

**Sở GD và ĐT Thành Phố Đà Nẵng**

**Trường THPT Đỗ Đăng Tuyển**

**Giáo viên soạn: Bùi Thị Tuyết**

**Ngày soạn: 22/03/2026**

**Thời gian thực hiện: tuần học 28**

**Lớp dạy: 10/2,10/4,10/6**

**CHỦ ĐỀ 5. GIẢI QUYẾT VẤN ĐỀ VỚI SỰ TRỢ GIÚP CỦA MÁY TÍNH**  
**BÀI 28. PHẠM VI CỦA BIẾN**

Môn học: Tin Học; Lớp: 10

Thời gian thực hiện: 3tiết (1LT+2TH)

**I. MỤC TIÊU**

**1. Kiến thức**

- Biết và trình bày được ý nghĩa của phạm vi hoạt động của biến trong chương trình và hàm.

**2. Về năng lực**

**2.1 Năng lực chung**

- Năng lực tự chủ tự học.
- Năng lực giao tiếp và hợp tác.
- Năng lực giải quyết vấn đề.

**2.2 Năng lực tin học**

- Năng lực tự chủ và tự học
- Năng lực giao tiếp và hợp tác
- Năng lực sáng tạo và giải quyết vấn đề

**2.3. Về phẩm chất**

- Nhân ái: Thể hiện sự cảm thông và sẵn sàng giúp đỡ bạn trong quá trình thảo luận nhóm.

- Chăm chỉ: Thực hiện đầy đủ các hoạt động học tập một cách tự giác, tích cực.

- Trung thực: Thật thà, thẳng thắn trong báo cáo kết quả hoạt động cá nhân và theo nhóm, trong đánh giá và tự đánh giá.

- Trách nhiệm: Hoàn thành đầy đủ, có chất lượng các nhiệm vụ học tập.

**2.4 Phát triển các năng lực số cốt lõi**

- 3.4.NC1a:Tự thao tác được bằng các hướng dẫn dành cho hệ thống máy tính để giải quyết một vấn đề khác hoặc thực hiện các nhiệm vụ khác nhau.

**II. THIẾT BỊ DẠY HỌC VÀ HỌC LIỆU**

**1. Thiết bị dạy học:** Máy tính, tivi (máy chiếu), bảng phụ (nếu có)

**2. Học liệu:**

- **Học sinh:** SGK Tin học 10, SBT Tin học 10, Vở ghi chép

- Chuẩn bị một số chương trình trong quá trình học chạy bị lỗi hoặc một số lỗi thường mắc phải khi học và chạy chương trình để thảo luận cùng nhóm hoặc hỏi GV cùng giải quyết.

- **Giáo viên:** Kế hoạch bài dạy, phiếu giao nhiệm vụ (phiếu học tập), phiếu đánh giá kết quả thực hiện của nhóm và một số tài liệu có liên quan.

### III. TIẾN TRÌNH DẠY HỌC

#### 1. Hoạt động 1: KHỞI ĐỘNG (5 phút)

a. **Mục tiêu:** Tạo hứng thú học tập cho học sinh

b. **Nội dung:** Hs dựa vào hiểu biết để trả lời câu hỏi

c. **Sản phẩm:** Từ yêu cầu Hs vận dụng kiến thức để trả lời câu hỏi GV đưa ra

d. **Tổ chức thực hiện:** GV giới thiệu và dẫn dắt vào bài

1. Một biến được định nghĩa trong chương trình chính (bên ngoài các hàm) thì sẽ được sử dụng như thế nào bên trong các hàm ?

2. Một biến được khai báo bên trong một hàm thì có sử dụng được ở bên ngoài hàm đó hay không?

Bài này sẽ giúp em tìm câu trả lời cho các câu hỏi trên

HS: trả lời câu hỏi

#### 2. HÌNH THÀNH KIẾN THỨC MỚI

##### 2.1 Phạm vi của biến khai báo trong hàm (15 phút)

a. **Mục Tiêu:**

- Biết và trình bày được ý nghĩa của phạm vi hoạt động của biến trong chương trình và hàm.

- 3.4.NC1a: Tự thao tác được bằng các hướng dẫn dành cho hệ thống máy tính để giải quyết một vấn đề khác hoặc thực hiện các nhiệm vụ khác nhau.

b. **Nội dung:** HS quan sát SGK để tìm hiểu nội dung kiến thức theo yêu cầu của GV

c. **Sản phẩm:** Hs hoàn thành tìm hiểu kiến thức

d **Tổ chức thực hiện:**

\* **Bước 1: Chuyển giao nhiệm vụ:**

**GV:** Nêu đặt câu hỏi

Quan sát các lệnh sau để tìm hiểu phạm vi có hiệu lực của biến khi khai báo bên trong một hàm.

**HS:** Thảo luận, trả lời hoàn thành **PHIẾU HỌC TẬP SỐ 1**

#### PHIẾU HỌC TẬP SỐ 1

CHƯƠNG TRÌNH	TRẢ LỜI
(1) >>> def func(a,b) :	Câu 1: Tên hàm .....
(2)       n = 10	Câu 2: Tham số .....
(3)       a = a * 2	Câu 3: Đối số .....
(4)       b = a+b	Câu 4: Giá trị trả lại .....
(5)       return a + b + n	Câu 5: Từ khóa .....
(6) >>> a = 1	Câu 6: Đoạn chương trình con .....
(7) >>> b = 2	Câu 7: Đoạn chương trình chính .....
(8) >>> func(a,b)	Câu 8: Tên biến cục bộ .....
(9) 16	Câu 9: Tên biến toàn cục.....
(10)>>> a,b	Câu 10: Câu lệnh gọi hàm .....
(11) (1,2)	Giá trị bằng bao nhiêu .....
(12)>>>n	Câu 11: Tên mã lỗi ngoại lệ là .....
Traceback (most recent call last):	Câu 12: Sau khi chạy hàm, các biến a=.....;b=.....
File "<pyshell#11>", line 1, in <module> n	Câu 13: Vì sao biến n bị báo lỗi ..... .....

NameError: name 'n' is not defined	Câu 12: Các biến được khai báo .....một hàm chỉ được sử dụng .....hàm đó. Chương trình chính ..... sử dụng được.
------------------------------------	--

**\* Bước 2: Thực hiện nhiệm vụ:**

- + HS: Suy nghĩ, tham khảo sgk trả lời câu hỏi
- + GV: quan sát và trợ giúp các cặp.

**\* Bước 3: Báo cáo, thảo luận:**

- + HS: Lắng nghe, ghi chú, một HS phát biểu lại các tính chất.
- + Các nhóm nhận xét, bổ sung cho nhau.

**\* Bước 4: Kết luận, nhận định:** GV chính xác hóa và gọi 1 học sinh nhắc lại kiến thức

Các biến được khai báo bên trong một hàm chỉ được sử dụng bên trong hàm đó. Chương trình chính không sử dụng được.

**Ghi nhớ:**

**Trong Python tất cả các biến khai báo bên trong hàm đều có tính địa phương (cục bộ), không có hiệu lực ở bên ngoài hàm.**

**2.2 Phạm vi của biến khai báo ngoài hàm (15 phút)**

**a. Mục Tiêu:**

- Nắm được cách dùng biến khai báo ngoài hàm.
- Biết và trình bày được ý nghĩa của phạm vi hoạt động của biến trong chương trình và hàm.

**b. Nội dung:** HS quan sát SGK để tìm hiểu nội dung kiến thức theo yêu cầu của GV

**c. Sản phẩm:** Hs hoàn thành tìm hiểu kiến thức

**d. Tổ chức thực hiện:**

**\* Bước 1: Chuyển giao nhiệm vụ:**

**GV:** Hoàn thành **PHIẾU HỌC TẬP SỐ 2**

Quan sát các lệnh sau, tìm hiểu phạm vi có hiệu lực của biến khi khai báo bên ngoài một hàm.

**PHIẾU HỌC TẬP SỐ 2**

Bài 1	Bài 2
<pre>(1) &gt;&gt;&gt; def f(n): (2)         t = n + 1 (3)         return t (4)&gt;&gt;&gt; t = 10 (5)&gt;&gt;&gt; f(5) (6)6 (7)&gt;&gt;&gt; t (8)10</pre>	<pre>(1)&gt;&gt;&gt; def f(a, b): (2)         return a + b + N (3)&gt;&gt;&gt; N = 10 (4)&gt;&gt;&gt; f(1, 2) (5)13</pre>
<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục.....</p> <p>Câu 3: Biến ở ngoài hàm có giá trị t =10. Sau khi chạy hàm biến t vẫn ..... (có/không) thay đổi. Biến t=10</p> <p>Câu 4: Biến khai báo bên ngoài hàm ..... (có/không) có tác dụng bên trong hàm.</p>	<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục.....</p> <p>Câu 3: Khi gọi hàm f(1,2) giá trị trả lại là biểu thức có ..... Vậy trong hàm f() được phép truy cập giá trị của biến N.</p> <p>Câu 4: Bên trong hàm có thể truy cập để sử dụng .....của biến đã khai báo trước đó ở bên ngoài hàm.</p>
Bài 3	Bài 4

<pre>(1)&gt;&gt;&gt; def f(n): (2)         t = 2*n + 1 (3)         return t (4)&gt;&gt;&gt; t = 10 (5)&gt;&gt;&gt; f(1) (6)3 (7)&gt;&gt;&gt; t (8)10</pre>	<pre>(1)&gt;&gt;&gt; def f(n): (2)         global t (3)         t = 2*n + 1 (4)         return t (5)&gt;&gt;&gt; t = 10 (6)&gt;&gt;&gt; f(1) (7)3 (8)&gt;&gt;&gt; t (9)3</pre>
<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục .....</p>	<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục .....</p> <p>Câu 3: Sự khác nhau của 2 chương trình trên? ..... ..... .....</p> <p>Câu 4: Biến đã khai báo bên ngoài sẽ không có tác dụng bên trong hàm như một biến. Nếu muốn có tác dụng thì cần khai báo lại biến này trong hàm với từ khóa .....</p>

**HS:** Thảo luận, trả lời

**\* Bước 2: Thực hiện nhiệm vụ:**

- + HS: Suy nghĩ, tham khảo sgk trả lời câu hỏi
- + GV: quan sát và trợ giúp các cặp.

**\* Bước 3: Báo cáo, thảo luận:**

- + HS: Lắng nghe, ghi chú, một HS phát biểu lại các tính chất.
- + Các nhóm nhận xét, bổ sung cho nhau.

**\* Bước 4: Kết luận, nhận định:** GV chính xác hóa và gọi 1 học sinh nhắc lại kiến thức

**Ghi nhớ:** Biến đã khai báo bên ngoài sẽ không có tác dụng bên trong hàm như một biến. Nếu muốn có tác dụng thì cần khai báo lại biến này trong hàm với từ khóa global.

**Hoạt động 3: THỰC HÀNH (80 phút)**

**a. Mục tiêu:** Rèn kỹ năng lập trình

**b. Nội dung:** HS quan sát SGK để tìm hiểu nội dung kiến thức theo yêu cầu của GV.

**c. Sản phẩm:** HS hoàn thành tìm hiểu kiến thức

**d. Tổ chức thực hiện:**

**\* Bước 1: Chuyển giao nhiệm vụ:**

**GV:** giao nhiệm vụ 1, nhiệm vụ 2

**HS:** Thảo luận, trả lời

**HS:** Lấy các ví dụ trong thực tế.

**\* Bước 2: Thực hiện nhiệm vụ:**

- + HS: Suy nghĩ, tham khảo sgk trả lời câu hỏi
- + GV: quan sát và trợ giúp các cặp.

**\* Bước 3: Báo cáo, thảo luận:**

- + HS: Lắng nghe, ghi chú, một HS phát

biểu lại các tính chất.

+ Các nhóm nhận xét, bổ sung cho nhau.

\* **Bước 4: Kết luận, nhận định:** GV chính xác hóa và gọi 1 học sinh nhắc lại kiến thức

**Nhiệm vụ 1.** Viết hàm với đầu vào là danh sách A chứa các số và số thực x. Hàm trả lại một danh sách kết quả B từ danh sách A bằng cách chỉ giữ lại các phần tử lớn hơn hoặc bằng x.

**Hướng dẫn.** Biến B kiểu danh sách cần được định nghĩa trong hàm và được bổ sung thêm các phần tử từ A nếu thỏa mãn điều kiện lớn hơn hoặc bằng x.

def **Select**( A, x ) :

    B = [ ]

    for k in range(len(A)):

        if A[k] >= x : B.append(A[k])

    return B

```
>>> def Select(A,x):
...     B=[]
...     for k in range(len(A)):
...         if A[k]>=x:
...             B.append(A[k])
...     return B
...
>>> S=[5,-5,10,15,-1,2,-6]
>>> T=[2.5,1.3,47,-6.7,0.25,-1,-3.75]
>>> Select(S,5)
[5, 10, 15]
>>> Select(T,0)
[2.5, 1.3, 47, 0.25]
>>>
```

**Nhiệm vụ 2.** Viết hàm với đầu vào là chuỗi ký tự Str và số c, đầu ra là danh sách các từ được tách ra từ chuỗi Str nhưng đã được chuyển thành chữ in hoa hoặc chữ in thường. Hoặc chỉ chuyển ký tự đầu các từ thành chữ in hoa tùy thuộc vào tham số đầu vào c như sau :

- Nếu c = 0, danh sách A là các từ được chuyển thành chữ in hoa.

- Nếu c = 1, danh sách A là các từ được chuyển thành chữ in thường.

- Nếu c = 2, danh sách A là các từ được chuyển viết chữ hoa ký tự đầu của mỗi từ.

**Hướng dẫn.** Chúng ta cần sử dụng các lệnh sau:

**Str.upper()** – chuyển ký tự của chuỗi thành chữ in hoa.

**Str.lower()** – chuyển ký tự của chuỗi thành chữ in thường.

**Str.title()** – chuyển ký tự đầu mỗi từ của chuỗi thành chữ in hoa, các ký tự khác chuyển về chữ thường

- Hàm được định nghĩa có dạng **Tách\_tu(Str, c)**. Đầu tiên chuỗi Str cần được tách từ bằng lệnh split(). Sau đó danh sách kết quả sẽ được chuyển đổi chữ in hoa, in thường sử dụng một trong các lệnh trên tùy thuộc vào giá trị của đối số c.

```

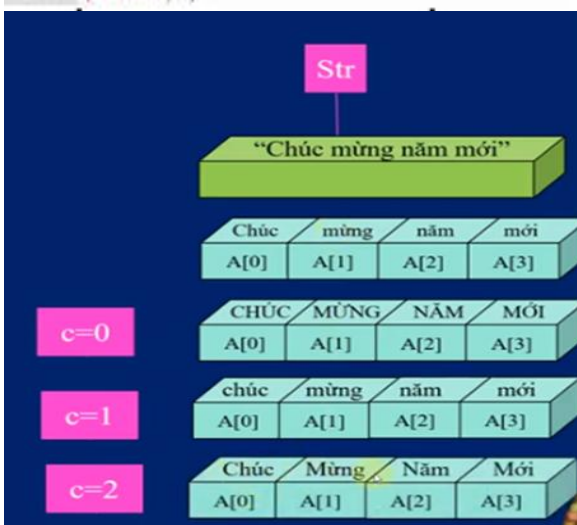
1 def Tach_tu(Str, c):
2     A = Str.split()
3     for k in range(len(A)):
4         if c == 0:
5             A[k] = A[k].upper()
6         if c == 1:
7             A[k] = A[k].lower()
8         if c == 2:
9             A[k] = A[k].title()
10    return A
11 s="hoàng thị thanh tâm"
12 c=int(input("nhập c = 0,1,2: "))
13 a=Tach_tu(s,c)
14 print(a)

```

```

r more information.
>>> def Tach_tu(Str,c):
...     A=Str.split()
...     for k in range(len(A)):
...         if c==0:
...             A[k]=A[k].upper()
...         if c==1:
...             A[k]=A[k].lower()
...         if c==2:
...             A[k]=A[k].title()
...     return A
...
>>> S="Chúc mừng năm mới"
>>> Tach_tu(S,0)
['CHÚC', 'MỪNG', 'NĂM', 'MỚI']
>>> Tach_tu(S,1)
['chúc', 'mừng', 'năm', 'mới']
>>> Tach_tu(S,2)
['Chúc', 'Mừng', 'Năm', 'Mới']
>>>

```



**Nhiệm vụ 3.** Viết chương trình yêu cầu thực hiện lần lượt các việc sau, mỗi việc cần được thực hiện bởi một hàm:

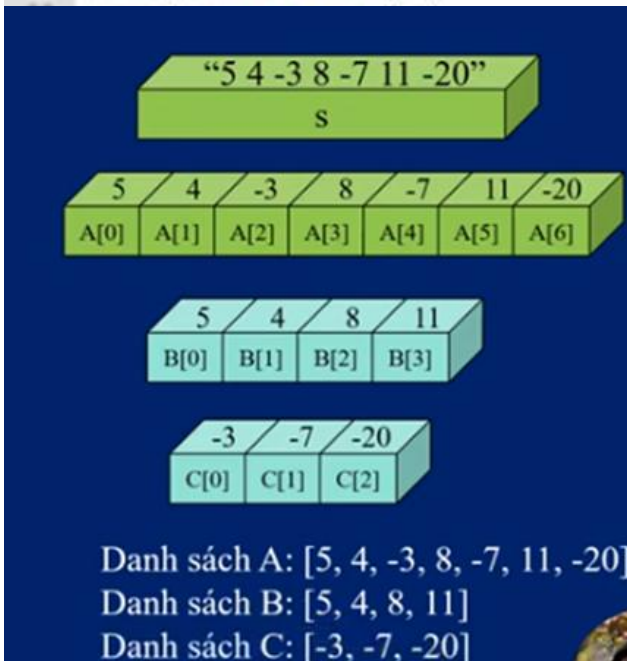
1. Nhập từ bàn phím một dãy các số nguyên, mỗi số cách nhau bởi dấu cách. Chuyển các số này vào danh sách A và in danh sách A ra màn hình.
2. Trích từ danh sách A ra một danh sách B gồm các phần tử lớn hơn 0. In danh sách B ra màn hình.
3. Trích từ danh sách A ra một danh sách C gồm các phần tử nhỏ hơn 0. In danh sách C ra màn hình.

**Hướng dẫn.** Với mỗi việc trên được viết thành một hàm. Toàn bộ chương trình có thể như sau:

```

1 t = 0
2 def Nhap_Dulieu():
3     s = input("Nhập các số nguyên cách nhau bởi dấu cách: ")
4     A = s.split()
5     for k in range(len(A)): A[k] = int(A[k])
6     return A
7 def getB(A):
8     B = []
9     for x in A:
10        if x > 0: B.append(x)
11    return B
12 def getC(A):
13    C = []
14    for x in A:
15        if x < 0: C.append(x)
16    return C
17 # Chương trình chính
18 A = Nhap_Dulieu()
19 print("Danh sách A:", A)
20 B = getB(A); C = getC(A)
21 print("Danh sách B:", B)
22 print("Danh sách C:", C)

```



### 3. HOẠT ĐỘNG LUYỆN TẬP (10 phút)

a. **Mục tiêu:** Củng cố, luyện tập kiến thức vừa học.

b. **Nội dung:** HS đọc SGK làm các bài tập.

c. **Sản phẩm:** Bài làm của học sinh, kỹ năng giải quyết nhiệm vụ học tập.

d. **Tổ chức thực hiện:**

Gv Cho HS nhắc lại KT:

Hs: Nhắc lại các vấn đề đã học

1. Viết hàm với đầu vào, đầu ra như sau:

- Đầu vào là danh sách slist, các phần tử là xâu kí tự.

- Đầu ra là danh sách clist, các phần tử là kí tự đầu tiên của các xâu kí tự tương ứng trong danh sách slist.

2. Viết hàm `Tach_day()` với đầu vào là danh sách A, đầu ra là hai danh sách B, C được mô tả như sau:

- Danh sách B thu được từ A bằng cách lấy ra các phần tử có chỉ số chẵn.

- Danh sách B thu được từ A bằng cách lấy ra các phần tử có chỉ số lẻ.

#### **4. HOẠT ĐỘNG VẬN DỤNG (10 phút)**

**a. Mục tiêu:** Vận dụng các kiến thức vừa học quyết các vấn đề học tập và thực tiễn.

**b. Nội dung:**

**c. Sản phẩm:** HS vận dụng các kiến thức vào giải quyết các nhiệm vụ đặt ra.

**d. Tổ chức thực hiện:**

Gv đưa câu hỏi về nhà:

**1.** Viết hàm có hai tham số đầu vào là m, n. Đầu ra trả lại hai giá trị là:

- ƯCLN của m, n.

- Bội chung nhỏ nhất (BCNN) của m, n.

*Gợi ý:* Sử dụng công thức  $ƯCLN(m, n) \times BCNN(m, n) = m \times n$

**2.** Viết chương trình nhập ba số tự nhiên từ bàn phím day, month, year, các số cách nhau bởi dấu cách. Các số này biểu diễn giá trị của ngày, tháng, năm nào đó. Chương trình cần kiểm tra và in ra thông báo số liệu đã nhập vào đó có hợp lệ hay không.

**5. Hướng dẫn học sinh tự học:**

- Hướng dẫn học bài cũ:

- Hướng dẫn chuẩn bị bài mới: Bài 29 NHẬN BIẾT LỖI CHƯƠNG TRÌNH



**PHỤ LỤC**  
**PHIẾU HỌC TẬP SỐ 1**

CHƯƠNG TRÌNH	TRẢ LỜI
(1) >>> def func(a,b) :	Câu 1: Tên hàm .....
(2)           n = 10	Câu 2: Tham số .....
(3)           a = a * 2	Câu 3: Đối số .....
(4)           b = a+b	Câu 4: Giá trị trả lại .....
(5)           return a + b + n	Câu 5: Từ khóa .....
(6) >>> a = 1	Câu 6: Đoạn chương trình con .....
(7) >>> b = 2	Câu 7: Đoạn chương trình chính .....
(8) >>> func(a,b)	Câu 8: Tên biến cục bộ .....
(9) 16	Câu 9: Tên biến toàn cục.....
(10)>>> a,b	Câu 10: Câu lệnh gọi hàm .....
(11) (1,2)	Giá trị bằng bao nhiêu .....
(12)>>>n	Câu 11: Tên mã lỗi ngoại lệ là .....
Traceback (most recent call last):	Câu 12: Sau khi chạy hàm, các biến a=.....;b=.....
File "<pyshell#11>", line 1, in <module> n	Câu 13: Vì sao biến n bị báo lỗi .....
NameError: name 'n' is not defined	..... Câu 12: Các biến được khai báo .....một hàm chỉ được sử dụng .....hàm đó. Chương trình chính ..... sử dụng được.

## PHIẾU HỌC TẬP SỐ 2

Bài 1	Bài 2
<pre>(1) &gt;&gt;&gt; def f(n): (2)         t = n + 1 (3)         return t (4)&gt;&gt;&gt; t = 10 (5)&gt;&gt;&gt; f(5) (6)6 (7)&gt;&gt;&gt; t (8)10</pre>	<pre>(1)&gt;&gt;&gt; def f(a, b): (2)         return a + b + N (3)&gt;&gt;&gt; N = 10 (4)&gt;&gt;&gt; f(1, 2) (5)13</pre>
<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục.....</p> <p>Câu 3: Biến ở ngoài hàm có giá trị t =10. Sau khi chạy hàm biến t vẫn ..... (có/không) thay đổi. Biến t=10</p> <p>Câu 4: Biến khai báo bên ngoài hàm ..... (có/không) có tác dụng bên trong hàm.</p>	<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục.....</p> <p>Câu 3: Khi gọi hàm f(1,2) giá trị trả lại là biểu thức có ..... Vậy trong hàm f() được phép truy cập giá trị của biến N.</p> <p>Câu 4: Bên trong hàm có thể truy cập để sử dụng .....của biến đã khai báo trước đó ở bên ngoài hàm.</p>
Bài 3	Bài 4
<pre>(1)&gt;&gt;&gt; def f(n): (2)         t = 2*n + 1 (3)         return t (4)&gt;&gt;&gt; t = 10 (5)&gt;&gt;&gt; f(1) (6)3 (7)&gt;&gt;&gt; t (8)10</pre>	<pre>(1)&gt;&gt;&gt; def f(n): (2)         global t (3)         t = 2*n + 1 (4)         return t (5)&gt;&gt;&gt; t = 10 (6)&gt;&gt;&gt; f(1) (7)3 (8)&gt;&gt;&gt; t (9)3</pre>
<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục .....</p>	<p>Câu 1: Tên biến cục bộ .....</p> <p>Câu 2: Tên biến toàn cục .....</p> <p>Câu 3: Sự khác nhau của 2 chương trình trên? ..... ..... .....</p> <p>Câu 4: Biến đã khai báo bên ngoài sẽ không có tác dụng bên trong hàm như một biến. Nếu muốn có tác dụng thì cần khai báo lại biến này trong hàm với từ khóa .....</p>

**ĐÁP ÁN**  
**PHIẾU HỌC TẬP SỐ 1**

CHƯƠNG TRÌNH	TRẢ LỜI
<pre>(1) &gt;&gt;&gt; def func(a,b) : (2)         n = 10 (3)         a = a * 2 (4)         b = a+b (5)         return a + b + n (6) &gt;&gt;&gt; a = 1 (7) &gt;&gt;&gt; b = 2 (8) &gt;&gt;&gt; func(a,b) (9) 16 (10)&gt;&gt;&gt; a,b (11) (1,2) (12)&gt;&gt;&gt;n Traceback (most recent call last): File "&lt;pyshell#11&gt;", line 1, in &lt;module&gt; n NameError: name 'n' is not defined</pre>	<p>Câu 1: Tên hàm: func          Câu 2: Tham số: a,b          Câu 3: Đối số: a,b          Câu 4: Giá trị trả lại: a+b+n          Câu 5: Từ khóa: def; return          Câu 6: Đoạn chương trình con: dòng 1 đến dòng5          Câu 7: Đoạn chương trình chính: dòng 6 đến dòng12          Câu 8: Tên biến cục bộ n          Câu 9: Tên biến toàn cục a,b          Câu 10: Câu lệnh gọi hàm: func(a,b)          Giá trị bằng bao nhiêu: 16          Câu 11: Tên mã lỗi ngoại lệ: NameError          Câu 12: Sau khi chạy hàm, các biến a=1,b=2 vẫn không thay đổi giá trị          Câu 13: Vì sao biến n bị báo lỗi: Biến n có tác dụng bên trong hàm func, gọi bên ngoài hàm này sẽ bị báo lỗi.          Câu 12: Các biến được khai báo bên trong một hàm chỉ được sử dụng bên trong hàm đó. Chương trình chính không sử dụng được.</p>

## PHIẾU HỌC TẬP SỐ 2

Bài 1	Bài 2
<pre>(1) &gt;&gt;&gt; def f(n): (2)         t = n + 1 (3)         return t (4) &gt;&gt;&gt; t = 10 (5) &gt;&gt;&gt; f(5) (6) 6 (7) &gt;&gt;&gt; t (8) 10</pre>	<pre>(1) &gt;&gt;&gt; def f(a, b): (2)         return a + b + N (3) &gt;&gt;&gt; N = 10 (4) &gt;&gt;&gt; f(1, 2) (5) 13</pre>
<p>Câu 1: Tên biến cục bộ <b>t</b></p> <p>Câu 2: Tên biến toàn cục <b>t</b></p> <p>Câu 3: Biến ở ngoài hàm có giá trị <math>t = 10</math>. Sau khi chạy hàm biến <b>t</b> vẫn <b>không</b> (có/không) thay đổi. Biến <math>t = 10</math>.</p> <p>Câu 4: Biến khai báo bên ngoài hàm <b>không</b> (có/không) có tác dụng bên trong hàm.</p>	<p>Câu 1: Tên biến cục bộ <b>N</b></p> <p>Câu 2: Tên biến toàn cục <b>N</b></p> <p>Câu 3: Khi gọi hàm <math>f(1,2)</math> giá trị trả lại là biểu thức có <b>N tham gia</b>. Vậy trong hàm <math>f()</math> được phép truy cập giá trị của biến <b>N</b>.</p> <p>Câu 4: Bên trong hàm có thể truy cập để sử dụng <b>giá trị</b> của biến đã khai báo trước đó ở bên ngoài hàm.</p>
Bài 3	Bài 4
<pre>(1) &gt;&gt;&gt; def f(n): (2)         t = 2*n + 1 (3)         return t (4) &gt;&gt;&gt; t = 10 (5) &gt;&gt;&gt; f(1) (6) 3 (7) &gt;&gt;&gt; t (8) 10</pre>	<pre>(1) &gt;&gt;&gt; def f(n): (2)         global t (3)         t = 2*n + 1 (4)         return t (5) &gt;&gt;&gt; t = 10 (6) &gt;&gt;&gt; f(1) (7) 3 (8) &gt;&gt;&gt; t (9) 3</pre>
<p>Câu 1: Tên biến cục bộ <b>t</b></p> <p>Câu 2: Tên biến toàn cục <b>t</b></p>	<p>Câu 1: Tên biến cục bộ <b>t</b></p> <p>Câu 2: Tên biến toàn cục <b>t</b></p> <p>Câu 3: Sự khác nhau của 2 chương trình trên?  <b>Có khai báo từ khóa global. Giá trị của biến t thay đổi với <math>t = 10</math> kết quả trả về <math>t = 3</math></b></p> <p>Câu 4: Biến đã khai báo bên ngoài sẽ không có tác dụng bên trong hàm như một biến.          Nếu muốn có tác dụng thì cần khai báo lại biến này trong hàm với từ khóa <b>global</b>.</p>