

ĐỀ THI CUỐI KỲ HÈ 2019

Môn: Tin học cơ sở 4

ĐỀ 1

Thời gian: 90 phút

Họ và tên:..... Mã sinh viên:.....

Bài 1 (6 test) **Homestay** ở nước ngoài được hiểu là bạn sẽ sống trong căn nhà của người bản địa, có thể sống chung và sinh hoạt như một thành viên trong gia đình. Phần mềm quản lý đặt phòng **homestay** gồm *Room.java* và *Management01.java*:

Lớp *Room.java* (phòng) với các thuộc tính:

- *String id* (mã phòng)
- *int capacity* (sức chứa)
- *int current* (số khách hiện tại)

Điều kiện: $current \leq capacity$

Hoàn thiện các phương thức sau:

- (1 test)

```
public int calAvaivable()
```

Trong lớp *Room*. Phương thức thực hiện tính số lượng khách du lịch có thể thêm vào phòng.

- (2 test)

```
public boolean canService(int nTourist, ArrayList<Room> rooms)
```

Trong lớp *Management01*. Phương thức thực hiện kiểm tra trong danh sách phòng (*rooms*) thì còn đủ chỗ để phục vụ *nTourist* khách du lịch hay không? Nếu có trả về giá trị *true*, trong trường hợp còn lại trả về giá trị *false*.

- (2 test)

```
public int calMinRoom(int nTourist, ArrayList<Room> rooms)
```

Trong lớp *Management01*. Phương thức thực hiện tính số lượng phòng ít nhất trong danh sách phòng (*rooms*) dùng để phục vụ *nTourist* khách du lịch. Trong trường hợp dùng tất cả các phòng có thể sử dụng mà không phục vụ hết khách du lịch thì trả về -1 ;

- (1 test)

```
public int calMinNewRoom(int nTourist, ArrayList<Room> rooms)
```

Trong lớp *Management01*. Phương thức thực hiện tính số lượng phòng **trống** ít nhất trong danh sách phòng (*rooms*) dùng để phục vụ *nTourist* khách du lịch. Trong trường hợp dùng tất cả các phòng có thể sử dụng mà không phục vụ hết khách du lịch thì trả về -1 . Phòng **trống** là phòng chưa có khách du lịch thuê (*current* = 0)

Bài 2 (4 test) Cho trước tệp *IncCompress.java*. Hoàn thiện các phương thức sau:

- (1 test)

```
public static String[] readTokens(String fileName)
```

Phương thức yêu cầu đọc mảng các *token* từ tệp *fileName*. Mỗi *token* được định nghĩa là một chuỗi các ký tự liên nhau không bao gồm dấu cách và ký tự trắng. Dữ liệu trong tệp được cho như sau:

- Dòng đầu tiên là số nguyên dương n , số lượng *token* của mảng.
- Dòng tiếp theo là n *token*, hai *token* cách nhau một dấu cách.

- (2 test)

```
public static boolean canIncCompress(char[] a, char[] b, int k)
```

Phương thức phép kiểm tra xem k ký tự *cuối cùng* của mảng a có khớp với k ký tự *đầu tiên* của mảng b hay không? Nếu khớp, phương thức trả về giá trị *true*, trong trường hợp còn lại trả về giá trị *false*.

Ví dụ:

- Với $a = \{x, x, 1, 2, 3\}$, $b = \{1, 2, 3, z\}$, $k = 3 \Rightarrow \text{canIncCompress}(a, b, k) = \text{true}$
- Với $a = \{x, x, 1, 2, 3\}$, $b = \{1, 2, z, z\}$, $k = 3 \Rightarrow \text{canIncCompress}(a, b, k) = \text{false}$

- (1 test)

```
public static char[] incCompress(char[] a, char[] b, int k)
```

Hoàn thiện phương thức trả về một mảng ký tự kết quả nén của mảng a và mảng b với độ nén k . Kiểm tra k ký tự *cuối cùng* của mảng a có khớp với k ký tự *đầu tiên* của mảng b hay không? Nếu khớp trả về kết quả chuỗi ký tự gồm các ký tự mảng a nối với các ký tự mảng b khi bỏ đi k ký tự đầu tiên của mảng b . Trong trường hợp còn lại trả về mảng ký tự kết quả nối của hai mảng a và mảng b .

Ví dụ:

- Với $a = \{x, x, 1, 2, 3\}$, $b = \{1, 2, 3, z\}$, $k = 3 \Rightarrow \text{incCompress}(a, b, k) = \{x, x, 1, 2, 3, z\}$
- Với $a = \{x, x, 1, 2, 3\}$, $b = \{1, 2, z, z\}$, $k = 3 \Rightarrow \text{incCompress}(a, b, k) = \{x, x, 1, 2, 3, 1, 2, z, z\}$

Chú ý:

Sinh viên đọc cẩn thận các lưu ý sau đây để tránh mắc các lỗi khi làm bài

- Sinh viên chỉ cần hoàn thiện các phương thức được yêu cầu, tuyệt đối không chỉnh sửa nguyên mẫu các phương thức này, chỉ viết thêm vào phần nội dung phương thức.
- Sinh viên có thể viết thêm các phương thức phụ trợ khác nếu cần, tuy nhiên tránh đặt trùng tên các phương thức đã cho.
- Mỗi phương thức được chấm điểm độc lập, đúng phương thức nào sẽ được điểm phương thức đó, tuy nhiên cần đảm bảo bài làm có thể biên dịch và chạy được. Các bài có lỗi biên dịch, hoặc không chạy được sẽ nhận điểm 0.
- Các phương thức sinh viên chưa làm đến, hoặc không hoàn thiện được thì để lại như nguyên mẫu ban đầu, không chỉnh sửa, thêm bớt, tránh không biên dịch được.
- Khi làm bài sinh viên có thể tự nộp bài và chấm điểm trên một bộ dữ liệu mẫu, kết quả này chỉ là kết quả tham khảo, bài làm của sinh viên sẽ được chấm lại trên bộ dữ liệu khác và thông báo sau.