


数据库工程师-性能组面试题

要求

请编程实现以下各题目, 请尽量优化你的程序以**高效处理海量数据**, 请注重代码风格与程序结构, 请以注释对程序的行为进行解释. 我们推荐你采用 C / C++ 进行答题, 但是你也可以选择任意你擅长的语言. 你可以使用语言自带的基础库, 如 C 中可使用 libc 提供的函数, C++ 中可以使用 STL. 可以假设内存大到足以容纳所有数据.

 提交内容不要包含测试所需的输入文件等无关文件或信息, 单个文件或打包文件务必控制在 3MB 以内!

任务 1

基于以下 C 程序框架, 我们希望你实现 `task1()` 这个函数, 它把 `rows` 中所有满足 `b >= 10 && b < 500000` 并且 `a == 10 || a == 200 || a == 3000` 的行的内容都打印到终端.

```
1 typedef struct Row
2 {
3     int a;
4     int b;
5 } Row;
6
7 /*!
8  * Task 1.
9  *
10  * Find out all the rows that satisfy below conditions:
11  *
12  * ((b >= 10 && b < 500000) &&
13  * (a == 10 || a == 200 || a == 3000))
14  *
15  * Print them to the terminal, one row per line, for example:
16  *
17  *      10,31
18  *      10,23
19  *      200,16
20  *
21  * @param nrows    The total number of rows.
22  * @param rows      The rows, for example rows[0] is the first row.
23  */
```

```
24 void
25 task1(const Row *rows, int nrows)
26 {
27     /*
28      * TODO: implement this function.
29      */
30 }
```

任务 2

在任务 1 的基础上, 如果输入的参数 `rows` 已经按照 `(a, b)` 进行过排序, 请实现 `task2()`, 对 `task1()` 的执行性能进行优化. 示例输入 (再次提醒, 实际输入为海量数据):

```
1 Row sorted_rows[] =
2 {
3     { 10, 31 },
4     { 10, 720000000 },
5     { 200, 22 },
6     { 200, 33 },
7     { 1500, 12 },
8     { 1500, 34 },
9     { 3000, 5 },
10 };
```

示例输出:

```
1 10,31
2 200,22
3 200,33
```

任务 3

在任务 2 的基础上, 我们期望你打印出的匹配行是按照 `b` 列进行排序的, 请实现 `task3()`. 示例输出:

```
1 200,22
2 10,31
3 200,33
```

