Assignment #2: Divide & conquer

to March 17th, 23:59

CSED331 (Hee-kap Ahn)

Jongmin Choi, Sunghwan Kim

Contents

Problem 1. n^k

Problem Statement

Get two integers n and k and compute n^k mod 201703040317.

Input Statement

First line contains t which is the number of testcases.
Each 2nd,..., $t+1$-th line contains a test case consisting of two integers n and k ($0 \leq n, k < 2^{32}$).

Output Statement

The program should print n^k mod 20170317 for each input.
Each output should be separated by empty line.

Input Example

```
4
2 4
5 10
10 20
1 2147483647
```

Output Example

```
16
9765625
2451835
1
```

Problem 2. Closest pair

Problem Statement

Closest pair is one of the important application on geometric problems. In this assignment, you need to implement program that find closest pair among given set of points.

There are two kind of well known function to compute the distance of two points: The Euclid and Manhattan distance. In this assignments, we use the Manhattan distance, which is simple sum of the distance bewteen each dimension. For example, the distance between two points $x = (1, 1, 6)$ and $y = (5, -1, 0)$ is $|1 - 5| + |1 - (-1)| + |6 - 0| = 12$. We define the Manhattan distance $D(p, q)$ of two points $p = (p_1, p_2, p_3)$ and $q = (q_1, q_2, q_3)$ in 3-dimension as $D(p, q) = \sum_{i=1}^{3} |p_i - q_i|$.

Your goal is to implement a program that find closest pair over 3-dimensional space using divide and conquer way.

Input Statement

First line contains an integer t, which is the number of test cases.
First line of each test case has an integer $N(\leq 30000)$.
Each 2nd,..., $(N + 1)$-th line contains three numbers which describe a point.
Each number is separated by a space, and each test case is separated by a line.

Output Statement

For each test case, prints out the minimum distance of the problem.
Each test case should be separated by a line.

Input Example

```
2
3
3 0 0
2 0 0
1 0 0
3
5 7 0
10 2 0
11 3 0
```

Output Example

```
1
2
```

Problem 3. Fibonacci number

Problem Statement

Write a program that gets an integer n then returns the $n-$th Fibonacci number $F(n)$.
0th and 1st fibonacci numbers are $F(0) = 0, F(1) = 1$ respectively.

Input Statement

First line gives $t(\leq 1000)$ which is the number of test cases. Each 2nd ... $t+1$-th line includes a non-negative integer $n \leq 2^{31} - 1$.

Output Statement

For each test case, prints out $F(n)$ mod 20170317
Each test case should be separated by a line.

Input Example

3
1
2
3

Output Example

1
1
2