**> recently played songs that can accommodate N songs per user, with a fixed initial capacity,**

**import java.util.\*;**

**public class RecentlyPlayedSongsStore {**

**private final int capacity;**

**private final Map<String, Deque<String>> store;**

**public RecentlyPlayedSongsStore(int capacity) {**

**this.capacity = capacity;**

**this.store = new HashMap<>();**

**}**

**public void add(String user, String song) {**

**Deque<String> songs = store.computeIfAbsent(user, k -> new LinkedList<>());**

**songs.addLast(song);**

**if (songs.size() > capacity) {**

**songs.removeFirst();**

**}**

**}**

**public List<String> getRecentlyPlayedSongsForUser(String user) {**

**Deque<String> songs = store.get(user);**

**return songs == null ? Collections.emptyList() : new ArrayList<>(songs);**

**}**

**}**