Practice Arena

Practice problems aimed to improve your coding skills.

- PRACTICE-02_SCAN-PRINT
- PRACTICE-03_TYPES
- LAB-PRAC-02_SCAN-PRINT
- LAB-PRAC-01
- PRACTICE-04_COND
- BONUS-PRAC-02
- LAB-PRAC-03_TYPES
- PRACTICE-05_COND-LOOPS
- LAB-PRAC-04_COND
- LAB-PRAC-05_CONDLOOPS
- PRACTICE-07_LOOPS-ARR
- LAB-PRAC-06_LOOPS
- LAB-PRAC-07_LOOPS-ARR
- LABEXAM-PRAC-01_MIDSEM
- PRACTICE-09_PTR-MAT
- LAB-PRAC-08_ARR-STR
- PRACTICE-10_MAT-FUN
- LAB-PRAC-09_PTR-MAT
- LAB-PRAC-10_MAT-FUN
- PRACTICE-11_FUN-PTR
- LAB-PRAC-11_FUN-PTR
- LAB-PRAC-12_FUN-STRUC
- LABEXAM-PRAC-02_ENDSEM
- LAB-PRAC-13_STRUC-NUM
  - Too tired to create a story - part I
  - Too tired to create a story - part II
  - Too tired to create a story - part III
  - Point Proximity
  - The Bisection Method
  - The pace is too fast
  - A Question on Quadrilaterals
  - The Trapezoidal Technique
  - Constrained Candy Crush
  - Major Mobile Madness
  - The Newton Raphson Method
  - The Palindrome Decomposition
- LAB-PRAC-14_SORT-MISC

The Trapezoidal Technique

LAB-PRAC-13_STRUC-NUM

The Trapezoidal Technique [20 marks]

Problem Statement
Mr C has just learnt the trapezoidal technique for calculating the area under a curve. Given a function $f : \mathbb{R} \to \mathbb{R}$, two points $a, b \in \mathbb{R}$ such that $a \le b$ and a resolution parameter n, we first chop up the interval $[a, b]$ into n equal pieces (n will always be a strictly positive integer) in the following manner.

Define $\Delta x = \frac{b-a}{n}$ and let $x_i := a + i \cdot \Delta x$. The first piece is $[x_0, x_1] = [a, a + \Delta x]$, the second piece is $[x_1, x_2] = [a + \Delta x, a + 2\Delta x]$ and so on. On every piece, say $[x_i, x_{i+1}]$ we can define a trapezoid with heights at the end points as $f(x_i)$ and $f(x_{i+1})$ respectively.

The area under the curve in the interval $[a, b]$ is approximated by adding up the areas of all the trapezoids. Our function will always be a cubic polynomial of the form

$$f(x) = p \cdot x^3 + q \cdot x^2 + r \cdot x + s$$

The input will first give you p, q, r, s as integers on the first line, all separated by a space. Then we will give you a, b as integers on the second line, separated by a space. Finally we will give you n as a strictly positive integer on the third line. Your output should be the area of the curve from a to b, calculated as above. Give your output rounded off to 4 decimal places using the %0.4lf format specifier in printf. Use double variables for all calculations.

Caution

1. Even though your inputs are integers, your output is not an integer. Use double variables for all calculations.
2. Do not try to cheat by doing the area calculation yourself using a definite integral and printing it directly. We are looking for the area given by the trapezoidal method which will have errors depending on the resolution errors. A definite integral will have no errors and hence will not match the expected output.
3. Area under the x axis is considered negative area as usual.
4. In case you get a situation where $f(x_i) < 0$ and $f(x_{i+1}) < 0$, count that entire trapezium as negative area.
5. In case $f(x_i) < 0$ and $f(x_{i+1}) > 0$ (or the other way round), your "trapezium" will look like a combination of two triangles, one below the x axis and one above the x axis. The area of the trapezium in this case will be found by adding the area of the triangle above the x axis and subtracting the area of the triangle below the x axis.
6. There is only one line in your output with no spaces.

EXAMPLE:
INPUT
0 1 0 0
-2 2
5

OUTPUT:
5.7600

Explanation: the function is $f(x) = x^2$. Note that the exact integral is 5.3333... but the expected output is 5.7600 since the trapezoidal method makes mistakes due to low resolution (5 is a very small number). The trapezoidal method output will approach the true integral output as the number 5 is increased to larger values like 5000 or so.

Grading Scheme:
Total marks: **[20 Points]**

There will be no partial grading in this question. An exact match will receive full marks whereas an incomplete match will receive 0 points. Please be careful of missing/extra spaces and missing/lines (take help of visible test cases). Each visible test case is worth 1 point and each hidden test case is worth 2 points. There are 2 visible and 4 hidden test cases.

Start Solving! (/editor/practice/6262)