

05 Hr **59** Min
14 Sec

Guidelines

Coding Area

Editor | Compile &
Run History

Submissions

Feedback Form

Result

Graphs

Coding Area

A

B

C

D

E

F

ONLINE EDITOR (B)

Number Gravity

+ Problem Description

Prior to discovery of laws of gravitation, a brave young scientist attempted decoding the how two natural bodies "move" or "accelerate" towards each other. Later Newton, formulated the laws and the equations. Your task is to compute some metrics when two bodies move towards each other with postulates that the young scientist hypothesized.

Newton's law of universal gravitation from classical physics states that the gravitational force between two masses is proportional to the product of the masses and inversely proportional to the square of the distance between them. Imagine a scenario where a similar law applies to two specified natural numbers (positive integers) that "move" or "accelerate" towards each other on the number line. Let us explain with an example: Let us say we start with the following two different natural numbers:

$$n1 = 11$$

$$n2 = 22$$

$$n2 > n1$$

We will assume that the two numbers are initially at rest at their respective positions on the number line at positions $L1 = 11$ and $L2 = 22$ units from zero (the origin).

We calculate the initial "force" between them as a constant 10 times their product divided by the square of their difference: $F = 10 \cdot 11 \cdot 22 / (22 - 11)^2 = 20$ units.

The initial "acceleration" of $n1$ towards $n2$ would be $a1 = F/n1 = 1.82$ unit while the initial "acceleration" of $n2$ towards $n1$ would be $a2 = F/n2 = 0.91$ unit.

With initial velocities $u1$ and $u2$ being zero,

the "distance" moved by $n1$ and $n2$ in unit time would be

$$s1 = 0 + a1/2 = 1.82/2 = 0.91 = 1 \text{ unit (rounded using [HALF_UP semantics](#))}$$

$$s2 = 0 + a2/2 = 0.91/2 = 0.45 = 0 \text{ unit (rounded using [HALF_UP semantics](#))}$$

The new velocities after unit time would be

$$v1 = u1 + a1 = 0 + 1.82 = 1.82 \text{ units and}$$

$$v2 = v2 + a2 = 0 + 0.91 = 0.91 \text{ unit.}$$

Hence after a unit of time, n1 would be at location 12 with a velocity of 1.82 units and n2 would be near location 22 itself with a velocity of 0.91 units. We can proceed in a similar way i.e.

Note the existing positions and velocities,

Calculate the revised force, accelerations, new positions, new velocities

Track the movements of the two numbers as follows:

| t | l1 | l2 | l2-l1 | F | u1 | u2 | a1 | a2 | s1 | s2 | v1 | v2 |
|---|------------|------------|-------|-------|-------|------|------|------|-------|------|-------|------|
| 0 | 11 (n1) | 22 (n2) | 11 | 20.00 | 0.00 | 0.00 | 1.82 | 0.91 | 0.91 | 0.45 | 1.82 | 0.91 |
| 1 | 12 | 22 | 10 | 24.20 | 1.82 | 0.91 | 2.20 | 1.10 | 2.92 | 1.46 | 4.02 | 2.01 |
| 2 | 15 | 21 | 6 | 67.22 | 4.02 | 2.01 | 6.11 | 3.06 | 7.07 | 3.54 | 10.13 | 5.06 |
| 3 | 22 | 17 | -5 | 96.80 | 10.13 | 5.06 | 8.80 | 4.40 | 14.53 | 7.26 | 18.93 | 9.46 |

Hence the two numbers n1 and n2 "meet" on the number line between time units 2 and 3 and the location could be taken as the average of positions $(15 + 22 + 21 + 17)/4 = 18.75$. The relevant equations for each unit time slice ($\Delta t = 1$) are:
Equation Set 1: $F = 10 \cdot n1 \cdot n2 / (l2 - l1)^2$ $a1 = F/n1$; $a2 = F/n2$ $s1 = u1 + a1/2$; $s2 = u2 + a2/2$ $v1 = u1 + a1$; $v2 = u2 + a2$ If it is found that the average of the last four positions exceeds the position of "crossing", then we could calculate intermediate positions by slicing the last time slice into two and using the following equations (for a half unit time slice, $\Delta t = 1/2$):
Equation Set 2: $F = 10 \cdot n1 \cdot n2 / (l2 - l1)^2$ $a1 = F/n1$; $a2 = F/n2$ $s1 = u1/2 + a1/8$; $s2 = u2/2 + a2/8$ $v1 = u1 + a1/2$; $v2 = u2 + a2/2$ (Note: It may be necessary to calculate intermediate positions by slicing again, $\Delta t = 1/4$ or $3/4$) and use the appropriate motion equations.) Consider $n1 = 11$ and $n2 = 40$:

| t | l1 | l2 | l2-l1 | F | u1 | u2 | a1 | a2 | s1 | s2 | v1 | v2 |
|---|------------|------------|-------|------|------|------|------|------|------|------|------|------|
| 0 | 11 (n1) | 40 (n2) | 29 | 5.23 | 0.00 | 0.00 | 0.48 | 0.13 | 0.24 | 0.07 | 0.48 | 0.13 |
| 1 | 11 | 40 | 29 | 5.23 | 0.48 | 0.13 | 0.48 | 0.13 | 0.71 | 0.20 | 0.95 | 0.26 |
| 2 | 12 | 40 | 28 | 5.61 | 0.95 | 0.26 | 0.51 | 0.14 | 1.21 | 0.33 | 1.46 | 0.40 |
| 3 | 13 | 40 | 27 | 6.04 | 1.46 | 0.40 | 0.55 | 0.15 | 1.74 | 0.48 | 2.01 | 0.55 |
| 4 | 15 | 40 | 25 | 7.04 | 2.01 | 0.55 | 0.64 | 0.18 | 2.33 | 0.64 | 2.65 | 0.73 |

| | | | | | | | | | | | | |
|---|----|----|-----|--------|-------|------|-------|------|-------|------|-------|------|
| 5 | 17 | 39 | 22 | 9.09 | 2.65 | 0.73 | 0.83 | 0.23 | 3.06 | 0.84 | 3.48 | 0.96 |
| 6 | 20 | 38 | 18 | 13.58 | 3.48 | 0.96 | 1.23 | 0.34 | 4.09 | 1.13 | 4.71 | 1.30 |
| 7 | 24 | 37 | 13 | 26.04 | 4.71 | 1.30 | 2.37 | 0.65 | 5.89 | 1.62 | 7.08 | 1.95 |
| 8 | 30 | 35 | 5 | 176.00 | 7.08 | 1.95 | 16.00 | 4.40 | 15.08 | 4.15 | 23.08 | 6.35 |
| 9 | 45 | 31 | -14 | 22.45 | 23.08 | 6.35 | 2.04 | 0.56 | 24.10 | 6.63 | 25.12 | 6.91 |
| | | | | | | | | | | | | |

We find that the average of positions $(30+35+45+31)/4 = 35.25$ exceeds 35, the last position of $n2$. Hence to obtain the meeting point more accurately, we recalculate the parameters at $t=8$ and $t = 8.5$ as follows:

| | | | | | | | | | | | | |
|-----|----|----|----|--------|-------|------|-------|-------|-------|-------|-------|-------|
| 8.5 | 36 | 33 | -3 | 488.89 | 15.08 | 4.15 | 44.44 | 12.22 | 37.30 | 10.26 | 59.52 | 16.37 |
| 8 | 30 | 35 | 5 | 176.00 | 7.08 | 1.95 | 16.00 | 4.40 | 5.54 | 1.52 | 15.08 | 4.15 |

Hence we can surmise that the two numbers 11 and 40 meet at the location $(30 + 36 + 35 + 33)/4 = 33.50$. Write a program to accept two natural numbers, carry out the calculations as above and output their meeting location (up to 2 decimal places of accuracy using Rounding down method) under the influence of "gravity".

+ Constraints

$$1 \leq L1 < L2 \leq 50$$

+ Input Format

First Line contains two Integer Numbers $L1$ $L2$ delimited by space

+ Output

Meeting Point up to 2 decimal places of accuracy using Rounding down method.

+

+ Explanation

Example 1

Input

11 22

Output

18.75

Example 2

Input

11 40

Output

33.50

Upload Solution [Question : B]

☐ I, **shivansh kumar** confirm that the answer submitted is my own.☐ Took help from online sources (attributions)

Choose a
File ...

[CodeVita FAQ's](#)
[CodeVita Blog](#)
[Privacy Policy](#)
[Careers](#)

CONNECT WITH US



© 2018 Tata Consultancy Services Limited. All Rights Reserved.

