

BÀI TẬP SINH TEST LỚP TƯƠNG ĐƯƠNG TỰ ĐỘNG

Viết chương trình sinh các ca kiểm thử giá trị biên tự động cho hàm “main(x, y, z, ...)” viết bằng ngôn ngữ Python. Hàm Python được khai báo và cài đặt trong file “input.py”, nhiệm vụ của các bạn là viết một chương trình Python lấy đầu vào là file “input.py”, xử lý và cho đầu ra là các ca kiểm thử được viết dưới dạng unit test được lưu ở file “output.py”.

Input: Cho trong file “input.py”, trong đó có hàm main(x, y, z, ...) với x, y, z là các biến đầu vào, hàm main có thể có n ($0 < n \leq 10$) biến đầu vào. Ngay sau dòng khai báo hàm là n + 2 dòng comment, mỗi dòng sẽ khai báo khoảng của biến đầu vào theo format “<biến><:>(<[><int><;><int><]>)*<n>” (trong đó 2 dòng đầu và cuối là chuỗi <”> là dấu doc string của ngôn ngữ Python).

Output: Lưu ở file “output.py” ở bất kì dạng nào sao cho “output.py” chứa toàn bộ ca kiểm thử giá trị biên dành cho hàm main(x, y, z, ...). Lưu ý file này phải là file chạy được (không có lỗi cú pháp, không có lỗi runtime), các ca kiểm thử chỉ quan trọng đầu vào mà không quan trọng đầu ra. Nếu các khoảng [int; int] không hợp lệ (không phải là các lớp tương đương thì chương trình raise ra Exception với message là “wrong input”)

Hệ số điểm: 10%

Deadline: 2 tuần

Hard deadline: 3 tuần (50% số điểm sẽ bị trừ khi nộp sau 2 tuần và trước 3 tuần, 100% số điểm sẽ bị trừ khi nộp sau 3 tuần, nhưng vẫn được thông báo chương trình đúng hay sai)

Cách chấm tự động: Chương trình chấm tự động sẽ có 10 hàm main(x, y, z, ...) với số biến đầu vào khác nhau và khoảng của các biến khác nhau. Chương trình sinh đủ các ca kiểm thử lớp tương đương mạnh với mỗi hàm sẽ được 1 điểm, thiếu 1 ca kiểm thử sẽ bị trừ điểm tương đương với ca kiểm thử đó (ví dụ có 10 ca kiểm thử, điểm của mỗi ca kiểm thử là 0.1 điểm)

Ví dụ:

Input

```
#Determine if input is a triangle using triangle inequality function
```

```
def is_triangle(a, b, c):
```

```
    return True if (a < b + c and b < a + c and c < a + b) else False
```

```
def is_equilateral(a, b, c):
```

```
    return True if a == b == c else False
```

```
def is_isosceles(a, b, c):
```

```
    return True if a == b or b == c or a == c else False
```

```
#Triangle determiner:
```

```
# Input: length of the 3 edges
```

```
# Output: Not a triangle | type of triangle
```

```
def main(a, b, c):
```

```
    '''
```

```
    a: [0; 10] [11; 20]
```

```
    b: [0; 10]
```

```
    c: [0; 10]
```

```
    '''
```

```
    if is_triangle(a, b, c):
```

```
        if is_equilateral(a, b, c):
```

```
            return 'Equilateral triangle'
```

```
        elif is_isosceles(a, b, c):
```

```
            return 'Isosceles triangle'
```

```
        else:
```

```
            return 'Triangle'
```

```
    else: return 'Not a triangle'
```

