GOLD PROBLEMS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Four problems numbered 1 through 4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Problem 1: Ski Lessons [Brian Jacokes, 2002]

Farmer John wants to take Bessie skiing in Colorado. Sadly, Bessie

is not really a very good skier.

Bessie has learned that the ski resort is offering S (0 <= S <=

100) ski classes throughout the day. Lesson i starts at time M\_i

(1 <= M\_i <= 10,000) and lasts for time L\_i (1 <= L\_i <= 10,000).

After lesson i, Bessie's ski ability becomes A\_i (1 <= A\_i <= 100).

Note: this ability is an absolute, not an incremental change.

Bessie has purchased a map which shows all N (1 <= N <= 10,000) ski

slopes along with the time D\_i (1 <= D\_i <= 10,000) required to ski

down slope i and the skill level C\_i (1 <= C\_i <= 100) required to

get down the slope safely. Bessie's skill level must be greater

than or equal to the skill level of the slope in order for her to

ski down it.

Bessie can devote her time to skiing, taking lessons, or sipping

hot cocoa but must leave the ski resort by time T (1 <= T <= 10,000),

and that means she must complete the descent of her last slope

without exceeding that time limit.

Find the maximum number of runs Bessie can complete within the time

limit. She starts the day at skill level 1.

Extra feedback will be provided on the first 50 submissions.

PROBLEM NAME: ski

INPUT FORMAT:

\* Line 1: Three space-separated integers: T, S, and N

\* Lines 2..S+1: Line i+1 describes ski lesson i with three

space-separated integers: M\_i, L\_i, and A\_i

\* Lines S+2..S+N+1: Line S+i+1 describes ski slope i with two

space-separated integers: C\_i and D\_i.

SAMPLE INPUT (file ski.in):

10 1 2

3 2 5

4 1

1 3

OUTPUT FORMAT:

A single integer on a line by itself, the maximum number of runs that

Bessie may ski within the time limit.

SAMPLE OUTPUT (file ski.out):

6

OUTPUT DETAILS:

Ski the second slope once, take the lesson, and ski the first slope 5 times

before time is up: a total of 6 slopes.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Problem 2: Work Scheduling [Richard Peng, 2008]

Farmer John has so very many jobs to do! In order to run the farm

efficiently, he must make money on the jobs he does, each one of

which takes just one time unit.

His work day starts at time 0 and has 1,000,000,000 time units (!). He

currently can choose from any of N (1 <= N <= 100,000) jobs

conveniently numbered 1..N for work to do. It is possible but

extremely unlikely that he has time for all N jobs since he can

only work on one job during any time unit and the deadlines tend

to fall so that he can not perform all the tasks.

Job i has deadline D\_i (1 <= D\_i <= 1,000,000,000). If he finishes

job i by then, he makes a profit of P\_i (1 <= P\_i <= 1,000,000,000).

What is the maximum total profit that FJ can earn from a given list

of jobs and deadlines? The answer might not fit into a 32-bit

integer.

PROBLEM NAME: job

INPUT FORMAT:

\* Line 1: A single integer: N

\* Lines 2..N+1: Line i+1 contains two space-separated integers: D\_i

and P\_i

SAMPLE INPUT (file job.in):

3

2 10

1 5

1 7

OUTPUT FORMAT:

\* Line 1: A single number on a line by itself that is the maximum

possible profit FJ can earn.

SAMPLE OUTPUT (file job.out):

17

OUTPUT DETAILS:

Complete job 3 (1,7) at time 1 and complete job 1 (2,10) at time 2

to maximize the earnings (7 + 10 -> 17).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Problem 3: Tower of Hay [Brian Dean, 2009]

Cows so hate the dark. In order to change a lightbulb at the top

of the barn, Bessie must build a tower out of bales of hay to climb

so that she can reach it. The N (1 <= N <= 100,000) bales of hay

(conveniently numbered 1..N) enter the barn sequentially on a

conveyor belt. Bale i has integer width w\_i (1 <= w\_i <= 10,000);

all bales have depth and height of one unit.

Bessie must use all N hay bales to build the tower and must lay

them down in the order they arrive. She can lay as many bales as

she wishes (tightly packed in a line, of course) to create the

foundation of the tower. She can then lay some of the next bales

on top of the previous level to create the next level (no level can

be wider than the level beneath it). She continues this process

until all the bales are used. She must stack the bales in the order

they arrive, from the bottom to the top of the tower. To be clear:

she must not try to sneak a bale back on the foundation after a

bale is placed on the second layer.

Bessie's goal is to create the tallest tower possible -- and she

already knows how wide each every bale of hay will be as it comes

into the barn on the belt. How tall a tower can she construct?

PROBLEM NAME: tower

INPUT FORMAT:

\* Line 1: A single integer: N

\* Lines 2..N+1: Line i+1 contains a single integer: w\_i

SAMPLE INPUT (file tower.in):

3

1

2

3

OUTPUT FORMAT:

\* Line 1: A single integer, the height of the tallest possible tower

that can be built

SAMPLE OUTPUT (file tower.out):

2

OUTPUT DETAILS:

Use the first bales with width 1 and 2 for the bottom row (total

width: 3), then the next bale (width 3) for the top row:

+----------+

| 3 |

+---+------+

| 1 | 2 |

+---+------+

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Problem 4: Bovine Embroidery [Brian Dean, 2009]

Bessie has taken up the detailed art of bovine embroidery. Cows

embroider a cloth mounted in a circular hoop of integer radius d

(1 <= d <= 50,000). They sew N (2 <= N <= 50,000) threads, each in

a straight line from one point on the edge of the hoop to another

point on the edge of the hoop (no two embroidered points share a

location on the hoop's edge).

Being mathematically inclined, Bessie knows a formula of the form

ax + by + c = 0 for each straight line piece of thread. Conveniently,

a, b, and c are integers (-1,000,000 <= a <= 1,000,000; -1,000,000

<= b <= 1,000,000; -1,000,000 <= c <= 1,000,000). Even more

conveniently, no two threads coincide exactly.

Perhaps less conveniently, Bessie knows that her set of formula

coefficients also includes a number of formulae for threads that

do not appear to pass inside the hoop's circle. She regrets this

greatly.

The origin (0,0) is in the precise middle of the hoop, so all points

on the hoop's edge are distance d from the origin. At least one of

the coefficients a and b is non-zero for each thread's formula.

Bovine embroidery is more highly regarded when the number of thread

intersections is maximized. Help Bessie: count the number of pairs

of threads that intersect on the cloth (i.e., within distance d of

the origin). Note that if three threads happen to coincide at the

same point, that would be three pairs of intersections. Four threads

at the same point -> six pairs of intersections, etc.

PROBLEM NAME: cowemb

INPUT FORMAT:

\* Line 1: Two space-separated integers: N and d

\* Lines 2..N+1: Line i+1 describes thread i with three integers: a, b,

and c

SAMPLE INPUT (file cowemb.in):

2 1

1 0 0

0 1 0

INPUT DETAILS:

The two lines are x=0 and y=0.

OUTPUT FORMAT:

\* Line 1: One integer, on a line by itself, that is the count of pairs

of threads that intersect.

SAMPLE OUTPUT (file cowemb.out):

1

OUTPUT DETAILS:

The two lines intersect at (0,0), which is clearly with 1 of the origin.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*