**Java SHA-256**

<https://www.hackerrank.com/challenges/sha-256/problem>

Cryptographic hash functions are mathematical operations run on digital data; by comparing the computed *hash* (i.e., the output produced by executing a hashing algorithm) to a known and expected hash value, a person can determine the data's integrity. For example, computing the hash of a downloaded file and comparing the result to a previously published hash result can show whether the download has been modified or tampered with. In addition, cryptographic hash functions are extremely collision-resistant; in other words, it should be extremely difficult to produce the same hash output from two different input values using a cryptographic hash function.

*Secure Hash Algorithm 2* (*SHA-2*) is a set of cryptographic hash functions designed by the National Security Agency (NSA). It consists of six identical hashing algorithms (i.e., *SHA-256*, *SHA-512*, *SHA-224*, *SHA-384*, *SHA-512/224*, *SHA-512/256*) with a variable digest size. *SHA-256* is a *256*-bit (*32* byte) hashing algorithm which can calculate a hash code for an input of up to *264-1* bits. It undergoes *64* rounds of hashing and calculates a hash code that is a *64*-digit hexadecimal number.

Given a string, *s*, print its *SHA-256* hash value.

**Input Format**

A single alphanumeric string denoting *s*.

**Constraints**

* *6 <= |s| <= 20*
* *String s consists of English alphabetic letters (i.e., [a-zA-Z] and/or decimal digits (i.e., 0 through 9) only.*

**Output Format**

Print the *SHA-256* encryption value of *s* on a new line.

**Sample Input 0**

HelloWorld

**Sample Output 0**

872e4e50ce9990d8b041330c47c9ddd11bec6b503ae9386a99da8584e9bb12c4

**Sample Input 1**

Javarmi123

**Sample Output 1**

f1d5f8d75bb55c777207c251d07d9091dc10fe7d6682db869106aacb4b7df678