Lab 01
Programming Overview

1

Content

Algorithm is the core of an application.

Flowchart is a common way to describe an algorithm. (Another way is pseudo-code)

6 fundamental symbols:

- "Start": a rounded rectangle
- "Input": a parallelogram
- "Process": a rectangle
- "Output": a parallelogram
- "End": a rounded rectangle
- And a line with arrow to connect all above elements. In this lab, we practice to draw several basic flowchart.

2

Basic flowchart understanding

Programming is the process to write a program / an application.

Specifically, programming includes 4 steps:

- Step1: Understanding your problem.

- Step 2: Designing an algorithm.

- Step 3: Writing source code.

- Step 4: Packaging source code files into an executable file, a website, an apk, or an ipa file...

You may think step 3 is the most important step. But, actually, step 2 is the most one.

Without step 2, you don't know how to solve a problem manually. Then, you don't know to teach your computer how to solve it, by writing source code.

For example, do you know an algorithm to find the shortest path from 2 specific locations? If you don't, how can to implement a C++ / Python source code to find the shortest path?

There are 2 ways to express an algorithm: drawing a flowchart, or writing a pseudo-code.

Here is an example of a flowchart to solve a problem: compute the sum of 2 integers a, b.

(Students can you **draw.io** application to create flowchart)
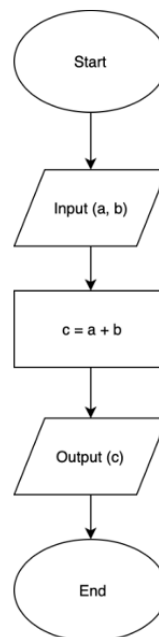
3

Requirements

In-class assignment: P02, P05, P13.

Homework: P10, P12, P16, P17, P23, P24, P25.

(Students need to check 'W01-Problems.pdf' file for Px)

3.1. P02

Write a program that inputs two integers. Calculate the sum of these two integers and print the result on the screen.

Input data:

A single line containing two integers a and b, separated by a space.

Where $-10^9 <= a, b <= 10^9$.

Output data:

The sum of the two integers, in the format a + b = c.

Example:

Input	Output
3 5	3+5=8

3.2. P05

Write a program that inputs two integers. Divide the first number by the second and print the result on the screen.

Input data:

A single line containing two integers a and b, separated by a space.

Where $-10^9 <= a, b <= 10^9$.

Output data:

The result is in the format a / b = c, rounded to 2 decimal places.

Example:

Input	Output
9 2	9 / 2 = 4.50

3.3. P13

Write a program that inputs the lengths of the three sides of a valid triangle. Calculate the perimeter and area of that triangle and print them on the screen.

Input data:

A single line containing three positive real numbers, edge1, edge2, edge3, representing the lengths of the three sides of the triangle.

Where $0 <$ edge1, edge2, edge3 $<= 10^9$.

Output data:

The perimeter and area of the triangle on the same line, rounded to two decimal places.

Example:

Input	Output
3.0 4.0 5.0	12.00 6.00

3.4. P10

Write a program that inputs the electricity meter reading from the previous month and the current month. Calculate the number of KWh we have consumed and print the result on the screen.

Input data:

A single line containing two positive integers, previous and current, which are the electricity meter readings for the previous and current months, separated by a space.

Where $0 <=$ previous $<=$ current $<= 10^9$.

Output data:

The consumed electricity reading.

Example:

Input	Output
1000 1211	211

3.5. P12

Write a program that inputs the quantity and unit price of a product.

Calculate the total amount to be paid = cost of goods + tax.

        Cost of goods = quantity * unit price.

        Tax = 10% of the cost of goods.

Input data:

Line 1: a positive integer, quantity, representing the quantity of the product.

Line 2: a positive real number, price, representing the unit price of the product.

Where -10^9 <= quantity, price <= 10^9.

Output data:

The total amount to be paid, with 0 decimal places.

Example:

Input	Output
7 30000	231000

3.6. P16

Write a program that inputs a Vehicle registration plate, which is a positive integer with 5 digits. Calculate the last digit of the sum of the digits (called "nut") of that license plate number

Input data:

A positive integer plate with 5 digits, where 10000 <= plate <= 99999.

Output data:

The sum of the digits (nut).

Example:

Input	Output
12345	5

3.7. P17

Write a program that inputs a positive integer money, which is an even number in the thousands. Consider the following banknote denominations: 500.000; 200.000; 100.000; 50.000; 20.000; 10.000; 5000; 2000; and 1000. Using a method that prioritizes higher denominations first, print the number of each banknote needed for the given amount.

Input:

A single positive integer money, representing the amount to be exchanged, where 0 < money <= 10^9.

Output:

9 lines, each indicating the denomination and the number of notes of that denomination, in the format: denomination: number of notes.
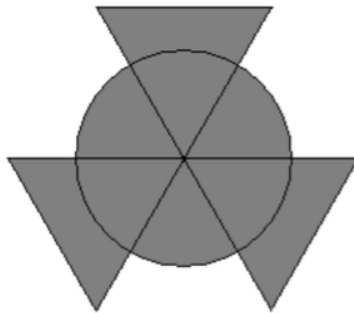
Example:

Input	Output
2361000	500.000: 4

200.000: 1 100.000: 1 50.000: 1 20.000: 0 10.000: 1 5000: 0 2000: 0 1000: 0

3.8. P23

Write a program that inputs the side length of an equilateral triangle and the radius of a circle. Calculate the total area of the gray region.

Use the convention PI = 3.14

Input Data:

A single line containing 2 positive real numbers, edge and radius, separated by a space.

Where $0 <$ edge, radius $<= 10^9$.

Output Data:

A single number, area, representing the area of the shaded region, rounded to 2 decimal places.

Example:

Input	Output
3 2	10.17

3.9. P24

A promotional program offers a 40% cashback on all transactions, with a maximum cashback limit of 100,000 VND. Determine how much a user should spend to receive the maximum cashback amount.

Input Data:

A single line containing 2 positive real numbers, percent and quota, separated by a space.

Where $0 <$ percent, quota $<= 10^9$.

Output Data:

A single number representing the amount to spend to receive the maximum cashback, rounded to 2 decimal places.

Example:

Input	Output
40 100000	250000

3.10. P25

On 12-10-2019, Eliud Kipchoge completed 42.195 km in 1 hour, 59 minutes, and 40.2 seconds, becoming the first person to run a marathon distance of 42 km in under 2 hours.

Write a program to calculate the pace (minutes per km) and speed (km/h) with given the distance (km) and the running time (hours, minutes, seconds).

Input Data:

A single line containing 4 positive real numbers: km, hour, minute, and second, separated by spaces.

Where $0 < $ km, hour, minute, second $<= 10^9$.

Output Data:

A single line with two numbers representing the pace and speed, each rounded to 2 decimal places.

Example:

Input	Output
42.195 1 59 40.2	21.16 2.84