

2014 年安徽省青少年信息学奥林匹克竞赛

中学组试题

第一试

AOI 2014

(请选手仔细阅读本页内容)

比赛用时 (5 小时)

竞赛时间：2014 年 4 月 18 日 8:00–13:00

题目名称	拼图	宅男计划	骑士游戏
目录	puzzle	food	knight
可执行文件名	puzzle.exe	food.exe	knight.exe
输入文件名	puzzle.in	food.in	knight.in
输出文件名	puzzle.out	food.out	knight.out
每个测试点时限	1 秒	1 秒	3 秒
内存限制	256M	256M	256M
测试点数目	10	10	10
每个测试点分值	10	10	10
是否有部分分	否	否	否
题目类型	传统型	传统型	传统型

提交源程序须加后缀

对于 Pascal 语言	puzzle.pas	food.pas	knight.pas
对于 C 语言	puzzle.c	food.c	knight.c
对于 C++ 语言	puzzle.cpp	food.cpp	knight.cpp

注意：最终测试时，所有编译命令均不打开任何优化开关。

拼图

【故事背景】

JYY 最近迷上了拼图游戏。作为一个计算机科学家，JYY 有一套黑白色的拼图，他希望通过合理的拼接，使得拼出的最终图案中，能包含面积最大的全白色子矩形。

【问题描述】

JYY 一共有 S 块拼图，并且由 1 到 S 编号。编号为 i 的拼图是一个 N 行 W_i 列的方格矩形，每个方格都为黑色或者白色。

一开始 JYY 将他的这 S 块拼图按照编号顺序左右相连依次放在桌上拼成了一个 N 行 M 列（这里 $M = \sum_{i=1}^S W_i$ ）的大矩形。

之后 JYY 发现，可以通过改变这 S 块拼图的连接次序，使得拼成的 N 行 M 列的大矩形中，最大全白子矩形面积变大。

现在 JYY 想知道，怎么拼才能得到最大的全白子矩形呢？请你帮助他计算出最佳的拼接方案。

【输入格式】

从文件 **puzzle.in** 中读入数据。

每个输入文件中包含多组测试数据。输入文件第一行包含一个整数 T ，代表测试数据的组数，接下来按顺序描述了每组测试数据。

每组测试数据的第一行包含两个整数 S 和 N 。

接下来 S 组输入，第 i 组对应编号为 i 的拼图。

在第 i 组输入中，第一行包含一个整数 W_i ；

接下来 N 行描述一个 N 行 W_i 列的 0/1 矩形；

其中第 x 行 y 列为 0 则表示该拼图对应位置的颜色是白色，反之则为黑色。

【输出格式】

输出到文件 **puzzle.out** 中。

对于每组数据输出一行包含一个整数 **ans**，表示最大可能的全白色子矩形的面积。

【样例输入】

```
1
3 4
4
1001
0000
```

0010
1001
3
000
010
000
011
2
00
10
01
00

【样例输出】

6

【样例说明】

最佳方案所最终拼成的大矩形为

000100100
01**000**0010
00**000**1001
011100100

粗体部分是面积为 6 的白色子矩形。

【数据规模与约定】

对于 10%的数据满足 $S = 1, N \leq 30, W_i \leq 30$;

对于 30%的数据满足 $S \leq 3$;

对于 50%的数据满足 $S \leq 1000$;

对于 100%的数据满足 $1 \leq S, N, W_i \leq 10^5, N \times \sum_{i=1}^S W_i \leq 10^5, T \leq 3$ 。

宅男计划

【故事背景】

自从迷上了拼图, JYY 就变成了个彻底的宅男。为了解决温饱问题, JYY 不得不依靠叫外卖来维持生计。

【问题描述】

外卖店一共有 N 种食物, 分别有 1 到 N 编号。第 i 种食物有固定的价钱 P_i 和保质期 S_i 。第 i 种食物会在 S_i 天后过期。JYY 是不会吃过期食物的。

比如 JYY 如果今天点了一份保质期为 1 天的食物, 那么 JYY 必须在今天或者明天把这个食物吃掉, 否则这个食物就再也不能吃了。保质期可以为 0 天, 这样这份食物就必须在购买当天吃掉。

JYY 现在有 M 块钱, 每一次叫外卖需要额外付给送外卖小哥外送费 F 元。送外卖的小哥身强力壮, 可以瞬间给 JYY 带来任意多份食物。JYY 想知道, 在满足每天都能吃到至少一顿没过期的外卖的情况下, 他可以最多宅多少天呢?

【输入格式】

从文件 **food.in** 中读入数据。

第一行包含三个整数 M , F 和 N 。

接下来 N 行, 第 i 行包含两个整数 P_i 和 S_i 。

【输出格式】

输出到文件 **food.out** 中。

输出仅包含一行一个整数表示 JYY 可以宅的最多的天数。

【样例输入】

```
32 5 2
5 0
10 2
```

【样例输出】

```
3
```

【样例说明】

JYY 的最佳策略是:

第一天买一份食物 1 和一份食物 2 并且吃一份食物 1;

第二天吃一份食物 2;

第三天买一份食物 1 并且吃掉。

【数据规模与约定】

对于 10%的数据满足 $N = 1$;

对于 30%的数据满足 $M, S_i \leq 300$;

对于 60%的数据满足 $M, S_i \leq 300000$;

对于 100%的数据满足 $0 \leq S_i \leq 10^{18}, 1 \leq F, P_i, M \leq 10^{18}, 1 \leq N \leq 200$ 。

骑士游戏

【故事背景】

长期的宅男生活中，JYY 又挖掘出了一款 RPG 游戏。在这个游戏中 JYY 会扮演一个英勇的骑士，用他手中的长剑去杀死入侵村庄的怪兽。

【问题描述】

在这个游戏中，JYY 一共有两种攻击方式，一种是普通攻击，一种是法术攻击。两种攻击方式都会消耗 JYY 一些体力。

采用普通攻击进攻怪兽并不能把怪兽彻底杀死，怪兽的尸体可以变出其他一些新的怪兽，注意一个怪兽可能经过若干次普通攻击后变回一个或更多同样的怪兽；而采用法术攻击则可以彻底将一个怪兽杀死。当然了，一般来说，相比普通攻击，法术攻击会消耗更多的体力值（但由于游戏系统 bug，并不保证这一点）。

游戏世界中一共有 N 种不同的怪兽，分别由 1 到 N 编号，现在 1 号怪兽入侵村庄了，JYY 想知道，最少花费多少体力值才能将所有村庄中的怪兽全部杀死呢？

【输入格式】

从文件 **knight.in** 中读入数据。

第一行包含一个整数 N 。

接下来 N 行，每行描述一个怪兽的信息：

其中第 i 行包含若干个整数，前三个整数为 S_i ， K_i 和 R_i ，表示对于 i 号怪兽，普通攻击需要消耗 S_i 的体力，法术攻击需要消耗 K_i 的体力，同时 i 号怪兽死亡后会产生 R_i 个新的怪兽。~~接下来 R_i 个 1 到 N 之间并且不为 i 的整数~~，表示一个新出现的怪兽编号。同一编号的怪兽可以出现多个。

【输出格式】

输出到文件 **knight.out** 中。

输出一行一个整数，表示最少需要的体力值。

【输入样例】

```
4
4 27 3 2 3 2
3 5 1 2
1 13 2 4 2
5 6 1 2
```

去掉

【输出样例】

26

【样例说明】

首先用消耗 4 点体力用普通攻击，然后出现的怪兽编号是 2，2 和 3。花费 10 点体力用法术攻击杀死两个编号为 2 的怪兽。剩下 3 号怪兽花费 1 点体力进行普通攻击。此时村庄里的怪兽编号是 2 和 4。最后花费 11 点体力用法术攻击将这两只怪兽彻底杀死。

一共花费的体力是 $4+5+5+1+5+6=26$ 。

【数据规模】

对于 20%的数据满足 $N \leq 5$;

对于 50%的数据满足 $N \leq 300$;

对于 100%的数据满足 $2 \leq N \leq 2 \times 10^5, 1 \leq R_i, \sum_{i=1}^N R_i \leq 10^6, 1 \leq K_i, S_i \leq 5 \times 10^{14}$ 。

