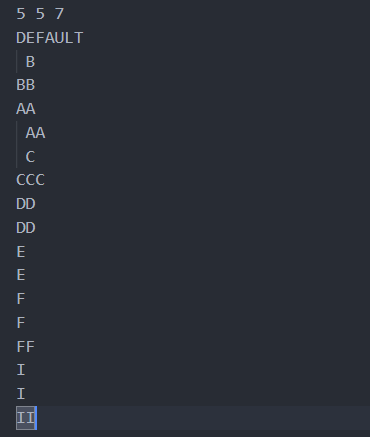
1. Input:



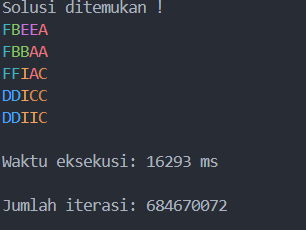
Output:



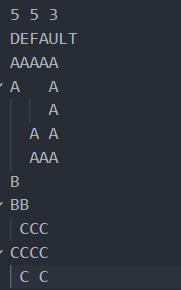
1. Input:



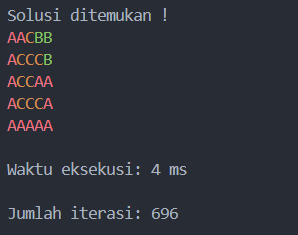
Output:



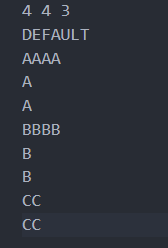
1. Input:



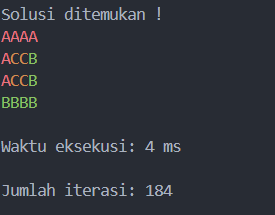
Output:



1. Input:



Output:



1. **Link Repository**

<https://github.com/AryoBama/Tucil1_13523088>

1. **Checklist**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Poin** | **Ya** | **Tidak** |
| 1 | Program berhasil dikompilasi tanpa kesalahan | ✓ |  |
| 2 | Program berhasil dijalankan | ✓ |  |
| 3 | Solusi yang diberikan program benar dan mematuhi aturan permainan | ✓ |  |
| 4 | Program dapat membaca masukan berkas .txt serta menyimpan solusi dalam berkas .txt | ✓ |  |
| 5 | Program memiliki Graphical User Interface (GUI) |  | ✓ |
| 6 | Program dapat menyimpan solusi dalam bentuk file gambar |  | ✓ |
| 7 | Program dapat menyelesaikankasus konfigurasi custom |  | ✓ |
| 8 | Program dapat menyelesaikan kasus konfigurasi Piramida (3D) |  | ✓ |
| 9 | Program dibuat oleh saya sendiri | ✓ |  |