**// part a**

public interface NumberGroup

{

public boolean contains(int n);

}

**// part b**

public class Range implements NumberGroup

{

private int minimum;

private int maximum;

public Range(int min, int max)

{

minimum = min;

maximum = max

}

public boolean contains(int n)

{

if(n >= min && n <= max)

return true

else

return false;

}

}

**// alternative solution for part b**

public class Range implements NumberGroup

{

private int[] numbers;

public Range(int min, int max)

{

numbers = new int[max - min + 1];

for(int i = 0; i < numbers.length; i++)

{

numbers[i] = min + i;

}

}

public boolean contains(int n)

{

for(int i = 0; i < numbers.length; i++)

{

if(numbers[i] == n)

return true;

}

return false;

}

}

**// part c**

public boolean contains(int n)

{

for(int i = 0; i < groupList.size(); i++)

{

if(groupList.get(i).contains(n) == true)

return true;

}

return false;

}