

# THỰC HÀNH

## VIẾT CHƯƠNG TRÌNH ĐƠN GIẢN

**Học xong bài này, em sẽ:**

- ✓ Viết và thực hiện được một vài chương trình Python đơn giản với dữ liệu nhập vào từ bàn phím.
- ✓ Sử dụng được một vài hàm toán học do Python cung cấp.
- ✓ Nhận biết được chú thích trong một chương trình Python.

### Bài 1. Giải phương trình bậc nhất

Chương trình ở *Hình 1a* được viết để giải phương trình bậc nhất  $ax + b = 0$ , với  $a$ ,  $b$  là hai số thực nhập từ bàn phím ( $a \neq 0$ ) và nghiệm được thông báo ra màn hình. Tuy nhiên, chương trình đó còn viết thiếu ở những vị trí “...”. Em hãy hoàn thiện chương trình và kiểm thử xem với dữ liệu vào  $a = 1$  và  $b = 2$ , chương trình em vừa hoàn thiện có cho kết quả giống như *Hình 1b* không?

<p style="text-align: center;"><b>Chương trình</b></p> <pre style="font-family: monospace; border: 1px solid #ccc; padding: 10px;">File Edit Format Run Options Window Help a = float(input("a = ")) b = ... print("Nghiệm của phương trình là ",...)</pre>	<p style="text-align: center;"><b>Ví dụ chạy chương trình với <math>a = 1</math>, <math>b = 2</math></b></p> <pre style="font-family: monospace; border: 1px solid #ccc; padding: 10px;">File Edit Shell Debug Options Window Help a = 1 b = 2 Nghiệm của phương trình là -2.0 &gt;&gt;&gt;</pre>
---	---

*Hình 1a.*

Chương trình giải phương trình bậc nhất

*Hình 1b.*

Kết quả một lần chạy chương trình ở *Hình 1a*

Chương trình sẽ đưa ra màn hình thông tin gì nếu nhập vào giá trị  $a = 0$ ?

### Bài 2. An ninh lương thực

Trung bình mỗi người dân cần có  $a$  kg gạo để ăn, chế biến và phục vụ chăn nuôi trong một năm. Để đảm bảo an ninh lương thực, tổng số gạo dự trữ trong các kho của nhà nước chia cho đầu người phải lớn hơn hoặc bằng  $a$  kg.

Một nước có số dân là  $b$  thì cần dự trữ tối thiểu bao nhiêu ki-lô-gam gạo? Em hãy viết chương trình nhập từ bàn phím hai số  $a$ ,  $b$  và đưa ra màn hình khối lượng gạo tối thiểu cần dự trữ.

*Yêu cầu:* Cần đưa ra màn hình hướng dẫn nhập dữ liệu và thông báo kết quả bằng tiếng Việt có dấu.

*Ví dụ:*

INPUT	OUTPUT
$a = 365$ $b = 91086294$	Số gạo cần dự trữ: 33246497310

### Bài 3. Tìm ước chung lớn nhất

Em hãy viết chương trình nhập vào từ bàn phím hai số nguyên  $a$  và  $b$ , tính và đưa ra màn hình ước chung lớn nhất của hai số đó.

*Gợi ý:* Hãy tìm hiểu một số hàm toán học thường dùng trong Python.

*Ví dụ:*

INPUT	OUTPUT
$a = 9855$ $b = 11556$	Ước chung lớn nhất: 27

#### Một số hàm toán học thường dùng

Để hỗ trợ cho người dùng trong các chương trình tính toán, mỗi ngôn ngữ lập trình bậc cao đều cung cấp sẵn nhiều hàm toán học. Các hàm tính toán có sẵn như vậy thường được lưu trữ trong một thư viện thuộc hệ thống lập trình của ngôn ngữ bậc cao đó.

Trong Python, các hàm toán học lưu trữ trong thư viện **math**. *Bảng 1* nêu một số hàm toán học thường dùng.

*Bảng 1. Các hàm toán học thường dùng*

Hàm	Ý nghĩa toán học
<b>abs (x)</b>	Tính $ x $
<b>ceil (x)</b>	Trả về số nguyên nhỏ nhất, lớn hơn hoặc bằng giá trị $x$
<b>gcd (x, y)</b>	Tính ước chung lớn nhất của số nguyên $x$ và $y$
<b>sqrt (x)</b>	Tính căn bậc hai của $x$
<b>log (x)</b>	Tính $\ln x$
<b>exp (x)</b>	Tính $e^x$

Có thể biết chi tiết về các hàm qua trang web chính thức của Python.

<https://www.python.org/>

Hàm **abs ()** có thể sử dụng trực tiếp. Với các hàm còn lại như **ceil ()**, **gcd ()**,... ta cần đưa vào chương trình câu lệnh **import math** trước khi gọi hàm lần đầu tiên. Thông thường câu lệnh này được viết ngay ở đầu chương trình (*Hình 2*).

Lời gọi tới hàm có dạng: **math .<tên\_hàm>**

```

File Edit Shell Debug Options Window Help
>>> import math <-
>>> x = math.sqrt(5) <-
>>> x
2.236067977499979 <-
>>> y = 16/3
>>> y
5.333333333333333 <-
>>> z = math.ceil(16/3) <-
>>> z
6 <-

```

Câu lệnh sử dụng thư viện hàm toán học  
Lời gọi hàm tính căn bậc hai của 5  
Kết quả tính căn bậc hai của 5  
Kết quả tính  $16/3$   
Lời gọi hàm trả về số nguyên nhỏ nhất  
lớn hơn hoặc bằng giá trị của  $16/3$   
Kết quả làm tròn  $16/3$  bằng hàm ceil

Hình 2. Ví dụ một chương trình sử dụng hàm trong Python

#### Bài 4. Làm quen với ghi chú thích trong chương trình

Em hãy soạn thảo rồi chạy thử chương trình ở *Hình 3* sau đây trong hai trường hợp là có chú thích và không có chú thích. Em có nhận xét gì khi so sánh kết quả thực hiện chương trình trong hai trường hợp nêu trên.

##### Tìm hiểu về ghi chú thích trong chương trình

Khi soạn thảo chương trình, ngoài các câu lệnh, người lập trình có thể viết thêm các dòng chú thích. Các dòng chú thích không ảnh hưởng đến nội dung chương trình mà chỉ giúp cho người đọc nhanh chóng biết được mục đích của các câu lệnh và ý nghĩa của chương trình. Trong Python, thông tin chú thích viết trên một dòng, bắt đầu bằng kí tự **#**. Nhờ kí tự đánh dấu đó mà máy tính nhận biết được dòng chú thích.

```

File Edit Format Run Options Window Help
#Giải phương trình bậc hai <-
import math
a = 1
b = -5
c = 6
x1 = (-b - math.sqrt(b*b-4*a*c)) / (2*a)
x2 = -b/a - x1 #Định lí Viet <-
print ( x1 )
print ( x2 )

```

Chú thích cho biết mục đích  
của chương trình  
Chú thích cho biết kiến thức  
câu lệnh sử dụng

Hình 3. Ghi chú thích trong chương trình



Viết chương trình tính và đưa ra màn hình vận tốc v (m/s) khi chạm mặt đất của một vật rơi tự do từ độ cao h, biết rằng  $v = \sqrt{2gh}$ , trong đó g là gia tốc trọng trường ( $g \approx 9.8 \text{ m/s}^2$ ). Độ cao h tính theo mét được nhập từ bàn phím.