# Evaluating e^x

The series expansion of $e^x$ is given by:

$1 + x + x^2/2! + x^3/3! + x^4/4! + \ldots\ldots$

Evaluate $e^x$ for given values of $x$ by using the above expansion *for the first* $10$ *terms*.

**Input Format**

The first line contains an integer $N$, the number of test cases.
$N$ lines follow. Each line contains a value of $x$ for which you need to output the value of $e^x$ using the above series expansion. These input values have exactly $4$ decimal places each.

**Output Format**

Output $N$ lines, each containing the value of $e^x$, computed by your program.

**Constraints**

$1 <= N <= 50$
$-20.00 <= x <= 20.00$
*Var*, *Val* in Scala and *def* and *defn* in Clojure are blocked keywords. The challenge is to accomplish this without either mutable state or direct declaration of local variables.

**Sample Input**

```
4
20.0000
5.0000
0.5000
-0.5000
```

**Sample Output**

```
2423600.1887
143.6895
1.6487
0.6065
```

**Explanation**

The output has the computed values of $e^x$ corresponding to each test case. They are correct up to $4$ decimal places and on separate lines.

**Scoring**

All test cases carry an equal weight in the final score. For your solution to pass a given test case, all the values of $e^x$ computed by you must be within $+/-0.1$ of the expected answers. This tolerance level has been kept to account for slightly different answers across different languages.