**Practice:** Redesign the *Progression* class to be abstract and generic producing a sequence of values of generic type T, and support a single constructor that accepts an initial value.

class TestDemo {

public static void main(String[] args){

Progression<Integer> a = new My<Integer>(10);

a.printProgression(6);

}

}

abstract class Progression<T> {

protected T current;

public Progression() { this.getInitial(); }

protected abstract T getInitial();

public T nextValue() {

T answer = current;

return answer;

}

public void printProgression(int n) {

System.out.print(nextValue());

for(int j = 1; j < n;j++)

System.out.print(" " + nextValue());

System.out.println();

}

}

class My<T> extends Progression<T>{

public My(T ini){current = ini;}

protected T getInitial(){return current;}

}