

Hints for Unit 5 Test Project:

Topic #1: Integer.MAX_VALUE & Integer.MIN_VALUE

MAX_VALUE is a constant of the Integer class holding the maximum value an int can have: $2^{31} - 1$

MIN_VALUE is a constant of the Integer class holding the minimum value an int can have: -2^{31}

These values are most commonly used when trying to determine the minimum or maximum integer in an array of integers, where the actual integral values stored in the array are unclear.

For Example:

```
int minNum = Integer.MAX_VALUE;
int maxNum = Integer.MIN_VALUE;

for (int i=0; i<ary.length; i++)
{
    if (ary[i] > maxNum)
        maxNum = ary[i];
    if (ary[i] < minNum)
        minNum = ary[i];
}
```

Integer.MAX_VALUE or Integer.MIN_VALUE might also replace a value in an array to guarantee that location within the array will not be processed a second time.

Topic #2: Determining the order you should visit different cities based on their distances apart.

Suppose the following chart represents the distances various Iowa cities are from each other:

	Iowa City	Ames	Waterloo	Ottumwa	Burlington
Iowa City	0	121	81	83	82
Ames	121	0	96	116	190
Waterloo	81	96	0	117	154
Ottumwa	83	116	117	0	77
Burlington	75	190	154	77	0

Write the code that would determine and return the index of the city closest to Iowa City.

```
String[] cities = {"Iowa City", "Ames", "Waterloo", "Ottumwa", "Burlington"};
int[][] distances = {{0, 121, 81, 83, 82},
                    {121, 0, 96, 116, 190},
                    {81, 96, 0, 117, 154},
                    {83, 116, 117, 0, 77},
                    {75, 190, 154, 77, 0}};
```

```
iMin = Integer.MIN_VALUE;
minDist = Integer.MAX_VALUE;

for (int i=0; i<cities.length; i++)
{
    if (i != 0 && distances[0][i] < minDist)
    {
        iMin = i;
        minDist = distances[0][i];
    }
}

return iMin;
```

Next, write the code that makes sure no other city can claim the city at iMin as its closest city.

```
for (int i = 0; i<distances.length; i++)
{
    distances[i][iMin] = Integer.MAX_VALUE;
}
```