1. **Create** a two-dimensional array which has 20 rows and 5 columns.

**(5 Marks)**

2. **Initialise** each element in the array to a random value in the range 1 – 10. Insure that each random number only appears once in every row and that only ODD numbers are stored at ODD column indexes in the array and that EVEN numbers are stored at EVEN column indexes (including 0) in the array.

**(35 Marks)**

3. **Print** the array to the screen (in columns of 5).

**(10 Marks)**

4. You must then **raise each number in the array to the power of the number that follows it** (succeeds it) in the array. The last number in the array (which has no succeeding number) must be raised to the power of the first number in the array.

**(40 Marks)**

5. **Print** the array to the screen (in columns of 5).

**(10 Marks)**

The output for the program should look something like the following:

```
run:
Orginal Array
4	5	8	3	6
8	3	6	7	4
6	9	2	7	8
2	7	6	9	4
2	7	8	3	6
6	3	10	5	4
4	3	2	5	10
4	1	10	9	2
8	5	6	3	10
10	3	6	9	8
6	5	10	3	2
8	5	2	7	6
8	9	6	1	4
6	7	2	3	10
8	9	4	1	2
4	5	2	7	8
2	9	6	5	8
8	7	10	9	6
10	7	8	1	2
8	9	6	5	4
```

Each Value Raised to the Power of its Successor

| | | | | |
|---|---|---|---|---|
| 1024 | 390625 | 512 | 729 | 1679616 |
| 512 | 729 | 279936 | 2401 | 4096 |
| 10077696 | 81 | 128 | 5764801 | 64 |
| 128 | 117649 | 10077696 | 6561 | 16 |
| 128 | 5764801 | 512 | 729 | 46656 |
| 216 | 59049 | 100000 | 625 | 256 |
| 64 | 9 | 32 | 9765625 | 10000 |
| 4 | 1 | 1000000000 | 81 | 256 |
| 32768 | 15625 | 216 | 59049 | 10000000000 |
| 1000 | 729 | 10077696 | 43046721 | 262144 |
| 7776 | 9765625 | 1000 | 9 | 256 |
| 32768 | 25 | 128 | 117649 | 1679616 |
| 134217728 | 531441 | 6 | 1 | 4096 |
| 279936 | 49 | 8 | 59049 | 100000000 |
| 134217728 | 6561 | 4 | 1 | 16 |
| 1024 | 25 | 128 | 5764801 | 64 |
| 512 | 531441 | 7776 | 390625 | 16777216 |
| 2097152 | 282475249 | 1000000000 | 531441 | 60466176 |
| 10000000 | 5764801 | 8 | 1 | 256 |
| 134217728 | 531441 | 7776 | 625 | 256 |

**It may prove difficult to align the (very large) numbers in to columns the second time you print the array. This is ok (as long as each row has five columns) – we will shortly look at ways to solve this problem in class.*

**Use Math.pow(x,y) to help you with part 4 of this exercise. For example:*

```
double d1 = Math.pow(4,2); //d1 will have a value of 16.0

double d2 = Math.pow(3,3); //d2 will have a value of 27.0

double d3 = Math.pow(8,4); //d3 will have a value of 4096.0
```

In other words, Math.pow(x,y) will raise x to the power of y and return the answer as a double.