

# Python for Data Science

## Midterm exam

Thời gian làm bài: 60 phút

1. (2đ) Một cửa hàng bán áo phông với giá 20\$ 1 cái. Mức độ giảm giá phụ thuộc vào số lượng áo mua như sau:

Số lượng	Mức giảm giá
1-9	0%
10-19	10%
20-39	25%
>40	40%

Cho người dùng nhập số lượng áo muốn mua, tính ra thành tiền.

2. (3đ) Cho string thể hiện số mặt con xúc xắc sau 1 số lần tung: "1,5,2,3,5,4,4,3,1,1,1,2,3,1,5,6,2" (lần 1 được mặt 1, lần 2 được mặt 5, ...)
- (1đ) Tính số lần tung xúc xắc.
  - (1đ) Tính tổng tất cả các mặt của các lần tung.
  - (1đ) Mặt nào xuất hiện nhiều nhất trong các lần tung ở trên.
3. (4đ)
- (a) (1đ) Viết hàm nhập vào 1 số nguyên dương, tính số chữ số của nó.
- (b) (2đ) Viết hàm tìm kiếm tra 1 số có tính chất sau: tổng các chữ số ở vị trí chẵn bằng tổng các chữ số ở vị trí lẻ. Ví dụ: số 143 thỏa mãn vì  $4 = 1 + 3$ , số 2530 thỏa mãn vì  $2 + 3 = 5 + 0$ , số 2890 không thỏa mãn vì  $2 + 9 \neq 8 + 0$ .
- (c) (1đ) In ra các số nguyên dương nhỏ hơn 1000 thỏa mãn tính chất phần b.
4. (1đ) Cho 1 lưới  $n \times n$  với các ô có 2 trạng thái: còn sống hoặc đã chết. 1 ô bất kỳ có nhiều nhất 8 ô hàng xóm của nó ( ngang, dọc, chéo). Cho 1 lưới với một số ô có thể còn sống hoặc đã chết ở thể hệ hiện tại. Các ô ở thể hệ tiếp theo sẽ là ô sống hay ô chết được xác định dựa vào các ô ở thể hệ hiện tại với các quy tắc sau:

1	2	3
4		5
6	7	8

Hình 1: Ví dụ các ô hàng xóm

- Bất kỳ ô sống nào có ít hơn 2 người hàng xóm còn sống thì sẽ chết.
- Bất kỳ ô sống nào có 2 hoặc 3 người hàng xóm còn sống thì sẽ sống.
- Bất kỳ ô sống nào có nhiều hơn 3 người hàng xóm còn sống thì sẽ chết.
- Bất kỳ ô chết nào có đúng 3 hàng xóm sống sẽ trở thành ô sống.
- Các ô chết còn lại thì vẫn chết.

Cho ô lưới 5x5, 1 đại diện cho ô còn sống, 0 đại diện cho ô đã chết. Viết hàm dự đoán thế hệ tiếp theo của ô lưới đã cho (input là lưới 5x5 thế hệ hiện tại, output là lưới 5x5 thế hệ tiếp theo). Các ô ở viền thì không cập nhật trạng thái (chỉ cập nhật các ô có đủ 8 ô hàng xóm).

0	0	0	0	0
0	1	1	0	0
0	1	1	0	0
0	0	1	0	0
0	0	0	0	0

(a) Input

0	0	0	0	0
0	1	1	0	0
0	0	0	1	0
0	1	1	0	0
0	0	0	0	0

(b) Output

Hình 2: Ví dụ chuyển trạng thái