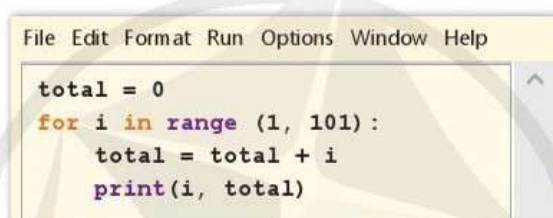


Học xong bài này, em sẽ:

- ✓ Viết được chương trình đơn giản có sử dụng câu lệnh lặp.
- ✓ Viết được chương trình đơn giản có sử dụng câu lệnh rẽ nhánh kết hợp với câu lệnh lặp.

Bài 1. Làm quen với câu lệnh lặp trong Python

Em hãy dự đoán xem chương trình ở *Hình 1* sau đây sẽ đưa ra màn hình những gì. Chạy chương trình để kiểm tra kết quả.

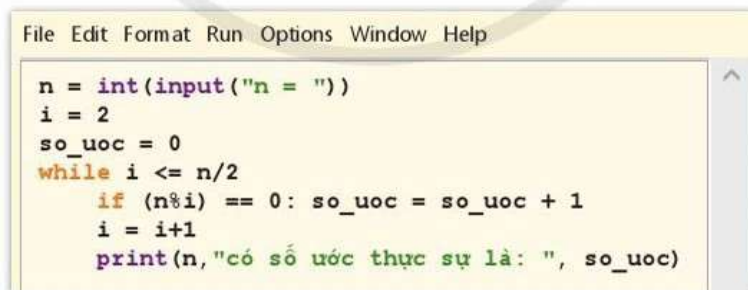


```
File Edit Format Run Options Window Help
total = 0
for i in range (1, 101):
    total = total + i
    print(i, total)
```

Hình 1. Chương trình với câu lệnh lặp while

Bài 2. Đếm các ước thực sự của một số nguyên

Bạn Hà viết chương trình ở *Hình 2* để đếm xem số nguyên n nhập vào từ bàn phím có bao nhiêu ước số thực sự (ước khác 1 và n). Tuy nhiên, chương trình chạy ra kết quả sai. Em hãy sửa lỗi giúp bạn Hà.



```
File Edit Format Run Options Window Help
n = int(input("n = "))
i = 2
so_uoc = 0
while i <= n/2:
    if (n%i) == 0: so_uoc = so_uoc + 1
    i = i+1
print(n, "có số ước thực sự là: ", so_uoc)
```

Hình 2. Chương trình của bạn Hà

Bài 3. Nhập dữ liệu có kiểm tra

Tham khảo chương trình ở *Ví dụ 5* trong Bài 8, em hãy viết chương trình yêu cầu người dùng nhập một số nguyên lớn hơn 1 000 000. Chừng nào người dùng nhập chưa đúng yêu cầu thì có thông báo yêu cầu nhập lại, chương trình chỉ kết thúc với thông báo “Cảm ơn, bạn đã nhập dữ liệu đúng yêu cầu.” khi số người dùng gõ vào thỏa điều kiện đặt ra.



Em hãy lập trình giải bài toán cổ ở hình bên một cách tổng quát bằng cách nhập hai số nguyên dương n , m tương ứng là tổng số con và tổng số chân sau đó đưa ra màn hình số lượng gà và số lượng chó. Kiểm thử chương trình với $n = 36$ và $m = 100$.

Vừa gà vừa chó
Bó lại cho tròn
Ba mươi sáu con
Một trăm chân chẵn
Hỏi có mấy con gà, mấy con chó?

BÀI TÌM HIỂU THÊM

CÁC CÂU LỆNH BREAK VÀ CONTINUE

Trong Python câu lệnh **break** dùng để thoát ra khỏi vòng lặp ngay kể cả khi điều kiện lặp còn đúng. Bất cứ những lệnh nào trong vòng lặp đứng sau **break** đều bị bỏ qua. Câu lệnh **continue** trong Python được dùng để bỏ qua các câu lệnh còn lại chưa được thực hiện trong vòng lặp, chuyển đến vòng lặp tiếp theo.

Ví dụ 1. Chương trình ở hình bên sử dụng **break** và **continue** trong câu lệnh **for** giải quyết bài toán sau:

Bài toán: Hãy viết chương trình nêu những câu hỏi để kiểm tra xem người ngồi trước máy tính có thuộc bảng nhân 6 hay không. Chương trình cho phép người trả lời bỏ qua một câu hỏi nào đó hoặc dừng kiểm tra.

```
File Edit Format Run Options Window Help
#Kiểm tra bảng nhân 6
bang = 6
for i in range(1, 11):
    print(bang, "x", i, " = ?")
    tra_loi = input()
    if tra_loi == "dừng": break
    if tra_loi == "bỏ qua":
        print("Không nhớ, bỏ qua")
        continue
    dap_an = i * bang
    if int(tra_loi) == dap_an:
        print("Đúng!")
    else:
        print("Sai! đáp án: ", dap_an)
print("Kết thúc")
```

Ví dụ 2. Chương trình ở hình sau đây sử dụng **break** và **continue** trong câu lệnh **while** để giải quyết bài toán sau:

Bài toán: Để thử nghiệm lâm sàng vacxin mới ở giai đoạn 1, người ta cần những người trong độ tuổi từ 18 đến 64 tuổi và thỏa mãn điều kiện $18.5 \leq \text{cân nặng}/(\text{chiều cao})^2 \leq 22.9$. Theo tập hồ sơ nhận được từ những người tình nguyện, hãy đưa ra màn hình số người sẽ được xét để tham gia thử nghiệm. Số liệu về tuổi, cân nặng (kg) và chiều cao (m) của mỗi hồ sơ nhập vào từ bàn phím, mỗi số trên một dòng. Nhập tuổi bằng 0 để kết thúc tập hồ sơ.

```
File Edit Format Run Options Window Help
p = 0
while True:
    print("Người tình nguyện:")
    tuoi = int(input("Tuổi: "))
    if tuoi == 0: break
    cao = float(input("Cao(m): "))
    nang = float(input("Nặng(kg): "))

    if (18 <= tuoi < 65 and not(18.5 <=
        nang/cao**2 <= 22.9)): continue
    p = p + 1
    print("Số người được xét:", p)
```