

ĐỀ CƯƠNG ÔN TẬP CUỐI HỌC KỲ 2 TIN HỌC 10

Chủ đề F. Giải quyết vấn đề với sự trợ giúp của máy tính

**/ Yêu cầu cần đạt:*

- Viết và thực hiện được một vài chương trình có sử dụng: hằng, biến, cấu trúc điều khiển, các toán tử, các kiểu dữ liệu chuẩn và danh sách, các câu lệnh vào – ra. Qua đó phát triển được năng lực giải quyết vấn đề và sáng tạo, rèn luyện được phẩm chất chăm chỉ, kiên trì và cẩn thận trong học và tự học.
- Viết được chương trình có sử dụng chương trình con trong thư viện chuẩn.
- Viết được chương trình con biểu diễn một thuật toán đơn giản và viết được chương trình có sử dụng chương trình con
- Đọc hiểu được chương trình
- Kiểm thử và gỡ lỗi được chương trình
- Viết và thực hiện được chương trình giải quyết bài toán đơn giản có vận dụng kiến thức liên môn.

I. TÓM TẮT KIẾN THỨC:

1/ Hàm trong python

a. Một số hàm thiết kế sẵn của Python

Python cung cấp sẵn nhiều hàm thực hiện công việc khác nhau cho phép người dùng sử dụng khi viết chương trình với các lệnh gọi hàm tương ứng.

abs()	len()	range()	bool()	float()
list()	round()	chr()	input()	ord()
str()	divmod()	int()	print()	type()

Bảng 26.1: Một số lệnh trong Python

- + Lệnh print("") thực hiện việc in chuỗi ký tự trong dấu ngoặc ra màn hình.
- + Lệnh type() trả lại kiểu dữ liệu của biến.
- + Lệnh int() chuyển chuỗi thành số nguyên.

...

Lưu ý: Chuỗi ký tự bên trong ngoặc của hàm int() và print() là tham số của hàm. Cú pháp có dạng chung như sau: <tên hàm>(<danh sách tham số hàm>)

b. Thiết lập các hàm tự định nghĩa

- Hàm trong Python định nghĩa bằng từ khóa **def**, theo sau là tên hàm (tên hàm sẽ theo quy tắc đặt tên định danh).
- Hàm có thể có hoặc không có tham số.
- Khối lệnh mô tả hàm được viết sau dấu ":" và viết lùi vào, thẳng hàng.
- Hàm có thể có hoặc không có giá trị trả lại sau từ khóa return.
- + Cú pháp thiết lập hàm có trả lại giá trị.

def <tên hàm> (<tham số>):

<khối lệnh>

return <giá trị>

- + Cú pháp thiết lập hàm không trả lại giá trị.

def <tên hàm> (<tham số>):

<khối lệnh>

return

2. Tham số và đối số của hàm

- *Tham số* của hàm được định nghĩa khi khai báo hàm và được dùng như biến trong định nghĩa hàm.
- *Đối số* là giá trị được truyền vào khi gọi hàm.
- Khi gọi hàm, các **tham số (parameter)** sẽ được truyền bằng giá trị thông qua **đối số (argument)** của hàm, số lượng giá trị được truyền vào hàm bằng với số tham số trong khai báo của hàm.

Ví dụ. Cách truyền dữ liệu qua tham số

```
def f(a,b,c):           # Hàm f() có 3 tham số a, b, c
    return a+b+c
```

```
print( f(1,2,3) )
```

Hàm f() được gọi với ba giá trị cụ thể: 1, 2, 3. Kết quả 6

hoặc:

```
x,y,z = 10,20,5
```

```
print( f(x,y,z) )
```

Hàm f() được gọi với ba biến đã có giá trị. Kết quả: 35

3. Cách sử dụng chương trình con (lợi ích khi sử dụng hàm)

- Một hàm có thể sử dụng nhiều lần trong chương trình  Sử dụng CTC (hàm) sẽ giảm số câu lệnh trong chương trình.

- Sử dụng chương trình con có thể giúp phân chia việc giải một bài toán lớn thành giải quyết các bài toán nhỏ và phát huy được tinh thần làm việc nhóm;

- Chương trình chính có cấu trúc rõ ràng, dễ hiểu hơn; Nếu cần hiệu chỉnh, phát triển và nâng cấp cũng thuận tiện hơn.

4. Phạm vi của biến khai báo trong hàm và ngoài hàm:

- *Phạm vi của biến khai báo trong hàm*

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.

- *Phạm vi của biến khai báo ngoài hàm*

Biến đã khai báo bên ngoài hàm (biến tổng thể) chỉ có thể truy cập giá trị để sử dụng bên trong hàm mà không làm thay đổi được giá trị của biến đó (trừ trường hợp với từ khóa **global**)

5. Nhận biết lỗi chương trình

a) Tổng thể có thể phân biệt lỗi chương trình Python làm ba loại:

1. Lỗi khi có lệnh viết sai cú pháp hoặc sai cấu trúc ngôn ngữ Python quy định. Chương trình sẽ lập tức dừng và thông báo lỗi Syntax Error
2. Lỗi khi không thể thực hiện một lệnh trong chương trình. Chương trình dừng lại và thông báo một mã lỗi. Lỗi này gọi là lỗi ngoại lệ (Exceptions Error), mã lỗi trả lại gọi là mã lỗi ngoại lệ.
3. Chương trình chạy không lỗi ngoại lệ, nhưng kết quả đưa ra sai không chính xác. Đây là lỗi logic bên trong chương trình.

+ Với mỗi loại lỗi trên, cách xử lý và kiểm soát lỗi sẽ khác nhau.

b) Một số lỗi ngoại lệ thường gặp

Chúng ta đã biết, nếu gặp lỗi ngoại lệ, chương trình Python sẽ dừng lại, báo lỗi. Một trong những vấn đề được đưa ra khi kiểm soát lỗi là làm thế nào để vẫn phát hiện lỗi, xử lý lỗi nhưng chương trình không bị dừng lại trong khi thực hiện.

Mã lỗi ngoại lệ	Mô tả lỗi
ZeroDivisionError	Lỗi này xảy ra khi lệnh thực hiện phép chia cho giá trị 0.
IndexError	Lỗi xảy ra khi lệnh cố gắng truy cập phần tử của danh sách nhưng chỉ số vượt quá giới hạn.
NameError	Lỗi xảy ra khi chương trình muốn tìm một tên nhưng không thấy. Ví dụ khi lệnh gọi một hàm nhưng không có hàm đó.
TypeError	Lỗi kiểu dữ liệu. Một số ví dụ lỗi loại này: - Lệnh truy cập một phần tử của danh sách nhưng chỉ số không là số nguyên - Lệnh tính biểu thức số nhưng lại có một toán hạng không phải là số
ValueError	Lỗi liên quan đến giá trị của đối tượng. Lỗi khi thực hiện lệnh chuyển đổi kiểu dữ liệu, đối số của hàm có giá trị mà hàm không hỗ trợ. Ví dụ khi thực hiện lệnh <code>int("1.55")</code> sẽ sinh lỗi loại này.
IndentationError	Lỗi khi các dòng lệnh thụt vào không thẳng hàng hoặc không đúng vị trí
SyntaxError	Lỗi cú pháp.

6. Kiểm thử và gỡ lỗi chương trình

Có rất nhiều phương pháp và công cụ khác nhau để kiểm thử chương trình. Các công cụ này không những có mục đích *tìm ra lỗi* (hay bug) của chương trình mà còn có tác dụng *phòng ngừa* và *ngăn chặn* các lỗi phát sinh tiếp trong tương lai.

a) Quan sát mã lỗi Runtime và bắt lỗi ngoại lệ

Nếu chương trình có lỗi Runtime (tức là đang chạy bị dừng lại), cần quan sát các mã lỗi (mã lỗi ngoại lệ) để kiểm tra vị trí dòng lệnh sinh ra lỗi này. Từ đó phân tích, tìm và sửa lỗi.

b) Kiểm thử chương trình với các bộ dữ liệu test

Cần chú ý một số điểm sau:

- Cần có nhiều bộ test (theo các tiêu chí khác nhau như độ lớn, tính đa dạng của dữ liệu.....)
- Cần có bộ test ngẫu nhiên. Việc sinh ngẫu nhiên dữ liệu đầu vào trong miền xác định của chương trình làm tăng khả năng tìm lỗi nếu có.
- Cần có bộ test dữ liệu ở vùng biên. Ví dụ dữ liệu đầu vào là cặp (x, y) xác định trên miền $0 \leq x, y \leq 1$. Khi đó cần kiểm tra chương trình với bộ dữ liệu biên là $(0; 0)$, $(0, 1)$, $(1; 0)$ và $(1; 1)$. Thực tế cho thấy thường phát sinh lỗi tại các vùng biên hoặc lân cận của biên. Một ví dụ khác của dữ liệu biên là cần tìm các bộ test với n và các giá trị $(x_1, x_2, \dots, x_n)$ rất lớn (vùng cận biên lớn)

c) In các thông số trung gian

Bổ sung vào giữa các dòng lệnh print() để in ra các biến trung gian, qua đó kiểm tra các quy trình hay thuật toán được viết có đúng không.

Giả sử chương trình có đầu vào là $(x_1, x_2, \dots, x_n)$, đầu ra là $(a_1, a_2, \dots, a_m)$ nhưng có sử dụng các biến trung gian $(y_1, y_2, \dots, y_k)$. Khi đó với mỗi bộ test đầu vào, chúng ta sẽ bổ sung vào các dòng lệnh của chương trình để in ra các giá trị trung gian:

$(x_1, x_2, \dots, x_n)$, $(y_1, y_2, \dots, y_k)$, $(a_1, a_2, \dots, a_m)$

Thông qua các giá trị trung gian trong quá trình thực hiện chương trình, nếu kết quả cuối cùng có lỗi thì sẽ dễ tìm ra lỗi đó.

d) Sử dụng công cụ break point (điểm dừng)

Công cụ break point cho phép tạo ra các “điểm dừng” bên trong chương trình. Khi chạy, chương trình sẽ tạm dừng lại tại các “điểm dừng” cho phép người kiểm thử có thể quan sát các thông tin khác bên trong chương trình, qua đó kiểm tra tính đúng đắn của chương trình.

Trên thực tế sử dụng phương pháp điểm dừng thường kết hợp với phương pháp in các giá trị trung gian sẽ là hiệu quả hơn để kiểm thử chương trình.

II. CÂU HỎI TRẮC NGHIỆM.

1. DẠNG 1: Học sinh trả lời các câu hỏi sau. Mỗi câu hỏi HS chỉ chọn một phương án.

Câu 1. (B26, 10F, B, CC4). Khối lệnh mô tả hàm được viết sau dấu “:” và viết

- A. thẳng hàng với từ khoá khai báo hàm
- B. lùi vào, thẳng hàng
- C. lùi vào và sau đó thì viết tự do không cần thẳng hàng
- D. tự do- không cần lùi vào hay thẳng hàng

Câu 2. (B26, 10F, B, CC4). Hãy chọn phát biểu SAI trong các phát biểu sau?

- A. Lệnh input() thực hiện yêu cầu nhập vào một biểu thức, số hay một xâu bất kì.
- B. Lệnh print() thực hiện việc in ra màn hình
- C. Lệnh type() trả lại kiểu dữ liệu của biểu thức trong ngoặc
- D. Lệnh str() chuyển đổi tương đương cho thành chuỗi

Câu 3. (B26, 10F, B, CC4). Từ khóa nào dùng để khai báo hàm người dùng tự định nghĩa?

- A. return
- B. dec
- C. def
- D. print

Câu 4. (B26, 10F, B, CC4). Hãy chọn phát biểu đúng trong các phát biểu sau?

- A. Lệnh bool() chuyển một giá trị sang True hoặc False
- B. Có ít hàm có sẵn được xây dựng trong Python
- C. Lệnh input() có thể nhập vào một số nguyên mà không cần chuyển đổi kiểu

D. Lệnh float() trả về số nguyên từ số hoặc chuỗi biểu thức

Câu 5. (B26, 10F, B, CC4). Mệnh đề nào dưới đây mô tả đúng về hàm tự định nghĩa không trả lại giá trị?

A. Trong mô tả hàm không có từ khóa return

B. Trong mô tả hàm hoặc không có return hoặc có return nhưng không có giá trị sau từ khóa return

C. Trong mô tả hàm chỉ có một từ khóa return

D. Trong mô tả hàm phải có tối thiểu hai từ khóa return

Câu 6. (B27, 10F, B, CC4). Khẳng định nào dưới đây KHÔNG đúng?

A. Hàm có thể có hoặc không có tham số

B. Khi truyền đối số (giá trị) đưa vào bằng số tham số trong hàm

C. Có thể truyền giá trị cho tham số

D. Hàm bắt buộc có tham số

Câu 7. (B27, 10F, B, CC4). Ý kiến nào đúng khi nói về hàm?

A. Hàm luôn trả về 1 giá trị

B. Hàm có thể trả về giá trị hoặc không trả về giá trị.

C. Kết quả trả về của hàm có thể gán cho một biến mới

D. chỉ có duy nhất một hàm trong chương trình

Câu 8. (B27, 10F, B, CC4). Đặc điểm nào KHÔNG thể hiện vai trò của chương trình con?

A. Có thể gọi sử dụng mọi thời điểm

B. Chương trình dài dòng, khó chỉnh sửa

C. Chương trình ngắn gọn, dễ hiệu chỉnh nâng cấp

D. Giúp giải các bài toán lớn hiệu quả

Câu 9. (B27, 10F, B, CC4). Trong Python, hàm tự định nghĩa có thể có bao nhiêu tham số?

A. 1

B. 0

C. không hạn chế

D. 2

Câu 10. (B27, 10F, B, CC4). Khi gọi hàm, dữ liệu được truyền vào hàm được gọi là gì?

A. tham số

B. hàm số

C. đối số

D. hiệu số

Câu 11. (B28, 10F, B, CC1) Phát biểu nào dưới đây về biến địa phương và biến tổng thể là SAI?

A. Biến tổng thể có thể được sử dụng ở trong hàm

B. Biến địa phương phải có tên khác với tên của biến tổng thể

C. Biến địa phương có thể có kiểu khác với kiểu của biến tổng thể có cùng tên

D. Biến địa phương được khai báo trong hàm

Câu 12. (B28, 10F, B, CC1) Trong Python, biến được khai báo và sử dụng bên trong một hàm được gọi là gì?

A. Biến riêng

B. Biến tổng thể

C. Biến thông thường

D. Biến địa phương

Câu 13. (B28, 10F, B, CC1) Nếu muốn biến bên ngoài hàm vẫn có tác dụng bên trong hàm thì cần khai báo lại biến này bên trong hàm với từ khóa

A. gloabl

B. float

C. global

D. int

Câu 14. (B28, 10F, B, CC1) Phạm vi hoạt động của biến địa phương?

- A. Trong chương trình chính
- B. Trong chương trình con chứa biến đó
- C. Chỉ một số chương trình con được sử dụng
- D. Trong chương trình chính và tất cả chương trình con

Câu 15. (B29, 10F, B, CC1) Lỗi NameError có nghĩa là gì?

- A. Lỗi xảy ra khi lệnh cố gắng truy cập phần tử của danh sách nhưng chỉ số vượt qua giới hạn
- B. Lỗi kiểu dữ liệu
- C. Lỗi cú pháp
- D. Lỗi xảy ra khi chương trình muốn tìm một tên nhưng không thấy

Câu 16. (B29, 10F, B, CC1) Lỗi nào khó phát hiện nhất?

- A. Lỗi ngoại lệ
- B. Lỗi cú pháp và ngoại lệ
- C. Lỗi ngữ nghĩa.
- D. Lỗi cú pháp

Câu 17. (B29, 10F, B, CC1) Tổng thể có thể phân biệt lỗi chương trình Python làm mấy loại?

- A. 4
- B. 3
- C. 7
- D. 5

Câu 18. (B29, 10F, B, CC1) Phát biểu nào sau đây là đúng về lỗi ngữ nghĩa (lỗi logic)?

- A. Là lỗi mặc dù các câu lệnh viết đúng quy định của ngôn ngữ nhưng sai trong thao tác xử lý nào đó.
- B. Là lỗi xảy ra khi chương trình muốn tìm một tên nhưng không thấy
- C. Là lỗi câu lệnh viết không theo đúng quy định của ngôn ngữ
- D. Là lỗi xảy ra khi chương trình đang chạy, một lệnh nào đó không thể thực hiện

Câu 19. (B29, 10F, B, CC1) Phát biểu nào sau đây là đúng về lỗi ngoại lệ?

- A. Là lỗi mặc dù các câu lệnh viết đúng quy định của ngôn ngữ nhưng sai trong thao tác xử lý nào đó
- B. Là lỗi xảy ra khi chương trình muốn tìm một tên nhưng không thấy
- C. Là lỗi xảy ra khi chương trình đang chạy, một lệnh nào đó không thể thực hiện
- D. Là lỗi câu lệnh viết không theo đúng quy định của ngôn ngữ

Câu 20. (B29, 10F, B, CC1) Phát biểu nào sau đây là đúng về lỗi cú pháp?

- A. Là lỗi mặc dù các câu lệnh viết đúng quy định của ngôn ngữ nhưng sai trong thao tác xử lý nào đó
- B. Là lỗi xảy ra khi chương trình muốn tìm một tên nhưng không thấy
- C. Là lỗi câu lệnh viết không theo đúng quy định của ngôn ngữ.
- D. Là lỗi xảy ra khi chương trình đang chạy, một lệnh nào đó không thể thực hiện

Câu 21. (B29, 10F, B, CC1) Lỗi xảy ra khi lệnh cố gắng truy cập phần tử của danh sách nhưng chỉ số vượt quá giới hạn, có mã lỗi là:

- A. IndentationError
- B. TypeError
- C. IndexError
- D. ValueError

Câu 22. (B29, 10F, B, CC1) Khẳng định nào về ngoại lệ là đúng nhất?

- A. Ngoại lệ là lỗi phát sinh khi đang thực thi chương trình
- B. Ngoại lệ được Python dùng để loại bỏ một khối code khỏi chương trình
- C. Lỗi cú pháp cũng là một ngoại lệ
- D. Ngoại lệ là lỗi mặc dù các câu lệnh viết đúng quy định của ngôn ngữ nhưng sai trong thao tác xử lý nào đó

Câu 23. (B30, 10F, B, CC4). Chương trình chạy phát sinh lỗi ngoại lệ ZeroDivision, nên xử lý như thế nào?

- A. Kiểm tra lại kiểu dữ liệu nhập vào
- B. Kiểm tra lại chỉ số trong mảng
- C. Kiểm tra lại giá trị số chia
- D. Kiểm tra lại giá trị của số bị chia

Câu 24. (B30, 10F, B, CC4). Hoàn thành phát biểu sau: “Có rất nhiều công cụ và phương pháp

khác nhau để kiểm thử chương trình. Các công cụ có mục đích ...(1) ... của chương trình và ... (2)....

...(3)... các lỗi phát sinh trong tương lai.”

- A. xử lý lỗi, phòng ngừa, ngăn chặn
- B. tìm ra lỗi, phòng ngừa, ngăn chặn
- C. phòng ngừa, ngăn chặn, xử lý lỗi
- D. tìm ra lỗi, phòng ngừa, xử lý

Câu 25. (B30, 10F, B, CC4). Chương trình chạy phát sinh lỗi ngoại lệ `NameError`, nên xử lý như thế nào?

- A. Kiểm tra kiểu dữ liệu nhập vào
- B. Kiểm tra lại tên các biến và hàm
- C. Kiểm tra giá trị của số bị chia
- D. Kiểm tra lại chỉ số trong mảng

Câu 26. (B30, 10F, B, CC4). Phần mềm soạn thảo lập trình có thể tạo bao nhiêu điểm dừng?

- A. 0
- B. 1
- C. 10
- D. Không hạn chế

Câu 27. (B30, 10F, B, CC4).. Chương trình chạy phát sinh lỗi ngoại lệ `IndexError`, nên xử lý như thế nào?

- A. Kiểm tra lại kiểu dữ liệu nhập vào
- B. Kiểm tra lại chỉ số trong danh sách
- C. Kiểm tra lại giá trị của số bị chia
- D. Kiểm tra lại giá trị số chia

Câu 28. (B30, 10F, B, CC4). Đâu KHÔNG phải là công cụ để kiểm thử chương trình?

- A. Công cụ sinh các bộ dữ liệu test
- B. Công cụ in biến trung gian
- C. Công cụ thống kê dữ liệu
- D. Công cụ điểm dừng trong phần mềm soạn thảo lập trình

Câu 29. (B30, 10F, B, CC4).. Bộ dữ liệu kiểm thử (test) có tính chất gì?

- A. Cần có càng nhiều càng tốt
- B. Không cần có tính chất gì.
- C. Cần được sinh ngẫu nhiên và phủ kín các trường hợp biên của dữ liệu đầu vào bài toán.
- D. Tính chất phát hiện lỗi của chương trình

Câu 30. (B30, 10F, B, CC4). Điểm dừng (break point) trong các phần mềm soạn thảo lập trình có ý nghĩa gì?

- A. Đó là vị trí chương trình tạm dừng, người lập trình sẽ quan sát các biến của chương trình và có thể điều khiển để chương trình tiếp tục chạy.
- B. Đó là vị trí chương trình chạy tới đó thì kết thúc
- C. Đó là vị trí chương trình mỗi khi chạy đến dòng lệnh đó sẽ kêu pip pip
- D. Đó là vị trí chương trình dừng lại để người lập trình quan sát phát hiện lỗi

Câu 31. (B30, 10F, B, CC4). Việc tạo ra các “điểm dừng” bên trong chương trình được tạo ra bởi công cụ nào?

- A. Các bộ test.
- B. Công cụ sinh các bộ dữ liệu test
- C. Công cụ break point.
- D. Công cụ in.

2. DẠNG 2: HS trả lời câu hỏi. Trong mỗi ý ở mỗi câu, học sinh chọn Đúng hoặc Sai.

Câu 1: Muốn chương trình Python có kết quả chính xác thì chúng ta phải kiểm thử chương trình. Vậy các phát biểu sau của An sau là Đúng hay Sai?

a) (B30, 10F, B, CC4) Kiểm thử chương trình với các bộ dữ liệu test là một phương pháp được thử

với một số bộ dữ liệu test gồm đầu vào tiêu biểu phụ thuộc vào đặc thù của bài toán.

b) (B30, 10F, B, CC4) Có nhiều phương pháp và công cụ khác nhau để kiểm thử chương trình. Các công cụ này không những có mục đích tìm ra lỗi của chương trình, mà còn giúp sửa lỗi.

c) B30, 10F, H, CC4) Chương trình tìm ước chung lớn nhất của 2 số m,n như sau: `m=int(input("Nhập m=''))`

`n=int(input("Nhập n='')) while m!=n:`

`if m>n: m=m-n else: n=n-m`

`print(m)`

Khi nhập `m=4, n=2` thì kết quả chương trình là 4

d) (B30, 10F, H, CC4) Chương trình tìm ước chung lớn nhất của 2 số m,n như sau:

`m=int(input("Nhập m=''))`

`n=int(input("Nhập n='')) while m!=n:`

`if m>n: m=m-n else: n=n-m`

`print(m)`

Với chương trình trên, ta dùng phương pháp thử với bộ dữ liệu test là nhập m, n với 2 trường hợp là `m>n` và `m<n`

Câu 2: Trong quá trình viết và chạy chương trình Python, học sinh có thể gặp các lỗi như lỗi cú pháp, lỗi khi chạy và lỗi logic. Việc nhận biết và xử lý các lỗi này giúp chương trình hoạt động đúng. Các phát biểu sau là Đúng hay Sai?

a) (B29, 10F, H, CC4).Chương trình sau báo lỗi `IndexError A=[1,4,5]`

`for i in A:`

`print(i)`

b) (B29, 10F, B, CC4). Chương trình chạy không có lỗi nhưng kết quả đưa ra sai là lỗi ngoại lệ.

c) (B29, 10F, H, CC4). Chương trình sau báo lỗi `TypeError print("10"*3.5)`

d) (B29, 10F, B, CC4). Lỗi cú pháp (Syntax Error) xảy ra khi viết sai cú pháp hoặc sai cấu trúc ngôn ngữ Python quy định.

Câu 3: Hàm trong Python là một khối lệnh được định nghĩa để thực hiện một chức năng cụ thể, giúp cho việc lập trình trở nên rõ ràng, dễ hiểu và dễ bảo trì hơn. Khi cần thực hiện một công việc nào đó nhiều lần, ta chỉ cần gọi lại hàm thay vì viết lại toàn bộ lệnh. Hàm có thể trả về kết quả thông qua từ khóa `return`. Python cũng hỗ trợ hàm có sẵn (built-in functions) và cho phép lập trình viên tự định nghĩa hàm theo nhu cầu sử dụng.

Các phát biểu sau về hàm trong Python Đúng hay Sai?

a) (B27, 10F, B, CC4). Hàm tự định nghĩa được khai báo bằng từ khóa **def**, theo sau là tên hàm và dấu ngoặc chứa các tham số (nếu có).

b) B27, 10F, B, CC4). Tham số của hàm là giá trị được truyền vào khi gọi hàm.

c) B27, 10F, H, CC4). Kết quả của chương trình sau là 8 `def Vd(a,b):`

`k=a+b**2 return k`

`print(Vd(2,3))`

d) B27, 10F, H, CC4). Một hàm khi khai báo có 1 tham số, khi gọi hàm có thể có 2 đối số.

III. THỰC HÀNH

1/ Viết hàm với đầu vào là chuỗi ký tự và số c, đầu ra là danh sách các từ được tách ra từ chuỗi 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 B là các từ được chuyển thành chữ in hoa.

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

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

2/ Nhập từ bàn phím một dãy các số nguyên, mỗi số cách nhau bởi dấu cách. Chuyển các số trong dãy này thành danh sách A các số và in danh sách A ra màn hình.

3/ Nhập từ bàn phím một dãy các số thực, mỗi số cách nhau bởi dấu phẩy và một số k, hãy in danh sách mới ra màn hình gồm các số lớn hơn hoặc bằng k.

4/ 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 ra danh sách B chỉ gồm các số chẵn có trong A.

5/ Nhập từ bàn phím một dãy các số nguyên, mỗi số cách nhau bởi dấu cách. Hãy in danh sách mới ra màn hình gồm các số chia hết cho 2 và 3.

6/ Nhập từ bàn phím một dãy các số nguyên, mỗi số cách nhau bởi dấu cách. Hãy in danh sách mới ra màn hình gồm các phần tử lớn hơn 0.

7/ Viết hàm với đầu vào, đầu ra như sau:

- Đầu vào các phần tử là xâu kí tự.

- Đầu ra là danh sách gồm các phần tử là kí tự đầu tiên của các xâu kí tự tương ứng trong danh sách tách ra từ xâu ban đầu

8/ Nhập từ bàn phím một dãy các số nguyên, mỗi số cách nhau bởi dấu phẩy. Hãy in danh sách mới ra màn hình gồm các phần tử có chỉ số chẵn, các phần tử có chỉ số lẻ.

Chú ý: Tất cả chương trình đều có sử dụng hàm.

9/ Câu hỏi và bài tập SGK Tin 10 từ bài 26 đến bài 30.

ĐÁP ÁN THAM KHẢO

DẠNG 1: Học sinh trả lời các câu hỏi sau.

Câu	Đáp án	Câu	Đáp án	Câu	Đáp án	Câu	Đáp án
1	B	9	C	17	B	25	B
2	A	10	C	18	A	26	D
3	C	11	B	19	C	27	B
4	A	12	D	20	C	28	C
5	B	13	C	21	C	29	D
6	D	14	B	22	A	30	A
7	B	15	D	23	C	31	C
8	B	16	C	24	B		

2. DẠNG 2: HS trả lời câu hỏi.

Câu	Ý a	Ý b	Ý c	Ý d
1	Đúng	Đúng	Sai	Đúng
2	Sai	Sai	Đúng	Đúng
3	Đúng	Sai	Sai	Sai