POUR1 - Pouring water

#gcd (/problems/tag/gcd) #recursion (/problems/tag/recursion)

Given two vessels, one of which can accommodate *a* litres of water and the other - *b* litres of water, determine the number of steps required to obtain exactly *c* litres of water in one of the vessels.

At the beginning both vessels are empty. The following operations are counted as 'steps':

- emptying a vessel,
- filling a vessel,
- pouring water from one vessel to the other, without spilling, until one of the vessels is either full or empty.

Input

An integer *t*, 1<=*t*<=100, denoting the number of testcases, followed by *t* sets of input data, each consisting of three positive integers *a, b, c*, not larger than 40000, given in separate lines.

Output

For each set of input data, output the minimum number of steps required to obtain c litres, or -1 if this is impossible.

Example

Sample input:

```
2
5
2
3
2
3
4
```

Sample output:

```
2
-1
```

Submit solution! (/submit/POUR1/)

hide comments

< Previous 1 2 (/problems/POUR1/cstart=10)
3 (/problems/POUR1/cstart=20) 4 (/problems/POUR1/cstart=30)
5 (/problems/POUR1/cstart=40) 6 (/problems/POUR1/cstart=50)
Next (/problems/POUR1/cstart=10) > (/problems/POUR1/cstart=50)

mahilewets (/users/mahilewets): 2017-08-29 20:18:26
I think BFS is bad here

Because 40000 limitation

If limitation is like 255

We can easily afford BFS even with three vessels

arthur1991 (/users/arthur1991): 2017-03-26 07:38:10
i love qianqian

osama2003 (/users/osama2003): 2017-01-25 00:31:58
-1 .. nguenthanhloc


muneebaadil (/users/muneebaadil): 2017-01-04 21:37:23
Doing through BFS+maps too. But TLE. Using the following algorithm.
Each node having pair (x, y) derives four more pairs (provided that this node isn't visited before)
(a, y), (x, b), (x-min(x, b-y), y+min(x, b-y)), (x+min(y, a-x), y-min(y, a-x)) and finally the node having pair(x, y) is marked visited.
What am I doing wrong?

flyingduchman_ (/users/flyingduchman_): 2016-12-27 14:37:10
Actual problem: You are at the side of a river. You have a "a" liter jug and a "b" liter jug. The jugs do not have markings to allow measuring smaller quantities. How canyou use the jugs to measure "c" liter of water?

Representing the problem in equation: ax + by = c where

x = number of time one jug is poured/discharged(+ve/-ve)

y = number of time the other jug is poured/discharged(+ve/-ve)

Theorem. The Diophantine equation ax+by = c is solvable

if and only if gcd(a, b) divides c.

Corner case :c must satisfy: c <= max(a,b)

Simulation:

You must simulate by doing "pouring action" only in one jug and "discharging action" only in the opposite jug.

(1)pour in a, discharge in b: if a is empty pour a full, if b is full the discharge and make b empty. Otherwise, transfer water from a to b. Each of those three condition is a step & can happen only once at a time.

(2)pour in b, discharge in a: if b is empty pour it full, if a is full make it empty. Otherwise, transfer water from b to a.

count total steps for (1) and (2), the minimum is the answer.

kushalanand (/users/kushalanand): 2016-10-29 23:37:15
My 69th. BFS + maps :D

albus111 (/users/albus111): 2016-09-16 07:23:49
Is there a math formula for this question?

conquistador (/users/conquistador): 2016-08-12 20:28:31
My 100th.. .Beginners believe in yourself . YOU CAN SOLVE IT. Dont rely on spojtoolkit for this question.

Last edit: 2016-08-12 21:13:54
Rajat Sharma (/users/rjtsh): 2016-08-03 20:26:02
AC Java : GCD along with logic(if conditions)
after solving this problem , one ll know : euclidean algorithm, extended euclidean algorithm, and most importantly this R (i+1) = R (i-1) - Q (i) R (i) which provides the synopsis for the extended euclidean algorithm.

Last edit: 2016-08-03 20:32:31
surayans tiwari(http://bit.ly/1EPzcpv) (/users/terminated): 2016-07-10 12:47:45
0.29 sec using bfs+map and 0.00 using eea,, nice question, worth the time;

Leave a Comment

Publish

Notes:
1. Don't post any source code here.
2. Please be careful, leave short comments only. Don't spam here.
3. For more discussion (hints, ideas, solutions) please visit our forum (/forum).
4. Authors of the problems are allowed to delete the post and use html code here (e.g. to provide some useful links).

Submit solution! (/submit/POUR1/)

Added by:	adrian (/users/adrian)
Date:	2004-05-31
Time limit:	1s
Source limit:	50000B
Memory limit:	1536MB
Cluster:	Cube (Intel G860) (/clusters/)
Languages:	All except: NODEJS PERL6 VB.NET
Resource:	An ancient problem, formulated in these words by Mr Tadeusz Ratajczak

Vote requirements

- ✔ be spoj user for at least 5 days
- ✔ solve at least 15 problems
- ✖ solve this problem