

QM算法接口说明

0x00 问题

请根据QM算法，对十变量以内的逻辑表达式进行化简。输入的逻辑表达式以最小项和无关项来表示，期望的输出是字母组成的最简与或表达式。例如，对函数

$$f(A, B, C, D) = \sum m(4, 8, 10, 11, 12, 15) + \sum d(9, 14)$$

期望输出为

$$AC + BC'D' + AB'$$

或

$$AC + BC'D' + AD'$$

0x01 化简要求

变量个数及字母表示

首先应确定变量个数。实际输入时只输入最小项和无关项的编号，并不显式地指定变量字母，所以需要先对变量个数进行判断。

- **变量个数**：最小项、无关项编号中最大的编号的比特数。例如上述例子中，最大编号为15，有4个比特，故变量数为4。如果最大编号为0或没有编号，则变量数为0。
- **变量字母**：最高位为A，次高位为B，以此类推。反变量在变量字母后加撇号（单引号'）。
- **常量**：如出现常量，以0、1表示。

化简规则

1. **正确性**：化简后表达式应与化简前等价，即对于任意非无关项的输入，化简后表达式输出应与化简前相同。
2. **本质本原蕴含项**：本质本原蕴含项必须完全正确。
3. **项数最少**：满足如上条件的多个表达式，取项数最少的。
4. **答案不唯一**：满足上述三项规则的答案可能有多个，均算正确，但还需满足下面的排序规则和输入输出要求。

排序规则

最简表达式的项应按如下规则排序（优先级从高到低）：

1. 本质本原蕴含项在前，非本质本原蕴含项在后。
2. 同为本质本原蕴含项或同为非本质本原蕴含项，各项按字母顺序确定优先级，各级 $A > A' > (A \text{不出现})$ ，即：
 - a. 首先根据 A ，将有 A 的排最前，有 A' 的排中间，无 A 排最后；
 - b. 然后对于每一部分，根据 B ，重复上述过程。例如： $AB > AC > A'BC > A'B' > BC$ 。
提示：排序规则描述看似复杂，但程序不一定按照上述流程实现。实际上如果程序设计合理，只需要简单几步操作即可完成满足上述要求的排序。允许并鼓励大家使用STL中的 `std::sort()` 函数。
3. 每一项中的各个字母，按照字母顺序排序， A 在最前。

边界规则

常量：当变量个数为 0 时，此时最小项和无关项的数目均为 0 ，该逻辑表达式为常量，输出不应包含任何字母。如果卡诺图的值仅包括 1 和 x ，应输出 1 ；如果卡诺图的值仅包括 0 和 x （包括全为 x ），应输出 0 。（注意表达式 1 有一项，这一项变量数为零；而表达式 0 有零项）。这里规定¹：

$$\sum m() + \sum d() = 0$$

$$\sum m() + \sum d(0) = 0$$

$$\sum m(0) + \sum d() = 1$$

提示：边界规则看似复杂，但如果程序设计合理，这是正常的输出结果。如果无论如何都难以用统一的规则实现上述结果，可以单独判断。

示例

对函数

$$\sum m(4, 8, 10, 11, 12, 15) + \sum d(9, 14)$$

确定其变量数为 4 ，从高位到低位分别是 $A \sim D$ 。

画出其卡诺图：

CD\AB	00	01	11	10
00	0	1	1	1
01	0	0	0	x

11	0	0	1	1
10	0	0	x	1

确定本原蕴含项：AC、AD'、AB'、BC'D'。

确定本质本原蕴含项：AC、BC'D'，它们放在输出的最前。按排序规则，AC 应在 BC'D' 前。

剩下的本原蕴含项取最小覆盖，可为 AD 或 AB'。

因此结果为 AC+BC'D'+AB' 或 AC+BC'D'+AD 均算正确。

0x02 接口要求

要求通过标准输入（键盘）输入数据，通过标准输出（命令行）输出结果。

输入格式

第一行输入一个整数 N_q ，表示接下来将会输入的逻辑表达式的总数目，即总的“查询”的数目。

从第二行开始，重复进行 N_q 次查询。每次查询占三行。

每次查询的第一行输入两个整数 N_m ， N_d ，分别表示最小项和无关项的数目。

每次查询的第二行输入 N_m 个整数，为最小项。

每次查询的第三行输入 N_d 个整数，为无关项。

例如：

```
2
4 3
2 3 5 6
0 4 7
6 2
4 8 10 11 12 15
9 14
```

第一行的 2 表示将进行两次查询。

对于第一条查询，有 4 个最小项和 3 个无关项，表达式为

$$f(A, B, C) = \sum m(2, 3, 5, 6) + \sum d(0, 4, 7)$$

对于第二条查询，有 6 个最小项和 2 个无关项，表达式

$$f(A, B, C, D) = \sum m(4, 8, 10, 11, 12, 15) + \sum d(9, 14)$$

注：行数仅供参考。如果最小项或无关项个数为0，可能不会输入该行。所以程序在处理输入时应根据 N_q ， N_m ， N_d 来判断，而非行数。

输出格式

为了便于测试，对于输出的格式要求如下：

第一行输出姓名和学号，中间用一个空格隔开。从第二行开始输出结果。对于每一条查询，单独输出一行。

例如，学号为 2014000000 的 张三 同学，对于上面的输入，应输出：

```
张三 2014000000
```

```
A+B
```

```
AC+BC'D'+AB'
```

注意：不要输出任何其他内容，包括提示语（“请输入最小项：”、“化简完成！”等，使用VS时自己未输出而系统自动输出“按任意键继续”则没有关系）、多余的换行、空格等。

0x03 常见错误

以下列出几种常见的答案错误，提醒大家注意避免。

1. 输出错误

- **非法字符**：表达式中存在除 A ~ J, 0, 1, +, ' 之外的字符，或本应是常量表达式却输出了字母。
- **变量过多**：表达式中存在超出变量数范围的字母，如只有四个变量的表达式却输出了 E。
- **格式错误**：表达式格式错误，如连续多个 + 或 '，结尾出现 + 等。
- **变量顺序错**：同一项中各字母不按顺序排列，或有重复字母。
- **项顺序错**：输出表达式各项未按上述规则排序。
- **输出太少**：输出表达式总个数与输入不符。
- **输出太多**：输出表达式总个数超过输入个数，或所有答案输出完成后还输出了其他内容。

2. 逻辑错误

- **本质本原蕴含项错误**：本质本原蕴含项与正确答案不符。
- **表达式不等价**：化简后表达式与化简前不等价。
- **未化到最简**：存在项数更少的表达式。

脚注：

1. 如果程序设计合理，这是正常的输出结果。原因是：首先这三个表达式均为 0 个变量，即为常量，故卡诺图只有一个格子， $\sum m() + \sum d()$ 在格子中填入 0， $\sum m(0) + \sum d() = 1$ 在格子中填入 1。故

$$\sum m() + \sum d() = 0$$

$$\sum m(0) + \sum d() = 1$$

$\sum m() + \sum d(0)$ 在格子中填入 \times , 注意到表达式 1 有一项, 这一项变量数为零; 而表达式 0 有零项。按照项数最少的规则

$$\sum m() + \sum d(0) = 0$$

↩