

Lời giải

Tóm tắt đề: Xây dựng một chương trình đơn giản, quản lý một Store bán hàng, trong đó hai cấu trúc dữ liệu chính bao gồm:

- Linked list: dùng để lưu trữ danh sách mặt hàng, mỗi mặt hàng có 3 thông tin gồm tên, số lượng, và giá bán
- Queue: dùng để biểu diễn các lượt order mua hàng, mỗi lượt mua hàng gồm có 2 thông tin gồm tên và số lượng.

Các yêu cầu của đề bài xoay quanh các yêu cầu như thêm dữ liệu, mua hàng, tính tiền, v.v. Về bản chất, tương ứng với các thao tác: thêm, sửa, xoá, duyệt của list và queue.

Lời giải cho các yêu cầu như sau:

F1() – 2.5 điểm

Yêu cầu: Thêm dữ liệu vào list và queue

- AddLast:

```
Node node = new Node(new Fruit(type, amount, price));
if (this.isEmpty()) {
    this.tail = this.head = node;
}
this.tail.next = node;
this.tail = node;
```

- EnQueue: (hoàn toàn tương tự AddLast, chỉ đổi tên head=>front, tail=>rear)

```
Node node = new Node(new Fruit(type, amount));
if (this.isEmpty()) {
    this.rear = this.front = node;
}
this.rear.next = node;
this.rear = node;
```

F2() – 2.5 điểm

Yêu cầu: (1) thực hiện dequeue để lấy ra lượt order đầu tiên, sau đó (2) thực hiện hoạt động mua hàng (tìm kiếm món hàng trong list các mặt hàng, nếu tìm thấy thì kiểm tra số lượng, nếu số lượng hàng trong list đủ (lớn hơn hoặc bằng order) thì thực hiện mua hàng: cập nhật lại số lượng và trả về số tiền bán được)

- DeQueue():

```

Fruit dequeue() {
    if (front == null) {
        return null;
    }
    Fruit tmp = front.info;
    front = front.next;
    return tmp;
}

```

- Mua hàng và trả về số tiền mua hàng:

```

int purchase(Fruit t) throws Exception {
    int r = 0;
    Node x = FList.head;
    while (x != null) {
        if (t.getType().equals(x.info.getType())) {
            if (t.getAmount() <= (x.info.getAmount())) {
                x.info.setAmount(x.info.getAmount() - t.getAmount());
                r = t.getAmount() * x.info.getPrice();
                break;
            }
        }
        x = x.next;
    }
    return r;
}

```

Search

Kiểm tra số lượng

Cập nhật lại số lượng

Tính tiền

- Hàm xử lý chính (trong hàm f2())

```

Fruit t = RQueue.dequeue();
if (t != null) {
    this.purchase(t);
}

```

F3() – 2.5 điểm

Yêu cầu: tương tự như câu F2, nhưng thực hiện cho toàn bộ queue. Nghĩa là lần lượt DeQueue tất cả các phần tử trong queue, sau đó thực hiện hành động mua bán.

```

Fruit t = RQueue.dequeue();
while (t != null) {
    this.purchase(t);
    t = RQueue.dequeue();
}

```

F4() – 2.5 điểm

Yêu cầu: tính tổng tiền thu được sau khi phục vụ cho toàn bộ queue

```

int S = 0;
Fruit t = RQueue.dequeue();
while (t != null) {
    S += this.purchase(t);
    t = RQueue.dequeue();
}

```

⇒ Toàn bộ đáp án chưa đến 50 dòng code 😊