

题目描述：

给定用户密码输入流 *input*，输入流中字符‘<’表示退格，可以清除前一个输入的字符，请你编写程序，输出最终得到的密码字符，并判断密码是否满足如下的密码安全要求。

密码安全要求如下：

1.密码长度 ≥ 8 ;

2.密码至少需要包含 1 个大写字母;

3.密码至少需要包含 1 个小写字母;

4.密码至少需要包含 1 个数字;

5.密码至少需要包含 1 个字母和数字以外的非空白特殊字符

注意空串退格后仍然为空串，且用户输入的字符串不包含‘<’字符和空白字符。

输入描述：

用一行字符串表示输入的用户数据，输入的字符串中‘<’字符标识退格，用户输入的字符串不包含空白字符，例如：

ABc<c89%000<

输出描述：

输出经过程序处理后,输出的实际密码字符串,并输出改密码字符串是否满足密码安全要求。

两者间由‘,’分隔， 例如：

ABc89%00,true

补充说明:

示例 1

输入:

ABC

输出:

ABc89%00,true

说明:

解释: 多余的 *C* 和 *O* 由于退格被去除,最终用户输入的密码为 *ABc89%00*, 且满足密码安全要求, 输出 *true*

示例 2

输入:

输出:

ABC,false

说明:

不满足密码安全要求

示例 3

输入:

AB<

输出:

, false

说明:

输出字符串为空, 不满足密码安全要求

```
#include <iostream>
#include <string>
using namespace std;
```

```

bool hasBig(string &str) {
    for (auto c : str) {
        if (c >= 'A' && c <= 'Z')return true;
    }
    return false;
}

bool hasSmall(string &str) {
    for (auto c : str) {
        if (c >= 'a' && c <= 'z')return true;
    }
    return false;
}

bool hasNum(string &str) {
    for (auto c : str) {
        if (c >= '0' && c <= '9')return true;
    }
    return false;
}

bool hasSpec(string &str) {
    for (auto c : str) {
        if (c >= 'A' && c <= 'Z' || c >= 'a' && c <= 'z'
            || c >= '0' && c <= '9' || c == ' ') {
            continue;
        }
        else {
            return true;
        }
    }
    return false;
}

int main()
{
    string str;
    getline(cin, str);
    string ans;
    for (auto c : str) {
        if (c == '<') {
            if (ans.size() > 0) ans.pop_back();
        }
        else {
            ans += c;
        }
    }
}

```

```
}  
bool flag = false;  
if (ans.size() >= 8 && hasBig(ans) && hasSmall(ans)  
    && hasNum(ans) && hasSpec(ans)) {  
    flag = true;  
}  
cout << ans << ", " << boolalpha << flag << endl;  
  
return 0;  
}
```