**Số chính phương**

**Server.java**

package sochinhphuong;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

public class Server {

public static void main(String[] args) {

try {

NumberProcessor processor = new NumberProcessorImpl();

Registry registry = LocateRegistry.createRegistry(1099);

registry.rebind("NumberProcessor", processor);

System.out.println("Server is running...");

} catch (Exception e) {

System.err.println("Server exception: " + e.toString());

e.printStackTrace();

}

}

}

**Client.java**

package sochinhphuong;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

public class Client {

public static void main(String[] args) {

try {

Registry registry = LocateRegistry.getRegistry("localhost", 1099);

NumberProcessor processor = (NumberProcessor) registry.lookup("NumberProcessor");

String numbers = "1,2,3,4,5,6,7,8,9,10,16";

String result = processor.findPerfectSquares(numbers);

System.out.println("Perfect squares: " + result);

} catch (Exception e) {

System.err.println("Client exception: " + e.toString());

e.printStackTrace();

}

}

}

**NumberProcessorImpl.java**

package sochinhphuong;

import java.rmi.RemoteException;

import java.rmi.server.UnicastRemoteObject;

public class NumberProcessorImpl extends UnicastRemoteObject implements NumberProcessor {

protected NumberProcessorImpl() throws RemoteException {

super();

}

@Override

public String findPerfectSquares(String numbers) throws RemoteException {

StringBuilder result = new StringBuilder();

String[] nums = numbers.split(",");

for (String num : nums) {

int n = Integer.parseInt(num);

double sqrt = Math.sqrt(n);

if (sqrt == Math.floor(sqrt)) {

result.append(n).append(",");

}

}

return result.toString();

}

}

**NumberProcessor.java**

package sochinhphuong;

import java.rmi.Remote;

import java.rmi.RemoteException;

public interface NumberProcessor extends Remote {

String findPerfectSquares(String numbers) throws RemoteException;

}