**Problem G**

**SHA-4**

Input File: pg.in

Time Limit: 3 seconds

This is a new hash algorithm invented for NCPC 2012, called SHA4 (simple hash algorithm version 4). Given a message string M, the SHA4 will hash the string into a 160 bits value, called the message digest d.

d = SHA4(M), where SHA4 is the hash function

We will give you the SHA4 algorithm and several hashed results of message digests: *d*0, *d*1, *d*2, …. You are to write a program to find the original message *M*0, *M*1, *M*2, …. Such that

*d*0 = *SHA*4(*M*0), *d*1 = *SHA*4(*M*1), *d*2 = *SHA*4(*M*2), …

Pseudocode for the SHA4 algorithm is listed in the following:

Note:  
1. All variables are unsigned 32 bists integers.

2. The input string **M** is with length 5.

3. Each input character **M[i]** is converted into an integer value, by subtracting ASCII value of space from the ASCII value of **M[i]**, that is, the difference between space and the character ASCII value. For example, the space value is 0.

4. After the value conversion, the five bytes values are stored in word[*i*], 0 <= *i* < 5.

**Technical Specification**

* The number of test cases is less the 1024.
* All output strings are with length 5.

The first line of the input file contains an integer, denoting the number of test cases to follow. For each test case, the message hash digest is given with values of five 32 bits integer in hexadecimal format, each integer separated with at least one space. For example, a sample digest is 0b8414f6 027eeb13 0453edaf 00f93379 002f2d88. All digest values are valid SHA4 outputs.

For each test case, output the original message with one line of string, containing only five alphanumeric characters (A-Za-z0-9). For example, you are to output ‘5dogs’ if the input is SHA4(‘5dogs’) = ‘0b8414f6 027eeb13 0453edaf 00f93379 002f2d88’.