import java.util.List;

import java.util.ArrayList;

class TimeServer {

private List<Integer> clocks;

public TimeServer(List<Integer> clocks) {

this.clocks = clocks;

}

/\*\*

\* The function synchronizes all clocks in a list by adjusting each clock to the average time.

\*/

public void synchronizeClocks() {

int sum = 0;

int average;

// Calculate the sum of all clocks

for (int clock : clocks) {

sum += clock;

}

// Calculate the average clock time

average = sum / clocks.size();

// Adjust each clock to the average time

for (int i = 0; i < clocks.size(); i++) {

clocks.set(i, average);

}

}

public List<Integer> getClocks() {

return clocks;

}

// Entry point of the program

/\*\*

\* The function creates a list of clocks with their initial times, synchronizes them using the

\* Berkeley algorithm, and prints the actual and synchronized clock times.

\*/

public static void main(String[] args) {

// Create a list of clocks with their initial times

List<Integer> clocks = new ArrayList<>();

clocks.add(100);

clocks.add(200);

clocks.add(150);

clocks.add(180);

// Create a time server with the clocks

TimeServer timeServer = new TimeServer(clocks);

// Print the actual clock times before synchronization

System.out.println("Actual Clock Times:");

for (int clock : clocks) {

System.out.println(clock);

}

System.out.println("");

// Synchronize the clocks using the Berkeley algorithm

timeServer.synchronizeClocks();

// Get the synchronized clocks

List<Integer> synchronizedClocks = timeServer.getClocks();

// Print the synchronized clocks

System.out.println("Synchronized Clock Times:");

for (int clock : synchronizedClocks) {

System.out.println(clock);

}

}

}