# 题目

你在和朋友一起玩猜数字（Bulls and Cows）游戏，该游戏规则如下：

你写出一个秘密数字，并请朋友猜这个数字是多少。

朋友每猜测一次，你就会给他一个提示，告诉他的猜测数字中有多少位属于数字和确切位置都猜对了（称为“Bulls”, 公牛），有多少位属于数字猜对了但是位置不对（称为“Cows”, 奶牛）。

朋友根据提示继续猜，直到猜出秘密数字。

请写出一个根据秘密数字和朋友的猜测数返回提示的函数，返回字符串的格式为 xAyB ，x 和 y 都是数字，A 表示公牛，用 B 表示奶牛。

xA 表示有 x 位数字出现在秘密数字中，且位置都与秘密数字一致。

yB 表示有 y 位数字出现在秘密数字中，但位置与秘密数字不一致。

请注意秘密数字和朋友的猜测数都可能含有重复数字，每位数字只能统计一次。

示例 1:

输入: secret = "1807", guess = "7810"

输出: "1A3B"

解释: 1 公牛和 3 奶牛。公牛是 8，奶牛是 0, 1 和 7。

示例 2:

输入: secret = "1123", guess = "0111"

输出: "1A1B"

解释: 朋友猜测数中的第一个 1 是公牛，第二个或第三个 1 可被视为奶牛。

说明: 你可以假设秘密数字和朋友的猜测数都只包含数字，并且它们的长度永远相等。

# 分析

**思路：**

1、首先想到的是，如果不考虑位置，所有猜对的数字，用个unordered\_map<char,int> 就可以求出

1）统计 secret 中每个字母出现的个数

2）在 guess 中检查每个字母，对 secret 中出现的字母计数

2、再想到，如果位置和数字都猜对，使用 for 循环便可求出，这就是 A

3、因为第一步求出的个数还包含第二步的情况，需要把它们排除，就得到 B

代码：

string getHint(string secret, string guess) {

int acnt = 0;

int bcnt = 0;

unordered\_map<char, int> um;

for (int i = 0; i < secret.size(); i++) {

um[secret[i]]++;

acnt += (secret[i] == guess[i]);

}

for (auto& c : guess) {

bcnt += (um[c]-- > 0);

}

bcnt -= acnt;

return to\_string(acnt) + "A" + to\_string(bcnt) + "B";

}