

# 矩阵连乘

---

实现矩阵连乘。

## 输入

---

第一行是矩阵的个数  $N$ ，保证  $N$  可用 `std::size_t` 容纳（可以通过引入头文件 `<cstdint>` 使用此类型），保证不小于 2。

从第二行开始是矩阵的内容。对于矩阵  $m_k$ ，首先一行是矩阵的行数  $R_k$  和列数  $C_k$ ，保证可用 `std::size_t` 容纳并且一定大于 0，接下来是矩阵的内容，每行  $C_k$  个值，一共  $R_k$  行，所有的值都可以用 `int` 类型容纳。

每行输入保证以 `\n` 结尾。

保证：

- 矩阵的形状是正确的
- 所有矩阵占用的内存数（以字节为单位）在 `std::size_t` 的范围内

## 输出

---

输出矩阵  $m_1$  到  $m_N$  连乘的结果。

## 示例

---

### 示例 1

---

输入

```
2
2 3
1 2 3
4 5 6
3 2
6 5
4 3
2 1
```

输出

```
20 14
56 41
```

## 提示

- 可用如下方式读取一个矩阵

```
#include <algorithm>
#include <cstdint>
#include <iostream>
#include <iterator>

int main() {
    int mat[10][10];
    std::size_t n_rows, n_cols;
    std::cin >> n_rows >> n_cols;
    for (int i = 0; i < n_rows; i++) {
        std::copy_n(std::istream_iterator<int>(std::cin), n_cols,
mat[i]);
    }
}
```

