Problem B
Aliquot Sum

Time limit: 8 seconds

Memory limit: 2048 megabytes

Problem Description

A divisor of a positive integer n is an integer d where $m = \frac{n}{d}$ is an integer. In this problem, we define the aliquot sum $s(n)$ of a positive integer n as the sum of all divisors of n other than n itself. For examples, $s(12) = 1 + 2 + 3 + 4 + 6 = 16$, $s(21) = 1 + 3 + 7 = 11$, and $s(28) = 1 + 2 + 4 + 7 + 14 = 28$.

With the aliquot sum, we can classify positive integers into three types: abundant numbers, deficient numbers, and perfect numbers. The rules are as follows.

1. A positive integer x is an abundant number if $s(x) > x$.
2. A positive intewer y is a deficient number if $s(y) < y$.
3. A positive integer z is a perfect number if $s(z) = z$.

You are given a list of positive integers. Please write a program to classify them.

Input Format

The first line of the input contains one positive integer T indicating the number of test cases. The second line of the input contains T space-separated positive integers $n_1, \dots, n_T$.

Output Format

Output T lines. If n_i is an abundant number, then print `abundant` on the i-th line. If n_i is a deficient number, then print `deficient` on the i-th line. If n_i is a perfect number, then print `perfect` on the i-th line.

Technical Specification

- $1 \leq T \leq 10^6$
- $1 \leq n_i \leq 10^6$ for $i \in \{1, 2, \dots, T\}$.

Sample Input 1

```
3
12 21 28
```

Sample Output 1

```
abundant
deficient
perfect
```

Almost blank page