Sở GD&ĐT Thành phố Đà Nẵng
Trường THPT Đỗ Đăng Tuyển
Giáo viên soạn: Lâm Thị Hương
Ngày soạn: 30/03/2026
Lớp dạy: 10/1, 10/3, 10/7
Thời gian thực hiện: Tuần 29

CHỦ ĐỀ 5: GIẢI QUYẾT VẤN ĐỀ VỚI SỰ TRỢ GIÚP CỦA MÁY TÍNH
BÀI 29: NHẬN BIẾT LỖI CHƯƠNG TRÌNH

Môn học: Tin học lớp 10. Thời gian thực hiện: 2 tiết

I. MỤC TIÊU

1. Về kiến thức

- Biết và phân loại được một số loại lỗi chương trình
- Biết được một vài lỗi ngoại lệ thường gặp

2. Về năng lực

2.1 Năng lực chung

- Năng lực tự chủ, tự học: chủ động và tích cực thực hiện nhiệm vụ học tập; vận dụng được những kiến thức, kĩ năng đã học để hoàn thành nhiệm vụ.
- Năng lực giao tiếp và hợp tác: hiểu được mục đích giao tiếp và giao tiếp hiệu quả trong hoạt động nhóm.
- Năng lực giải quyết vấn đề và sáng tạo: Phân tích được tình huống, đề xuất và lựa chọn giải pháp để giải quyết các vấn đề do giáo viên đặt ra.

2.2 Năng lực tin học

- NLc (Giải quyết vấn đề với sự hỗ trợ của công nghệ thông tin và truyền thông): HS được rèn luyện, bồi dưỡng năng lực giải quyết vấn đề thông qua việc sử ngôn ngữ lập trình Python để giải quyết các bài toán phát hiện và sửa được các lỗi gặp phải khi lập trình, đồng thời phát triển được năng lực tư duy lập trình.
- NLd (Ứng dụng công nghệ thông tin và truyền thông trong học và tự học): HS Khai thác được các dịch vụ tra cứu và trao đổi thông tin, các nguồn học liệu mở để cập nhật kiến thức về các lỗi chương trình khi dùng NNLT Python, hỗ trợ học tập và tự học.
- NLe (Hợp tác trong môi trường số): Có khả năng làm việc nhóm, hợp tác được trong việc tạo ra, trình bày giới thiệu được sản phẩm.

2.3 Phát triển các năng lực số cốt lõi

3.4.NC1a: HS tự thao tác được với các chỉ dẫn của hệ thống máy tính qua việc biết và phân loại được một số loại lỗi chương trình để giải quyết vấn đề khi lập trình.

3. Về phẩm chất

- Hình thành ý thức trách nhiệm, tính cẩn thận khi làm việc nhóm.
- Rèn luyện phẩm chất làm việc chăm chỉ, chuyên cần khi thực hiện nhiệm vụ.

II. THIẾT BỊ DẠY HỌC VÀ HỌC LIỆU

1. Đối với GV:

- Máy chiếu, máy tính đã cài đặt sẵn Python, các chương trình mẫu để chạy minh họa kết quả cho học sinh quan sát.
- Sách giáo khoa, Sách giáo viên, kế hoạch bài dạy, bài giảng điện tử.

2. Đối với HS:

- Sách giáo khoa, vở ghi
- Điện thoại thông minh, máy tính

III. TIẾN TRÌNH DẠY HỌC

A. HOẠT ĐỘNG MỞ ĐẦU (Thời gian 10 phút)

a. Mục tiêu:

- Học sinh phát hiện được vấn đề là chương trình có lỗi.
- Chỉ ra được những vị trí có lỗi đơn giản

b. Nội dung:

- Phát hiện lỗi trong chương trình giáo viên cho.
- Sửa lỗi để chương trình có thể chạy được.

c. Sản phẩm:

- Học sinh chỉ ra được các lỗi: sau if và else thiếu dấu (:), thiếu dấu đóng ngoặc trong câu lệnh *print("dien tich tam giac: ", math.sqrt(p*(p-a)*(p-b)*(p-c)*

- Phát hiện lỗi khai báo biến t

d. Tổ chức hoạt động

Bước 1. *Giao nhiệm vụ học tập*

- Giáo viên chiếu chương trình đã chuẩn bị sẵn lên ti vi. Yêu cầu các nhóm quan sát và thảo luận tìm ra những lỗi trong chương trình.

```
a=float(input("a= "))
b=float(input("b= "))
c=float(input("c= "))
t == (a+b)>c and (b+c)>a and (a+c)>b
if t
   print("chu vi tam giac: ", a+b+c)
   p= (a+b+c)/2
   import math

   print("dien tich tam giac: ", math.sqrt(p*(p-a)*(p-b)*(p-c)
else
   print(" khong phai tam giac")
```

Bước 2. *Thực hiện nhiệm vụ*

- Học sinh quan sát và tiến hành thảo luận.

Bước 3. *Báo cáo, thảo luận*

- Học sinh thảo luận và tìm được các lỗi sau if t và else thiếu dấu (:), thiếu dấu đóng ngoặc trong câu lệnh *print("dien tich tam giac: ", math.sqrt(p*(p-a)*(p-b)*(p-c)*

Bước 4. *Kết luận, nhận định*

- Giáo viên cho 1 học sinh lên sửa trực tiếp trên máy tính những lỗi các bạn đã phát hiện. Sau đó chạy chương trình. Học sinh sẽ thấy chương trình chạy được nhưng vẫn chưa cho kết quả được, và chương trình báo lỗi *NameError: name 't' is not defined.*

- Giáo viên sửa lỗi xong chạy cho học sinh xem.

- Nhận định: Trong quá trình lập trình chúng ta sẽ không tránh khỏi những sai sót. Để có thể xử lý những vấn đề phát sinh trong lập trình chúng ta cần biết và phân biệt được các loại lỗi trong Python, Với mỗi loại lỗi chúng ta sẽ có cách xử lí và kiểm soát khác nhau. Và để rõ hơn về vấn đề này hôm nay chúng ta sẽ cùng tìm hiểu bài 29: NHẬN BIẾT LỖI CHƯƠNG TRÌNH

B. HOẠT ĐỘNG HÌNH THÀNH KIẾN THỨC (Thời gian 50')

Hoạt động 1. Nhận biết và phân biệt một số loại lỗi chương trình (Thời gian 15')

a. Mục tiêu

- Học sinh nhận biết rằng khi lập trình có thể gặp phải nhiều loại lỗi khác nhau.

- Học sinh có thể phân biệt được các loại lỗi đó.

b. Nội dung

- Quan sát các ví dụ giáo viên thực hiện trực tiếp trên máy tính để nhận biết các loại lỗi gặp phải.

c. Sản phẩm:

- Học sinh có thể phân biệt được ba loại lỗi:
  + Lỗi cú pháp
  + Lỗi ngoại lệ.
  + Lỗi lôgic bên trong chương trình

d. Tổ chức hoạt động

HOẠT ĐỘNG CỦA GV – HS	DỰ KIẾN SẢN PHẨM
Bước 1. *Giao nhiệm vụ học tập* Giáo viên chia lớp thành 6 nhóm, yêu cầu học sinh thực hiện các nhiệm vụ sau: NV1: Dùng máy tính hoặc điện	

thoại thông minh thực hiện các ví dụ sau. Cho biết hiện tượng xảy ra đối với mỗi ví dụ. VD1: *>>> if 2>3 Print(2+3)* VD2: *>>> n = int(input("nhap n:"))* *Nhập n: 1.5* VD3 a=[1,3,10,0] for i in range(5): print(a[i],end = " ") VD4: Tính tổng 1+2+3 >>>s =0 >>>for i in range (3): s = s+i >>>print(s) NV2: Phân biệt các lỗi trong các ví dụ trên. Bước 2. *Thực hiện nhiệm vụ* HS tiếp nhận nhiệm vụ, hoạt động nhóm, thảo luận, hoàn thành các NV được giao - Nghiên cứu ví dụ 1, 2, 3,4 theo nhóm. Sử dụng điện thoại thông minh hoặc máy tính để thao tác các lệnh trong ví dụ và thu thập kết quả - Trả lời nhiệm vụ 2. Ghi lại đáp án thống nhất ra bảng phụ. GV quan sát học sinh thảo luận,	Sản phẩm 1 VD1: lỗi SyntaxError: invalid syntax VD2: lỗi Traceback (most recent call last): File "<pyshell#55>", line 1, in <module> n = int(input("nhap n:")) ValueError: invalid literal for int() with base 10: '1.5' VD3: lỗi 1 3 10 0 Traceback (most recent call last): File "C:/Users/Admin/Desktop/HINHTRON.py", line 3, in <module> print(a[i],end = " ") IndexError: list index out of range VD4 3 Sản phẩm 2 VD1: Lỗi sai cú pháp VD2: Lỗi nhập dữ liệu không đúng khuôn dạng VD3: Lỗi chỉ số vượt quá giới hạn cho phép VD4: Lỗi lôgic bên trong chương trình (cho kết quả sai) ……

tự học, trợ giúp kịp thời khi các em cần hỗ trợ Bước 3. *Báo cáo, thảo luận* GV gọi đại diện 2 nhóm HS hoàn thành nhanh nhất treo sản phẩn lên bảng và trình bày. HS trình bày kết quả thu nhận được GV yêu cầu các nhóm HS khác đưa ra ý kiến nhận xét, đánh giá Bước 4. *Kết luận, nhận định* Có thể phân biệt lỗi chương trình Python thành 3 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 mã lỗi Exceptions Error. Lỗi này gọi là lỗi ngoại lệ. 3. Chương trình chạy không lỗi ngoại lệ, nhưng kết quả đưa ra sai, không chính xác. Đây là lỗi lô gic bên trong chương trình	

Hoạt động 2. Một số lỗi ngoại lệ thường gặp (Thời gian 15')

a. Mục tiêu

- Nhận biết một số lỗi ngoại lệ thường gặp trong chương trình Python
- Xử lí được các lỗi đó.

b. Nội dung

- Quan sát và chạy chương trình mẫu để phát hiện các lỗi ngoại lệ, phát hiện lỗi chương trình gặp phải, thống kê mã lỗi ngoại lệ và mô tả lỗi ấy vào phiếu học tập
- Sau đó các nhóm sẽ sửa các lỗi ấy để được một chương trình hoàn chỉnh.

c. Sản phẩm

- Phiếu học tập của học sinh.
- Chương trình đã sửa lỗi

d. Tổ chức hoạt động

HOẠT ĐỘNG CỦA GV – HS	DỰ KIẾN SẢN PHẨM
Bước 1. *Giao nhiệm vụ học tập* Giáo viên chiếu các lệnh trên bảng, phát phiếu học tập. Yêu cầu học sinh thực hiện các nhiệm vụ sau: NV1: Gõ các lệnh đã cho vào máy tính hoặc điện thoại thông minh, phát hiện các lỗi sinh trong chương trình, thống kê các mã lỗi vào phiếu học tập. Lệnh 1: A[1.5] Lệnh 2: int (“abc”) Lệnh 3: “10”*3.5 Lệnh 4: 12 + x(10) Lệnh 5: n = int(input(“nhap n”)) print(10/n) NV2: Mô tả các lỗi vừa phát hiện theo mẫu trong phiếu học tập. Bước 2. *Thực hiện nhiệm vụ* HS tiếp nhận nhiệm vụ, hoạt động nhóm, thảo luận, hoàn thành các NV được giao - Nghiên cứu chương trình theo nhóm. Sử dụng điện thoại thông minh hoặc máy tính để thao tác các lệnh trong ví dụ và thu thập kết quả - Trả lời nhiệm vụ 2. Ghi lại đáp án thống nhất ra bảng phụ. GV quan sát học sinh thảo luận, tự học, trợ giúp kịp thời khi các em cần hỗ trợ Bước 3. *Báo cáo, thảo luận* GV gọi đại diện 2 nhóm HS hoàn thành nhanh nhất treo sản phẩn lên bảng và trình bày. HS trình bày kết quả thu nhận được GV yêu cầu các nhóm HS khác	Sản phẩm 1 A[1.5]: NameError int(“abc”): ValueError “10”*3.5: TypeError 12 + x(10): NameError n = int(input(“nhap n”)) print(10/n) ZeroDivisionError (nếu nhập số 0) Sản phẩm 2 Phiếu học tập

đưa ra ý kiến nhận xét, đánh giá Bước 4. *Kết luận, nhận định* Giáo viên nhận xét đánh giá về kết quả thảo luận của học sinh. Chuẩn hóa kiến thức và bổ sung thêm một số lỗi ngoại lệ khác	

Hoạt động 3. Lập trình và kiểm tra khả năng sinh lỗi khi chạy chương trình (Thời gian 20')

a. Mục tiêu

- Rèn luyện kỹ năng lập trình cho học sinh
- Biết cách kiểm tra khả năng sinh lỗi khi chạy chương trình

b. Nội dung

Thực hiện và kiểm tra lỗi phát sinh của hai chương trình

- Viết chương trình nhập các số nguyên m, n từ bàn phím, cách nhau bởi dấu cách. Chương trình đưa ra tổng, hiệu, thương của hai số đã nhập.
- Viết chương trình nhập số tự nhiên n và nhập lần lượt n số nguyên vào danh sách số A. Sau khi nhập xong in danh sách số A ra màn hình.

c. Sản phẩm:

- Những lỗi có khả năng sinh của chương trình được học sinh ghi ra trên bảng phụ

d. Tổ chức hoat động

HOẠT ĐỘNG CỦA GV – HS	DỰ KIẾN SẢN PHẨM
Bước 1. *Giao nhiệm vụ học tập* Giáo viên giao nhiệm vụ cho học sinh. Yêu cầu học sinh tìm hiểu thực hiện và kiểm tra khả năng sinh lỗi của chương trình. Viết ra các lỗi có thể sinh của chương trình ra bảng phụ. NV1: Viết chương trình nhập các số nguyên m, n từ bàn phím, cách nhau bởi dấu cách. Chương trình đưa ra tổng, hiệu, thương của hai số đã nhập. NV2: Viết chương trình nhập số tự nhiên n và nhập lần lượt n số nguyên vào danh sách số A. Sau khi nhập xong in danh sách số A ra màn hình. Bước 2. *Thực hiện nhiệm vụ*	Sản phẩm 1: Đối với nhiệm vụ 1 Lỗi 1: TypeError: 'str' object cannot be interpreted as an integer (các số m, n không phải số nguyên) Lỗi 2: IndexError: list index out of range (giữa m, m không có dấu cách) Lỗi 3: ZeroDivisionError: division by zero (lỗi chia cho số 0 lí do nhập n=0) Sản phẩm 2: Đối với nhiệm vụ 2 Lỗi 1: ValueError: invalid literal for int() with base 10: '2.3' (số n không phải là số nguyên. Lỗi 2: ValueError: invalid literal for

HS tiếp nhận nhiệm vụ, hoạt động nhóm, thảo luận, hoàn thành các NV được giao - Nghiên cứu chương trình theo nhóm. Sử dụng điện thoại thông minh hoặc máy tính để thao tác các lệnh trong NV1 và thu thập kết quả - Tiếp tục thực hiện với nhiệm vụ 2. Ghi lại đáp án thống nhất ra bảng phụ. GV quan sát học sinh thảo luận, tự học, trợ giúp kịp thời khi các em cần hỗ trợ Bước 3. *Báo cáo, thảo luận* GV gọi đại diện 2 nhóm HS hoàn thành nhanh nhất treo sản phẩn lên bảng và trình bày. HS trình bày kết quả thu nhận được GV yêu cầu các nhóm HS khác đưa ra ý kiến nhận xét, đánh giá Bước 4. *Kết luận, nhận định* Giáo viên nhận xét đánh giá về kết quả thảo luận của học sinh. Chuẩn hóa kiến thức: Có một số lỗi chúng ta sẽ phát hiện khi dịch chương trình (lỗi cú pháp) nhưng sẽ có những lỗi ta chỉ phát hiện khi chạy chương trình (lỗi ngoại lệ). Để kiểm soát được các lỗi phát phát sinh của chương trình thì bước kiểm thử là một công đoạn rất quan trọng. Cách kiểm thử và gỡ lỗi chương trình chúng ta sẽ tìm hiểu kỹ hơn ở bài 30.	int() with base 10: '1.2' (số hạng của danh sách nhập không phải là số nguyên)

C. HOẠT ĐỘNG LUYỆN TẬP (Thời gian 15')

a . Mục tiêu

- Luyện tập về phát hiện lỗi sinh trong chương trình
- Phân biệt được các loại lỗi

b. Nội dung

- Trả lời các câu hỏi 1 và 2 trang 144 SGK

c. Sản phẩm

- Phiếu học tập của học sinh

d. Tổ chức hoạt động

Bước 1. *Giao nhiệm vụ học tập*

- Yêu cầu học sinh thực hiện các bài tập 1,2 phần luyện tập trang 144 sách giáo khoa

Bước 2. *Thực hiện nhiệm vụ*

- HS thực hiện theo yêu cầu

Bước 3. *Báo cáo, thảo luận*

- Giáo viên kiểm tra sản phẩm của HS.
- Cho 2 học sinh trình bày sản phẩm của mình.

Bước 4. *Kết luận, nhận định*

- Giáo viên nhận xét bài làm của học sinh chuẩn hóa kiến thức

Bài 1: Các lệnh sau có sinh lỗi chương trình hay không? Đó là lỗi gì?

```
>>> A = [1,3,5,7,10,0]
>>> for k in range(1, len(A)+1):
        Print(A[k])
>>> s1, s2 = “101010”, 1010101
>>> s = s1 + s2
```

Lệnh 1: SyntaxError

Lệnh 2: TypeError

Bài 2: Để tính giá trị trung bình của một danh sách số A người lập trình đã dùng lệnh sau để tính:

gttb = sum(A)/len(A)

Lệnh này có thể sinh lỗi ngoại lệ không? Nếu có thì những lỗi gì?

Lỗi có thể phát sinh là lỗi chia cho 0 nếu dãy A rỗng

D. HOẠT ĐỘNG VẬN DỤNG (Thời gian 15’)

a. Mục tiêu

- Củng cố lại kiến thức học sinh đã được học.
- Vận dụng kiến thức để giải bài tập, hiểu và phân biệt được các loại lỗi có thể sinh trong chương trình

b. Nội dung

- Bài tập 1 trang 144

c. Sản phẩm

- Chương trình của HS
- Danh sách các lỗi sinh trong chương trình.

d. Tổ chức hoạt động

Bước 1. *Giao nhiệm vụ học tập*

- GV: yêu cầu học sinh đọc đề và thực hiện

Giả sử em được yêu cầu viết một chương trình nhập số tự nhiên n từ bàn phím, kết quả đưa ra là danh sách các ước số thực sự của n, tính cả 1 và không tính n. Hãy viết chương trình và kiểm tra các khả năng sinh lỗi khi thực hiện chương trình.

Bước 2. *Thực hiện nhiệm vụ*

- HS thực hiện nhiệm vụ

Bước 3. *Báo cáo, thảo luận*

- Cho một hs lên viết lại chương trình của mình lên máy tính
- Các hs khác thảo luận cho ý kiến.
- Hs chạy thử chương trình để kiểm tra sinh lỗi

Bước 4. *Kết luận, nhận định*

- GV nhận xét bài làm của học sinh, chuẩn hóa kiến thức

```
1  n=int(input("Nhập số tự nhiên n"))
2  A=[]
3  for i in range(1,n):
4      if n%i==0:
5          A.append(i)
6  print(A)
7
```

Các khả năng sinh lỗi của chương trình:

- Số n nhập vào không phải là số tự nhiên.
- Viết n = input(): lỗi chưa đổi kiểu dữ liệu.
- Cho range (1, n + 1): khi đó, kết quả cho cả n.

10 câu hỏi trắc nghiệm bài 29 nhận biết lỗi chương trình

Câu 1. Có mấy loại lỗi mà người lập trrình thường mắc khi giải bài toán bằng ngôn ngữ lập trinh Python

A. 3

B. 4

C. 1

D. 2

Câu 2. Trong các loại lỗi trên lỗi nào khó phát hiện – chỉ được phát hiện bởi con người?

A. Lỗi cú pháp

B. Lỗi ngữ nghĩa

C. Lỗi ngoại lệ

D. Lỗi cấu trúc

Câu 3. Lỗi khi không thực hiện một lệnh trong chương trình. Chương trình dừng lại và thông báo 1 mã lỗi . Lỗi này được gọi là lỗi ….

A. Lỗi cú pháp

B. Lỗi ngữ nghĩa

C. Lỗi ngoai lệ

D. Lỗi cấu trúc

Câu 4. Lỗi khi mà chương trình chạy không có lỗi ngoại lệ nhưng kết quả đưa ra sai, vậy đây là lỗi …

A. Lỗi cú pháp

B. Lỗi ngữ nghĩa

C. Lỗi ngoại lệ

D. Lỗi cấu trúc

Câu 5. Lệnh sau có lỗi nào? Print (“Ket qua phep chia la”, 145/0)

A. Lỗi cú pháp có mã lỗi là SyntaxError

B. Lỗi ngữ nghĩa có mã lỗi là ZeroDivisionError

C. Lỗi ngoại lệ có mã lỗi là ZeroDivisionError

D. Lỗi cấu trúc có mã lỗi là ZeroDivisionError

Câu 6. Lệnh sau có lỗi nào? Print (“Ket qua phep chia la , 145/5)

A. Lỗi cú pháp có mã lỗi là SyntaxError

B. Lỗi ngữ nghĩa có mã lỗi là ZeroDivisionError

C. Lỗi ngoại lệ có mã lỗi là ZeroDivisionError

D. Lỗi cấu trúc có mã lỗi là ZeroDivisionError

Câu 7. Lệnh sau có lỗi nào?

```
if (a==b):
        print (“a bang b”)
else:
        print (“a lớn hơn b”)
```

A. Lỗi cú pháp

B. Lỗi ngữ nghĩa

C. Lỗi ngoại lệ

D. Lỗi cấu trúc

Cho dãy lệnh giải phương trình bậc hai

```
delta = b*b-*a*c
if (delta>=0):
```

print (“Phương trình đã cho có nghiệm”)

else:

print (“Phương trình đã cho vô nghiệm”)

Câu 8. Dãy lệnh trên có lỗi ngữ nghĩa không? Nếu có là dòng lệnh nào?

A. Có, dòng lệnh tính delta

B. Có, dòng lệnh kiểm tra điều kiện delta

C. Có, dòng lệnh đưa ra thông báo sau else

D. Không

Câu 9. Dãy lệnh trên có lỗi nào khác nữa không?

A. Có, lỗi ngoại lệ

B. Có, lỗi cấu trúc

C. Có, lỗi ngữ pháp

D. Không

Câu 10. Em hãy sửa lại lỗi ngữ nghĩa trong dãy lệnh trên.

A. delta = b*b-4*a*c

B. if (delta >0)

C. Câu lệnh print sau else sẽ là print (“Phương trình có nghiệm kép)