ĐỀ THI KẾT THÚC HỌC PHẦN

THỜI GIAN LÀM BÀI: 120 PHÚT

# Câu 1 – Mẫu thiết kế State

Cho class sau

**public class** Utils {

**public** **static** **void** rung() {

// thực hiện rung thiết bị với mức độ cho trước

System.out.println(“Rung điện thoại”);

}

**public** **static** **void** phátÂmThanh(int âmLượng, int sốLần) {

// thực hiện phát âm thanh BEEP BEEP với âm lượng và  
// số lần cho trước

System.out.println(“BEEP-BEEP”);

}

}

Hãy áp dụng mẫu thiết kế State để cài đặt lớp ĐiệnThoại, cho phép thiết lập thiết bị ở một trong các trạng thái Yên Lặng, Chuẩn, Ngoài Trời theo đó:

| **Trạng thái** | **Khi có cuộc gọi đến  (on calling)** | **Khi có tin nhắn đến  (on SMS received)** | **Khi có sự kiện hệ thống  (System notification)** |
| --- | --- | --- | --- |
| Yên Lặng | Rung 1 lần | Không làm gì | Không làm gì |
| Chuẩn | Rung 1 lần, sau đó  Phát âm thanh âm lượng 80, 3 lần | Phát âm thanh âm lượng 60, 1 lần | Phát âm thanh âm lượng 50, 1 lần |
| Ngoài trời | Rung 3 lần, sau đó Phát âm thanh âm lượng 100, 5 lần | Phát âm thanh âm lượng 100, 1 lần | Phát âm thanh âm lượng 90, 1 lần |

Sử dụng các lớp đã thiết kế, viết chương trình để minh hoạ:

**public class** ChuongTrinh {

**public static void** main(String[] args)) {

// khởi tạo đối tượng điện thoại với trạng thái ban đầu là Chuẩn

ĐiệnThoại đt = new ĐiệnThoại(…);

// 1.a) thiết lập trạng thái Ngoài trời cho **dt**

// 1.b) thử minh hoạ khi có tin nhắn đến cho **dt**

// 1.c) thử minh hoạ khi có sự kiện hệ thống cho **dt**

// 1.d) thử minh hoạ khi có cuộc gọi đến cho **dt**

// 2.a) thiết lập trạng thái Yên lặng cho **dt**

// 2.b) thử minh hoạ khi có tin nhắn đến **dt**

// 2.c) thử minh hoạ khi có cuộc gói đến cho **dt**

}

}

**Yêu cầu**

1) Viết code cho các class cần thiết. Chú ý, tất cả các code đều được đặt trong package với tên **vn.edu.husc.s\_mãSinhViên** tương ứng với mã sinh viên của anh/chị, chẳng hạn vn.edu.husc.s\_t1021234

2) Thực hiện chạy chương trình **ChuongTrinh** theo yêu cầu và nộp hình ảnh minh hoạ

# Câu 2 – Mẫu thiết kế Iterator

Cho các interface sau đây:

// API của iterator

**interface** IteratorInteger {

// khởi tạo bộ duyệt

**void** start();

// cho biết còn phần tử kế tiếp hay không

boolean hasNext();

// lấy phần tử ở vị trí hiện tại và di chuyển đến vị trí tiếp theo

Integer next();

}

// API để tạo ra iterator

**interface** IContainer {

// Bộ duyệt qua các số dương của đối tượng chứa

IteratorInteger getIteratorSoDuong();

// Bộ duyệt qua tất cả các phần tử của đối tượng chứa

IteratorInteger getIteratorAll();

}

Cho file văn bản NUMBERS.TXT chứa danh sách các số nguyên, mỗi số nằm trên một dòng. Hãy sử dụng hai API nói trên để cài đặt class MyList và các class cần thiết khác theo gợi ý sau:

**class** MyList **implements** IContainer {

**private** List<Integer> a;

**public** MyList(String filename) {

// TODO: khởi tạo danh sách a gồm các số chứa trong file filename

}

**public** IteratorInteger getIteratorSoDuong() {

// Cài đặt class IteratorDuyetQuaSoDuong

**return new** IteratorDuyetQuaSoDuong(this.a);

}

**public** IteratorInteger getIteratorAll() {

// Cài đặt class IteratorAll

**return new** IteratorAll(this.a);

}

}

**class** IteratorDuyetQuaSoDuong **implements** IteratorInteger {

...

}

**class** IteratorAll **implements** IteratorInteger {

...

}

**class** ChuongTrinh2 {

**public static void** main(String[] args) {

// 2.a) Khởi tạo instance của MyList

// 2.b) In danh sách tất cả các phần tử của danh sách (sử dụng iterator đã cài đặt)

// 2.c) In danh sách tất cả các phần tử dương của danh sách (sử dụng iterator đã cài đặt)

}

}

**Yêu cầu**

1) Viết code cho các class cần thiết. Chú ý, tất cả các code đều được đặt trong package với tên **vn.edu.husc.s\_mãSinhViên** tương ứng với mã sinh viên của anh/chị, chẳng hạn vn.edu.husc.s\_t1021234

2) Thực hiện chạy chương trình **ChuongTrinh2** theo yêu cầu và nộp hình ảnh minh hoạ.