**CCF全国信息学奥林匹克联赛模拟赛(NOIP2020)复赛**

普及组

(请选手务必仔细阅读本页内容)

**一、题目概况**

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名称 | 氪金池 | 直播计划 | 2077 |
| 英文名与子文件夹名 | genshin | live | cyberpunk |
| 可执行文件名 | genshin | live | cyberpunk |
| 输入文件名 | genshin.in | live.in | cyberpunk.in |
| 输出文件名 | genshin.out | live.out | cyberpunk.out |
| 每个测试点时限 | 1s | 1s | 1s |
| 测试点数目 | 10 | 10 | 10 |
| 每个测试点分值 | 10 | 10 | 10 |
| 附加样例文件 | 无 | 无 | 无 |
| 结果比较方式 | 全文比较（忽略行末空格与回车） | | |
| 题目类型 | 传统 | 传统 | 传统 |
| 运行内存上限 | 256M | 256M | 256M |

**二、提交源程序文件名**

|  |  |  |  |
| --- | --- | --- | --- |
| 对于C++语言 | genshin.cpp | live.cpp | cyberpunk.cpp |
| 对于C语言 | genshin.c | live.c | cyberpunk.c |
| 对于pascal语言 | genshin.pas | live.pas | cyberpunk.pas |

**三、编译命令（不包含任何优化开关）**

|  |  |  |  |
| --- | --- | --- | --- |
| 对于C++语言 | g++ -o genshin genshin.cpp -lm | g++ -o live live.cpp -lm | g++ -o cyberpunk cyberpunk.cpp -lm |
| 对于C语言 | gcc -o genshin genshin.c -lm | gcc -o live live.c -lm | gcc -o cyberpunk cyberpunk.c -lm |
| 对于pascal语言 | fpc genshin.pas | fpc live.pas | fpc cyberpunk.pas |

**注意事项：**

1、文件名（程序名和输入输出文件名）必须使用英文小写。

2、C/C++中函数main()的返回值类型必须是int，程序正常结束时的返回值必须是0。

3、全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz，内存32GB。上述时限以此配置为准。本次评测机为 i7-10875h,内存16GB。

4、只提供Linux格式附加样例文件。

5、特别提醒：评测在当前最新公布的NOI Linux下进行，各语言的编译器版本以其为准。

**1.氪金池**

**(genshin.cpp/c/pas)**

【问题描述】

胡桃最近迷上了一款游戏，名为《Breath of the waifu》.这是一款类似塞尔达的开放世界游戏，一经上线便火爆全球。

游戏里有许多人物，每个人物都有不同的外貌技能等等。有一些人物是免费的，而有一些只能通过抽奖获得。最近才推出了许多花钱抽奖才能获得的角色，其中一个名为Klee的可爱的角色得到了胡桃的青睐。她发誓不管花多少钱也要得到Klee。

这款游戏的角色获取机制是这样的：

玩家需要通过人民币购买游戏内的货币，再通过游戏内的货币购买若干次抽奖次数。单次抽中想要的角色概率是很低的，所以想要抽出某个角色，必须要先准备大量的人民币。这一行为也被称之为“氪金”。为了避免出现“**连续几百次都抽不出想要的角色**”这样的运气爆棚情况，游戏引入了保底机制，即保证最多在n次连续抽取下，必定至少有一次出现想要的角色。

游戏商城提供三种优惠礼包。第i个礼包的价格是yi元人民币,礼包内容是xi次抽奖机会。玩家只可以选择三种中的**一种礼包**进行购买，不过可以购买无限次。

胡桃想问请你计算一下，她最少花多少钱才能保证获得Klee?

【输入输出】

输入文件为genshin.in,输出文件为genshin.out

输入第一行，一个整数n，表示保底次数。

接下来的三行，每行两个整数x,y，分别表示这个礼包包含xi次抽奖机会，价格是yi元。

输出一行，一个整数，表示保底获得Klee时花费的最少钱数。

【样例文件】

|  |  |
| --- | --- |
| genshin.in | genshin.out |
| 57  2 2  50 30  30 27 | 54 |

【样例说明】

保底57抽就能获得Klee，三种礼包分别是2抽/2元,50抽/30元,30抽/27元。

选择第一个礼包，要买29次，获得58抽，花费58元。

选择第二个礼包，要买2次，获得100抽，花费60元。

选择第三个礼包，要买2次，获得60抽，花费54元。

综合来看，54元最少。输出54.

【数据规模】

保证输入的所有数字不超过10000.

**2.直播计划**

**(live.cpp/c/pas)**

【问题描述】

啤梨啤梨是一家新起的视频网站，最近的直播行业也是发展的如火如荼。为了吸引新观众，啤梨啤梨花8亿元买下了联合传说电竞赛事的直播权。由于这项赛事的观众非常多，瞬间涌入的流量直接冲垮了破站的服务器。为了避免这种事情再次发生，董事长决定开启如下的直播计划。

根据流量统计发现，在不同的时间段观众人数是有变化的。观众人数多时，可以多开几台服务器来分散观众流量，观众人数少时可以关闭几台服务器来减少花费。这样一来既能保证观众的流畅直播体验，也能最大程度节省开支。

但是，破站的服务器很破，**一旦关闭，就无法再次打开了**，所以多准备些服务器才是良策。根据流量报告显示，一天被分为n个时间段，第i个时间段需要至少ai台服务器开启才能保证不被冲垮。每台服务器每个时间段的花费是固定的，为1，求**在花费最少的情况下**最少需要准备几台服务器？

【输入输出】

输入文件为live.in,输出文件为live.out

输入第一行，一个整数n表示有n个时间段。

输入第二行，n个整数，第i个整数ai表示这个时间段需要至少ai台服务器。

输出仅1行，一个整数，表示最少花销的前提下最少的服务器数。

【样例文件】

|  |  |
| --- | --- |
| live.in | live.out |
| 5  2 3 4 1 2 | 5 |

【样例说明】

这一天被分为5个时间段，每个时间段分别要2、3、4、1、2台服务器。

最终答案为5台服务器，第一台开启时间[1,5],第二台开启时间[1,3],第三台开启时间[2,3],第四台开启时间[3,3],第五台开启时间[5,5]，这种方案下总的花销为12，为最少的花销方案，在这个前提下服务器数量至少为5台。

【数据范围】

30%的数据 每天最多10个时间段。

70%的数据 每天最多1000个时间段。

100%的数据 每天最多100000个时间段，每个时间段的服务器数不超过10000台。

**3.2077**

**(cyberpunk.cpp/c/pas)**

【问题描述】

在2077年，夜之城被评为全球最差城市。离谱的犯罪率，高科技与低生活。无法否认，事实如此，但即使这样，来此居住的人还是络绎不绝——毕竟这个城市总会给人一些希望。谎言也好，幻觉也罢，皆近在咫尺，仿佛触手可及。这是座迷幻的城市，住满着逐梦之人。

在这里，义肢改造已经普及。每个人都可以随意更换身体的零部件。在这个弱肉强食的世界里，不会武装自己夜之城公民活不过三天。当然，改造加成也是有限制的。目前的技术下，人体一共有n个不同部位可供改造，第i个部位的可改造程度上限为a\_i。若某个部位不改造则改造程度为0。

目前人类的改造承受极限为m，也即所有部位的改造程度之和应该不大于m。改造策略很多，你可以选择每个身体部位都改造一些，也可以选择将某几个部位改造到上限，而放弃另一些部位的改造机会。当然，这全看您自己——优胜劣汰，适者生存。

问题来了：在极限情况下，你有多少种不同的改造策略呢？

【输入输出】

输入文件为cyberpunk.in,输出文件为cyberpunk.out

输入第一行，两个整数，n,m表示n个部位可供改造，m个改造点。

第二行有n个整数，每两个整数之间用一个空格隔开，依次表示a1,a2,…,an ​

【样例文件】

|  |  |
| --- | --- |
| cyberpunk.in | cyberpunk.out |
| 2 4  3 4 | 4 |

【样例说明】

2个部位，不妨假设为手臂和眼睛，改造上限分别为3与4。极限承受为4。

共4种方案，分别对应两个部位

（手臂升级0次+眼睛升级4）

（手臂升级1次+眼睛升级3）

（手臂升级2次+眼睛升级2）

（手臂升级3次+眼睛升级1）

【数据范围】

对于20%数据，有0<n≤8,0<m≤8,0≤ai≤8

对于50%数据，有0<n≤20,0<m≤20,0≤ai≤20

对于100%数据，有0<n≤100,0<m≤100,0≤ai≤100