

Lab 1



工具

1. 请在 Linux 虚拟机里面完成本次实验，ICS 课提供的虚拟机就可以。
2. 请在命令行中依次输入下面的命令安装 minisat。这是一种 SAT solver。注意区分数字 1 和字母 l。

```
sudo apt-get install build-essential
```

```
sudo apt-get install minisat
```

```
sudo apt-get install zlib1g
```

```
sudo apt-get install zlib1g-dev
```

3. 如果你不想用 vim 写代码，建议在虚拟机里面安装 vscode 或者其它编辑器写代码，但这不是必要的。
4. 建议在 windows 里面安装 winscp 用于在 windows 和虚拟机之间传递文件，但这不是必要的。
5. 请解压压缩包所有文件到同一个文件夹下。

问题

河中有 n 个石头，Tom 和 Jerry 想要踩着石头过河，这些石头有两种状态，要么浮在水面上，要么沉在水底。河边有 m 个开关，控制石头的沉浮。每个开关控制 1 或 2 个石头，

每块石头被 1 或 2 个开关控制。每次打开或关闭一个开关，这个开关控制的石头都会改变状态，从水下浮到水上或者从水上沉到水下。

一开始所有开关都是关闭的，石头有的在水上有的在水下。请利用 minisat 来判断如何控制这些开关使得所有石头同时浮在水上。

输入

输入数据在 test.txt 里面。

第一行的整数是测试用例的数量，后面跟着所有测试用例，不同测试用例之间隔一行。

每个测试用例第一行是两个数字 m 和 n。m 是开关的数量，n 是石头的数量。下一行是 n 个数字，对应 n 个石头的初始状态，0 表示这个石头在水下，1 表示石头在水上。接下来 m 行是 m 个开关的信息。每一行的第一个数字表示这个开关控制了几个石头，后面跟着石头的编号，石头的编号从 1 开始。

输出

框架代码会把结果输出到 answer.txt 里面，每个测试用例的结果占一行。每一行包含几个 0 或 1，0 表示对应的开关应该关闭，1 表示对应的开关应该打开，这样每块石头都会浮在水上。如果没办法过河，那么这一行会只有一个 UNSAT。

实例

test.txt 的内容如下。

```
1
2 2
1 0
1 1
1 2
```

第一行表示有 1 个测试用例。第二行表示有 2 个开关和 2 个石头。第 3 行表示一开始，1 号石头在水上，2 号石头在水下。第 4 行表示第一个开关控制 1 块石头，是 1 号石头，第二个开关控制 1 块石头，是 2 号石头。

正确答案输出到 answer.txt 里，内容如下。

```
0 1
```

这表示应该关闭第一个开关并打开第二个开关。

代码框架

代码主要在 `main.cpp` 和 `lab1.cpp` 中，你所有的代码都要写在 `lab1.cpp` 中，不能修改任何其它文件，也不能在 `lab1.cpp` 里面调用任何文件操作或 `printf` 等输入输出操作。

`main.cpp` 已经完成了文件读取和答案输出功能，你需要完成 `lab1.cpp` 中的函数 `lab1`。在编写代码前请自行删掉 `lab1` 函数中的 `assert(0)`。

函数 `lab1` 包括 5 个参数。

- `n` 是石头的数量。
- `m` 是开关的数量。
- 数组 `states` 是 `n` 个石头的初始状态。数组长度可能大于 `n`，可以忽略多出来的部分。
- `Button` 是一个二维数组，记录开关的控制信息，其中每个元素都是石头的编号。例如，`button[3][0]` 和 `button[3][1]` 表示被第 4 个开关控制的石头编号。石头的编号从 1 开始。如果 `button[3][0] = 1` 并且 `button[3][1] = 0`，那么这个开关只控制 1 号石头。
- 数组 `answer` 里面是问题的答案。例如，`answer[2] = true` 表示第 3 个开关应该打开。
- `lab1` 函数的返回值类型是 `bool`，如果返回 `true`，表示有办法让所有石头同时浮在水上，如果返回 `false`，表示 Tom 和 Jerry 无法过河了。

假设这些参数的值都是合法的。

`lab1` 必须使用 `minisat` 完成，`minisat.cpp` 中有一个示例代码，是求解 $(\neg A \vee \neg B \vee C) \wedge (\neg A \vee \neg B \vee \neg C) \wedge (A \vee \neg B \vee C)$ 的。打开命令行，`cd` 到 `minisat.cpp` 所在文件夹，输入 `make example` 可以编译运行 `minisat.cpp`。

完成 `lab1` 后请打开命令行，`cd` 到代码所在文件夹，输入 `make run` 就可以编译运行代码。