

# Điều khiển luồng

Th.S Trần Đức Lợi  
[Pythonvietnam.info](http://Pythonvietnam.info)

# Ôn tập bài cũ

- Ôn tập git
  - Đồng bộ
  - Add/rm cùng lúc
- Chữa bài python-calculator
- Chữa bài tập về nhà python-calculator-with-fibonacci

# Mục đích bài học

- Tìm hiểu về điều khiển luồng trong python với các câu lệnh if-else, while, for, ...
- Cấu trúc try-catch

# Điều khiển rẽ nhánh

- Đặt vấn đề: Viết thuật toán và code chương trình *python18+*:
  - Chào mừng người dùng
  - Hỏi năm sinh
  - Nếu người dùng nhỏ hơn 18 tuổi thì thông báo không đủ tuổi
  - Nếu đủ hoặc hơn 18 tuổi thì thông báo chào mừng

# Điều khiển rẽ nhánh if

- *If*  $a == b$  :
- *Print* ""
- *Else* :
- *Print* ""
- *Print* "finished."
- $<, <=, >, >=, !=$

# Điều khiển rẽ nhánh if

- Biểu thức boolean dùng để điều khiển rẽ nhánh
- Yêu cầu trả về giá trị True/False
- Các biểu thức điều kiện chỉ đọc giá trị các biến chứ không thay đổi giá trị của biến

# Điều khiển rẽ nhánh đa quyết định

- Đặt vấn đề: Viết thuật toán và code chương trình *python18++*:
  - Chào mừng người dùng
  - Hỏi năm sinh
  - Nếu người dùng nhỏ hơn 16 tuổi thì thông báo không đủ tuổi
  - Nếu đủ hoặc hơn 18 tuổi thì thông báo chào mừng
  - Nếu từ 16-18 tuổi thì thông báo nên cân nhắc

# Điều khiển rẽ nhánh if

- *If*  $a == b$  :
- *Print* «»
- *Elif*  $b == c$  :
- *Print* «»
- *Else:*
- *Print* «»



# Bài tập

- Bài đoán số: hãy nhận dữ liệu từ người dùng và so sánh với bình phương của số đó cũng do người dùng nhập vào xem có bằng nhau không? Nếu không thì đưa ra gợi ý quá cao hoặc quá thấp.»

# Switch

- Hãy sử dụng if – elif – else hoặc sử dụng dictionary
- Không có câu lệnh kiểu này trong python

# Vòng lặp While

- *While a == b:*
- *Print «»*
- *Else:*
- *Print «the loop is ended»*
- *Print «finished.»*
- Hãy sử dụng vòng while để thực hiện bài toán đoán số liên tục cho đến khi bấm phím «q»

# Vòng lặp for

- *For i in range( 1,5):*
- *Print «»*
- *Else:*
- *Print «»*
- Hãy sử dụng vòng for để thực hiện bài toán đoán số liên tục cho đến khi bấm phím «q»

# Câu lệnh break

- Thoát khỏi vòng lặp bằng câu lệnh break
- Hãy sử dụng câu lệnh break để thoát khỏi chương trình khi người dùng nhấn «q»

# Lệnh continue

- Bỏ qua một số câu lệnh trong vòng loop bằng lệnh continue
- Các dòng lệnh trong vòng loop và sau câu continue sẽ bị bỏ qua, không thực hiện mà chuyển sang vòng loop tiếp theo luôn

# Lệnh continue

- Sử dụng câu lệnh continue để thực hiện thêm chức năng nếu người dùng nhấn «x» thì thông báo bạn đã hủy lần đoán này, và tự động sinh ra số mới.

# Exception

- Exception hay ngoại lệ xảy ra theo một tình huống nào đó trong một chương trình.
- Ví dụ:
  - - đọc file nhưng file không tồn tại
  - - delete file khi đang sử dụng
  - ...
- và từ đây sinh ra khái niệm bắt ngoại lệ.



# Exception

- Nên bao quanh các đoạn code “nguy hiểm” bằng đoạn try...except
- Trong đoạn này, nếu try block được thực thi thì except block sẽ được bỏ qua và **ngược lại**
- Bài pythoncalculator

# Exception

- try:
- `f = open('x')`
- except:
- `print 'Exception occurred'`
- `print 'prog will now exit'`

# Exception

- try:
- f = open('x')
- except IOError:
- print 'caught'
- except:
- print 'x'
- print 'prog will now exit'

# Cấu trúc try-catch

- `d = int(raw_input("Nhap gioi han cua day fibonacci: "))`
- Try:
- Print «»
- Except Exception, e:
- Print «»

# Raise Exception

- Exception có sẵn của python
- try:
- raise IOError()
- except IOError, e:
- print e
- except:
- print 'x'

# Try ... Finally

- try:
- raise LoiEx('I hate')
- #f = open('x')
- except IOError, e:
- print e
- finally:
- print 'I am the final'

# Bài tập

- Lưu ý về cách thức ra bài tập
- Xây dựng một chương trình python stickers với điều khiển luồng có bắt exception và thoát khi người dùng bấm phím «q»
-