

Học xong bài này, em sẽ:

- ✓ Chạy và kiểm thử được chương trình.
- ✓ Rèn luyện được kĩ năng viết chương trình có khai báo và gọi hàm.
- ✓ Tìm hiểu và sử dụng được hàm time có trong thư viện.

Bài 1. Giải phương trình

Chương trình cho trong *Hình 1* nhằm tạo một bảng chọn việc, để người chạy chương trình chọn cho máy tính giúp giải phương trình bậc nhất hay giải phương trình bậc hai. Em hãy đưa khai báo của các hàm thực hiện hai việc nói trên và các lời gọi chúng vào đúng chỗ trong chương trình. Sau đó hãy chạy thử chương trình với một số dữ liệu đầu vào khác nhau để kiểm thử chương trình.

Bài 2. Thời gian gặp nhau

Hiện tại, anh trai Khánh Nam đang ở thành phố A còn em gái Sương Mai đang ở thành phố B. Khoảng cách giữa hai thành phố đó là d km. Hai anh em đi ô tô xuất phát

cùng một thời điểm từ hai thành phố, ô tô khởi hành từ A đi về B với tốc độ không đổi v_1 km/h, ô tô khởi hành từ B đi đến A với tốc độ không đổi v_2 km/h; trong đó d, v_1, v_2 là các số thực. Chương trình ở *Hình 2* khai báo hàm mtime với các tham số d, v_1, v_2 để xác định thời gian hai ô tô gặp nhau tính từ lúc xuất phát. Em hãy:

- a) Hoàn thiện chương trình ở *Hình 2* bằng cách bổ sung cho chương trình lời gọi hàm mtime với dữ liệu nhập từ bàn phím.
- b) Chạy chương trình và chạy thử chương trình với ít nhất hai bộ dữ liệu vào khác nhau.

Hướng dẫn: Viết hàm mtime với tham số d, v_1, v_2 và trả về thời gian gặp nhau $\frac{d}{v_1 + v_2}$.

```
File Edit Format Run Options Window Help

# Khai báo hàm Giải pt bậc nhất GPTB1
# Khai báo hàm Giải pt bậc hai GPTB2
# Tạo bảng chọn việc
while True:
    print ( "*****" )
    print ( ' BẢNG CHỌN VIỆC ' )
    print ( '1. Giải phương trình bậc nhất ' )
    print ( '2. Giải phương trình bậc hai ' )
    print ( '3. Thoát khỏi công việc ' )
    print ( "*****" )
    chon = input( ' Hãy chọn (1 hay 2 hay 3): ' )
    if chon == "1":
        print ( ' Giải phương trình bậc nhất ' )
        # lời gọi hàm GPTB1

    elif chon == "2":
        print ( ' Giải phương trình bậc hai ' )
        # lời gọi hàm GPTB2

    else:
        print ( ' Tạm biệt ' )
        break
```

Hình 1. Chương trình giải phương trình

```
File Edit Format Run Options Window Help
def mtime(d,v1,v2):
    t = v1+v2
    return d/t
d = float(input("d = "))
v1 = float(input("v1 = "))
v2 = float(input("v2 = "))
print(' Hai xe gặp nhau sau',
mtime(d,v1,v2), ' giờ.')
```

a) Chương trình

```
File Edit Shell Debug Options Window Help
d = 300
v1 = 70
v2 = 80
Hai xe gặp nhau sau 2.0 giờ.
>>>
```

b) Kết quả thực hiện

Hình 2. Ví dụ một chương trình cho bài toán thời gian gặp nhau

Bài 3. Thời gian thực hiện chương trình

Hàm `time` (với lời gọi `time()`) trong thư viện `time` cho biết thời gian tại thời điểm hiện tại (tính theo giây). Để biết thời gian thực hiện chương trình, người ta ghi nhận thời điểm lúc bắt đầu thực hiện chương trình, thời điểm lúc kết thúc chương trình và đưa ra hiệu các thời điểm đã xác định. Em hãy gắn hàm `time` từ thư viện `time` vào một số chương trình đã có của em và đưa ra thời gian thực hiện chương trình.

Hướng dẫn:

- Gắn thư viện `time` vào chương trình: `import time`
- Để ghi nhận thời điểm bắt đầu viết câu lệnh đầu tiên là: `tb = time.time()`
- Cuối chương trình, đưa ra thời gian thực hiện: `time.time() - tb`
- Để cho đẹp: Nên dùng quy cách `%.4f` để đưa ra thời gian thực hiện chương trình với bốn chữ số ở phần thập phân (Hình 3).

```
File Edit Format Run Options Window Help
import time
tb = time.time()
n = 0
s = 0
x = int(input())
while x > 0:
    n = n + 1
    s = s + x
    x = int(input())
if n > 0 : print('Trung bình cộng:',s/n)
print('\nTime: %.4f sec'%(time.time() -tb))
```

```
File Edit Shell Debug Options Window Help
5
6
4
0
Trung bình cộng: 5.0
Time: 9.9937 sec
>>>
```

Hình 3. Minh họa cách sử dụng hàm `time`



Viết chương trình vẽ một hình chữ nhật bằng các dấu `#` với một cạnh có độ dài bằng 10, một cạnh có độ dài bằng `a`. Ví dụ với `a = 4`, hình chữ nhật cần vẽ như hình bên:

```
#####
#####
#####
#####
```

Yêu cầu xây dựng một hàm `Drawbox` với tham số (`a`), hàm này đưa ra màn hình các dòng, mỗi dòng chứa 10 dấu `#` liên tiếp và tham số `a` quyết định số dòng sẽ được đưa ra. Chương trình gọi hàm `Drawbox(a)` với `a` nhập vào từ bàn phím.