

Dokumentacja projektu 05 – EJB-SERV

Program zawiera się w czterech plikach:

- IGameMonitor.java
- IMasterMind.java
- MasterMind.java
- MGame.java

W pliku MasterMind.java znajduje się implementacja mechanizmu rozwiązania.

Projekt zawiera 19 dodatkowych bibliotek wymaganych do stworzenia ArrayList, losowania liczb pseudolosowych za pomocą Random oraz do połączenia i pobrania danych z serwera aplikacyjnego Glassfish 4 jak i bibliotek wymaganych do poprawnego działania Servleta

- java.io.IOException;
 - java.io.PrintWriter;
 - javax.naming.Context;
 - javax.naming.InitialContext;
 - javax.naming.NamingException;
 - javax.servlet.ServletException;
 - javax.servlet.annotation.WebServlet;
 - javax.servlet.http.HttpServlet;
 - javax.servlet.http.HttpServletRequest;
 - javax.servlet.http.HttpServletResponse;
 - pl.jrj.game.IGameMonitor;
 - pl.jrj.game.IMasterMind;
 - java.util.ArrayList;
 - java.util.Random;
 - javax.ejb.Stateless;
 - javax.naming.Context;
 - javax.naming.InitialContext;
 - javax.naming.NamingException;
 - javax.ejb.Remote;
-

W pliku IGameMonitor.java znajdują się jeden interfejs:

- IGameMonitor

Interfejs **IGameMonitor** posiada prototypy metod których implementacja znajduje się w GameMonitor.java

W pliku IMasterMind.java znajdują się jeden interfejs:

- IMasterMind

Interfejs udostępniający metodę realizującą algorytm gry implementowanej w komponencie MasterMind.java

W pliku MasterMind.java znajduje się główny mechanizm realizujący algorytm gry.

Zawiera dwie klasy:

- GeneratorKombinacjiString - Główna rekurencyjna metoda do wypisywania wszystkich możliwych kombinacji Stringow.

- MasterMind - która implementuje następujący algorytm:

1. Tworzy listę wszystkich możliwych kombinacji: 32768
2. Losuje jeden z mozliwych kodow sposrod puli możliwych kombinacji
3. Wywołuje metodę verify i zapisuje zwrócony wynik do wVerify
4. Dla wszystkich kombinacji z listy sprawdza aby dawały taki sam wynik jaki został zwrócony dla kombinacji która była tzw. Strzałem za pomocą metody myVerify
5. Wszystkie kombinacje które zwróciły taki sam wynik trafiają do tymczasowej listy tmpKombinacje.
6. Czyścimy listę kombinacje.
7. Przepisujemy wszystkie elementy z listy tmpKombinacje do listy kombinacje
8. Czyścimy listę tmpKombinacje
9. Zwiększamy wartość pola iloscPodejsc.
10. Wykonujemy ten algorytm od punku 2 aż wynik nie będzie równy 50

W pliku MGame.java znajduje się mechanizm servleta łączący się z dwoma komponentami i wywołuje metodę rozwiązującą algorytm oraz zwraca wynik

KONIEC
