Name: _____   Date:_____

Student #: _____

COMP3021: Java Programming In-class Exercise

Your task is to rewrite the Queue class using generics with the following
requirements:

1. The queue must only handle citrus fruits such as orange, lemon, lime, etc,
assuming we have defined classes such as Citrus with subclasses Orange, Lemon,
Lime. Please note that we also have class Fruit as the super class of Citrus. Apple
is a subclass of Fruit.

2. The two methods "intake" and "toVector" must handle polymorphic parameters
appropriately, given all the class types defined above.

```java
class Queue<E> {
    Vector<E> v = new Vector<E>();
    public void enqueue(E o) {
        v.add(o);
    }

    public E dequeue() {
        if(v.size()==0) return null;
        E o = v.get(v.size()-1);
        v.remove(o);
        return o;
    }

    public void intake(Queue<E> temp) {
        E o = temp.dequeue();
        while(o!=null){
            enqueue(o);
            o = temp.dequeue();
        }
    }

    public void toVector(Vector<E> temp){
        for(E o:v)
            temp.add(o);
    }
}
```

Solution:

```java
class Queue<E extends Citrus> {
    Vector<E> v = new Vector<E>();
    public void enqueue(E o) {
        v.add(o);
    }

    public E dequeue() {
        if(v.size()==0) return null;
        E o = v.get(v.size()-1);
        v.remove(o);
        return o;
    }

    public void intake(Queue<? extends E> temp) {
        E o = temp.dequeue();
        while(o!=null){
            v.add(o);
            o = temp.dequeue();
        }
    }

    public void toVector(Vector<? super E> temp){
        for(E o:v)
            temp.add(o);
    }
}
```