

Computer Science Basic

MINI HACKATHON 3 - THỜI GIAN: 110 PHÚT

Phần 1. Thời gian: 20 phút

| Bài | Nội dung | Kết quả mong đợi |
|-----|--|---|
| 1 | Viết hàm kiểm tra số chẵn. Return True nếu số nhập vào là số chẵn, return False nếu số nhập vào là số lẻ. Sử dụng hàm trên để kiểm tra số nguyên người dùng nhập vào là chẵn hay lẻ. | Input a number: 3 This number is not even |
| 2 | Viết hàm cal_area() trả về diện tích của một hình tròn với đầu vào là bán kính. Sử dụng hàm trên để tính diện tích hình tròn từ bán kính nhập từ người dùng. | Input radius: 1 Circle's area: 3.14 |
| 3 | Viết hàm reverse_str() dùng để đảo ngược một string. Sử dụng hàm trên để đảo ngược nội dung nhập từ người dùng | Input a text: mindX Reversed text: Xdnim |
| 4 | Palindrome là một string có dạng đảo ngược giống hệt chính nó. Ví dụ: anna, nan, hannah,... Viết và sử dụng hàm is_palindrome() để kiểm tra một string nhập vào có phải palindrome hay không. | Input a text: anna This is a palindrome. |

*Phần in đậm trong kết quả mong đợi thể hiện nội dung nhập từ người dùng.

Phần 2. Thời gian: 20 phút

| Bài | Nội dung | Kết quả mong đợi |
|-----|---|--|
| 1 | Viết hàm tính giai thừa của số nguyên do người dùng nhập vào. Biết giai thừa của số nguyên n: $n! = 1.2.3...(n-1).(n)$ | Input a number: 5 $5! = 120$ |
| 2 | Cho list chứa các số nguyên 5, 1, 8, 92, -1, 30. Viết hàm sắp xếp các số trong list theo thứ tự từ bé đến lớn, thực thi hàm và in kết quả ra màn hình. Không sử dụng các hàm sort() và sorted(). | Original list: 5 1 8 92 -1 30 Sorted list: -1 1 5 8 30 92 |
| 3 | Viết hàm print_fibo(n) in ra n phần tử đầu tiên của dãy Fibonnaci. Biết dãy Fibonacci có tính chất: giá trị của một phần tử bằng tổng giá trị hai phần tử liền trước nó. Một số phần tử đầu tiên của dãy: 1, 1, 2, 3, 5, 8, 13, 21,... Yêu cầu người dùng nhập vào số $n > 0$ và sử dụng hàm trên để in ra dãy Fibonacci tương ứng. | Input a number: 5 First 5 Fibonacci numbers: 1 1 2 3 5 |

Phần 3. Thời gian: 70 phút

Nội dung

Làm game Escape trên terminal: Người chơi đi nhặt chìa khoá để mở cửa thoát ra ngoài.

- P: Player, di chuyển bằng các phím A, S, D, W được nhập bằng input qua bàn phím.
- K: Key. Player di chuyển đến K để lấy khoá. Khi di chuyển đến K thì K biến mất khỏi map.
- D: Door, sau khi lấy được K, đến D thì người chơi thắng. Nếu đến D trước khi đến K thì người chơi thua.

```
P - - - - -
- - - - - K - - - -
- - - - -
D - - - - -
- - - - -

== THE ESCAPE GAME ==
Use W A S D to move (P)layer.
Find (K)ey first then exit through the (D)oor.
```

Hướng dẫn: làm theo các bước sau:

1. Viết hàm khởi tạo bản đồ game.
2. Viết hàm xác định hướng di chuyển dựa vào phím người dùng nhập.
3. Viết hàm thay đổi vị trí Player sau mỗi lần di chuyển. Kiểm tra Player có đi vào ô K hay D không.

Một số lệnh hỗ trợ:

Xóa màn hình để cập nhật bản đồ game:

```
import os

os.system('cls') # Windows
os.system('clear') # macOS
```

Nhận phím người dùng nhập mà không cần ấn Enter

| | |
|-------------------------------------|------------------------------------|
| # Windows | # macOS |
| import msvcrt | import getch |
| ch = msvcrt.getch().decode('utf-8') | ch = getch.getch().decode('utf-8') |

Với macOS, sử dụng lệnh `pip install getch` để cài đặt module **getch**. [Nguồn](#)