

Hàm sinh số ngẫu nhiên

- Ý nghĩa và cách sử dụng
- Enum
- Mini game casino

Ý nghĩa

- Tạo ra các giá trị số nguyên ngẫu nhiên sử dụng trong mật khẩu cấp lần đầu cho người dùng hoặc các trò chơi mang tính chất random
- Các giá trị ngẫu nhiên được sinh ra nằm trong đoạn [0, RAND_MAX]
- Giá trị RAND_MAX có thể thay đổi tùy hệ điều hành, thường tối thiểu là 32767 và được định nghĩa trong file header <stdlib.h>

Cách sử dụng

- Để sinh số ngẫu nhiên ta dùng hàm `rand()` trong thư viện `<stdlib.h>`
- Nếu muốn sinh số ngẫu nhiên r trong đoạn $[1, n]$ với n nguyên dương không quá `RAND_MAX` ta làm theo công thức: $r = 1 + rand() \% n;$
- Khi chạy chương trình với hàm `rand()` thì hầu như chúng ta sẽ nhận được cùng một bộ kết quả, để các bộ kết quả khác nhau ta sử dụng hàm `srand(time(NULL));`
- Trong đó `time(NULL)` sẽ trả về thời gian(số giây) hiện tại của hệ thống nên ta `include<time.h>` vào đầu chương trình

Cách sử dụng

- Mỗi lần chạy chương trình thì `srand(time(NULL))` sẽ làm nhiệm vụ thay đổi các bộ số random được sinh ra. Nếu tham số của `srand()` là cùng 1 giá trị thì hàm `rand()` luôn sinh ra cùng một bộ số random không đổi
- Nếu muốn sinh ra các giá trị ngẫu nhiên r trong đoạn $[a, b]$ thì công thức là: $r = a + \text{rand()} \% (b - a + 1);$

Giới thiệu về enum

- Enum là một kiểu tập hợp của các hằng số kiểu nguyên, mỗi hằng số sẽ được đại diện bởi một định danh
- Giá trị mặc định của enum bắt đầu từ 0, phần tử kế tiếp sẽ có giá trị là 1, ...
- Ta có thể chủ động thay đổi các giá trị cho từng định danh
- Tên định danh trong enum là chữ hoa vì chúng là hằng số
- Ví dụ:

```
enum Status {CONTINUE, WON, LOST};
```

Giới thiệu về enum

- Tên các phần tử trong enum bắt buộc phải khác nhau nhưng giá trị của chúng có thể trùng nha

```
enum Status {CONTINUE = 10, WON = 20, LOST = 60};
```

```
// enum  
enum DayOfWeek {MON = 1, TUE, WED, THUR, FRI, SAT, SUN};  
// use enum  
enum DayOfWeek today = SUN;  
printf("%d", today);
```

Tiếp theo

Quy tắc phạm vi của một định danh