

# Experiment 2

## 1 数据库查询 v2

### 1.1 Description

勤奋的小明为了预习下学期的数据库课程，决定亲自实现一个简单的数据库系统。该数据库系统需要处理用户的数据库插入和查询语句，并输出相应的输出。具体来说，用户的输入共包含若干条插入语句和查询语句。其中每条插入语句包含一个非负整数表示需要插入的数据。每条查询语句包含一个整数表示待查询的键值，若该键值存在则直接输出该键值，否则输出数据库中比该键值小的最大键值。

### 1.2 Input Description

首先是若干行插入语句，每行的格式为如下的一种：INSERT *key*, FIND *key*。最后单独的一行EXIT 表示输入结束。

数据规模：

插入语句和查询语句一共不超过 2,000,000 条。

$0 \leq key \leq 10^9$

### 1.3 Output Description

对每条查询语句输出一行，每行输出 1 个数字，表示查询的结果。该键值存在则直接输出该键值，否则输出数据库中比该键值小的最大键值。

### 1.4 Submission Link

OnlineJudge Problem E2-1 数据库查询 v2: <https://202.38.86.171/problem/E2-1>

## 2 军训排队

### 2.1 Description

现有  $n$  个学生排成一个固定队伍进行军训，教官小明有一份所有  $n$  个人的名单（不同的人可能重名）。小明想要在整个队伍中找到一个连续的子队伍，并且满足该子队的所有人恰好有  $k$  个不重复的名字。请帮小明计算一下一共有多少种可能的子队伍。

### 2.2 Input Description

一共有两行，第一行有两个数字  $n$  和  $k$ ，用空格分隔。第二行有  $n$  个单词  $name_i$ ，用空格分隔。输入保证  $len(name_i) \leq 5$ （即输入的名字最多只含有 5 个字符）

数据规模：

$$0 < k < n \leq 10,000,000$$

### 2.3 Output Description

输出一共一个数字，即可能的子队伍数量。

### 2.4 Submission Link

OnlineJudge Problem E2-2 军训排队: <https://202.38.86.171/problem/E2-2>

## 3 内存分配

### 3.1 Description

C 语言中需要申请一块连续的内存时需要使用 malloc 等函数。如果分配成功则返回指向被分配内存的指针 (此存储区中的初始值不确定), 否则返回空指针 NULL。

现在小明决定实现一个类似 malloc 的内存分配系统, 具体来说, 他需要连续处理若干申请内存的请求, 这个请求用一个闭区间  $[a_i..b_i]$  来表示。当这个区间和当前已被申请的内存产生重叠时, 则返回内存分配失败的信息。否则返回内存分配成功, 并将该区间标记为已被占用。

假设初始状态下内存均未被占用, 管理的内存地址范围为  $0 \sim 1\text{GB}$  ( $0 \sim 2^{30}$ )。

### 3.2 Input Description

输入数据共  $n+1$  行。第一行一个整数  $n$  表示共需要处理  $n$  次内存分配。然后是  $n$  行数据, 每行的格式为  $a_i\ b_i$ , 表示申请区间为  $[a_i, b_i]$ 。

数据规模:

$$\begin{aligned} n &\leq 1,000,000 \\ 0 &< a_i \leq b_i \leq 2^{30} \end{aligned}$$

### 3.3 Output Description

输出共  $n$  行。对于每行内存分配的申请, 若申请成功则输出一行 0, 若申请失败则输出一行 -1。

### 3.4 Submission Link

OnlineJudge Problem E2-3 内存分配: <https://202.38.86.171/problem/E2-3>

## 4 危险品放置 (选做)

### 4.1 Description

现有若干危险品需要放置在  $A, B$  两个仓库。当两种特定的危险品放置在相同地点时即可能产生危险。我们用危险系数  $\alpha_{i,j}$  表示危险品  $i, j$  放置在一起的危险程度。一些危险品即使放置在一起也不会产生任何危险, 此时  $\alpha_{i,j} = 0$ , 还有一些危险品即使单独放置也会产生危险, 此时  $\alpha_{i,i} > 0$ 。定义两个仓库整体的危险系数为

$$\max(\max_{i,j \in A} \alpha_{i,j}, \max_{i,j \in B} \alpha_{i,j}),$$

即放置在一起的所有危险品两两组合的危险系数的最大值。现在对于一组给定的危险系数, 需要设计方案使得整体危险系数最小。

### 4.2 Input Description

输入共  $m+1$  行。第一行两个整数  $n$  和  $m$  表示共有  $n$  种危险品, 危险品之间的危险组合 (危险系数非零的物品组合) 共  $m$  种。接下来的  $m$  行, 每行三个整数  $i, j, \alpha_{i,j}$ , 表示  $(i, j)$  为危险组合 ( $i, j$  可能相等), 其危险系数为  $\alpha_{i,j} > 0$ 。

数据规模:

$$0 < n \leq 100,000$$

$$0 < m \leq 1,000,000$$

### 4.3 Output Description

输出共一行, 包含一个整数, 表示整体危险系数的最小值。

### 4.4 Submission Link

OnlineJudge Problem E2-EX 危险品放置: <https://202.38.86.171/problem/E2-EX>