

Points on a Sphere

Imagine a number of identically charged weightless dimensionless particles placed on the surface of a ball. They will instantly reach a state of equilibrium (a stable state of minimum energy), becoming distributed fairly evenly all round the sphere.

You probably won't be surprised to hear that Byteland has a sadly distorted electrostatic field, and the energy of the system is not governed by ordinary laws. Instead, it is inversely proportional to the distance between the closest pair of charges on the sphere.

Please help the charges find positions in which they will feel as comfortable as possible. Charges should be regarded as points in 3D space, located on the surface of the unitary sphere (with center (0,0,0) and a radius of 1).

Input

An integer t denoting the number of test cases ($t \leq 10$), followed by t test cases, each consisting of a line with a single integer n - the number of points on the sphere ($2 \leq n \leq 1000$).

Output

For each test case, output n lines consisting of three floating point numbers, corresponding to the x y z coordinates of successive points.

Scoring

The score of your program is the total of scores awarded for individual test cases.

For each test case you will receive $n \cdot d$ points, where d denotes the minimum distance between the closest pair of points in the solution output by your program. If you place some of the points further than 10^{-5} from the surface of the sphere, your solution will be regarded as incorrect.

Example

For the sample input:

```
1
2
```

a program outputting:

```
0.0 0.0 1.0
0.0 1.0 0.0
```

will receive 2.828 points.