

快快编程模拟赛

提高+/省选

(请选手务必仔细阅读本页内容)

一、题目概况

中文题目名称	树	质	串
英文题目与子目录名	tree	prime	string
可执行文件名	tree	prime	string
输入文件名	tree.in	prime.in	string.in
输出文件名	tree.out	prime.out	string.out
每个测试点时限	1秒	1秒	1秒
测试点数目	10	10	10
每个测试点分值	10	10	10
附加样例文件	有	有	有
结果比较方式	全文比较，过滤未行后空行，不过滤中间行行末空格		
题目类型	传统	传统	传统
运行内存上限	512M	512M	512M

二、提交源程序文件名

对于C++语言	tree.cpp	prime.cpp	string.cpp
对于C语言	tree.c	prime.c	string.c
对于Pascal语言	tree.pas	prime.pas	string.pas

三、编译命令（不包含任何优化开关）

对于C++语言	g++ -o tree tree.cpp -1m	g++ -o prime prime -1m	g++ -o string string.cpp -1m
对于C语言	gcc -o tree tree.c -1m	gcc -o prime prime.c -1m	gcc -o string string.c -1m
对于Pascal语言	fpc tree.pas	fpc prime.pas	fpc string.pas

1.树

(tree.cpp/c/pas)

【问题描述】

n 个点组成一个森林，点 i 的父节点是 R_i ($i=1..n$)，如 $R_i=0$ 表示 i 没有父节点。每个点有2个点权： P_i 表示收益， S_i 表示成本。现在要选其中 k 个点，使性价比（总收益/总成本）最大。如选了 i ，且 i 有父节点，则 R_i 也必须要选

【输入格式】

输入文件为tree.in
第1行2个正整数 k,n
后面 n 行每行3个整数 S_i,P_i,R_i

【输出格式】

输入文件为tree.out
一个实数表示最大的性价比，保留3位小数

【输入输出样例1】

tree.in	tree.out
1 2 1000 1 0 1 1000 1	0.0001

【数据规模与约定】

20%的数据， $n \leq 20$
50%的数据， $n \leq 100$
100%的数据， $1 \leq k \leq n \leq 2500, 0 < S_i, P_i \leq 10000, 0 \leq R_i < i$

2.质

(prime.cpp/c/pas)

【问题描述】

有 n 个数 $a_1..a_n$ ，选出一个最大的子集，使其中两两元素之和都不是质数

【输入格式】

输入文件为prime.in

第1行一个正整数 n

第2行 n 个正整数 $a_1..a_n$

【输出格式】

输出文件为prime.out

1个整数表示满足条件的最大子集的大小

【输入输出样例1】

prime.in	prime.out
6 1 2 2 3 4 10	4

【数据规模与约定】

10%的数据， $n \leq 10$

10%的数据， $n \leq 150$

10%的数据， $n \leq 1000$

10%的数据， $2 \leq n \leq 3000$ ， $1 \leq a_i \leq 100000$

3.串

(string.cpp/c/pas)

【问题描述】

两个长度为 n 的字符串 A, B 。求如下任一种字符串中最长的回文串长度

1. $A.substr(i, j)$, $1 \leq i \leq j \leq n$

2. $B.substr(i, j)$, $1 \leq i \leq j \leq n$

3. $A.substr(i, j) + B.substr(j, k)$, $1 \leq i \leq j \leq k \leq n$

$S.substr(i, j)$ 表示字符串 S 从 i 位到 j 位（包含 i, j ）的子串， $+$ 表示字符串连接

【输入格式】

输入文件为string.in

第1行1个正整数 n

第2行1个字符串 A

第3行1个字符串 B

【输出格式】

输出文件为string.out

一个整数表示答案

【输入输出样例1】

string.in	string.out
5 ABCDE BAECB	5

【输入输出样例1说明】

$A.substr(2, 3) + B.substr(3, 5) = "BC" + "ECB" = "BCECB"$ ，最优解不一定唯一

【数据规模与约定】

10%的数据， $n \leq 30$

30%的数据， $n \leq 200$

50%的数据， $n \leq 2000$

100%的数据， $1 \leq n \leq 100000$