

Bài 12.10: Lập lịch cho các luồng

- ✓ Lập lịch cho các luồng

Lập lịch cho luồng

- ✓ Mỗi thread đều có thứ tự ưu tiên gắn liền với nó.
- ✓ Các luồng tạo ra trong CLR sẽ được gán giá trị ưu tiên ThreadPriority.Normal theo mặc định.
- ✓ Các thread tạo ra bên ngoài runtime sẽ giữ thứ tự ưu tiên mà nó có trước khi vào môi trường được quản lý.
- ✓ Ta có thể thiết lập và lấy giá trị ưu tiên của từng thread qua thuộc tính Priority.
- ✓ Các luồng được lập lịch để thực thi dựa trên thứ tự ưu tiên của chúng. Các luồng sẽ được gán thời gian sử dụng vì xử lý thường gọi là lát cắt thời gian bởi hệ điều hành.
- ✓ Chi tiết về thuật toán lập lịch sử dụng để xác định thứ tự thực hiện của các thread sẽ tùy thuộc từng hệ điều hành.
- ✓ Thông thường, luồng(có thể chạy) có thứ tự ưu tiên cao nhất sẽ được lên lịch để chạy trước.

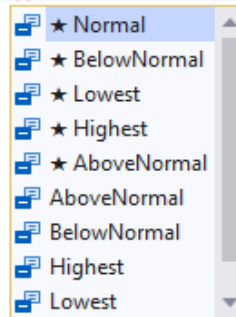
Lập lịch cho luồng

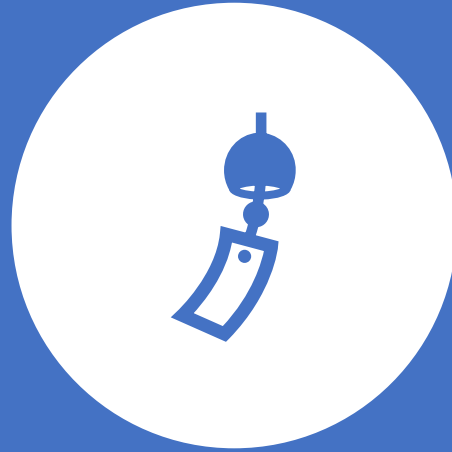
- ✓ Nếu có nhiều thread cùng thứ tự ưu tiên, bộ lập lịch sẽ quay vòng các thread đó, cấp cho mỗi thread một khung thời gian nhất định để sử dụng vi xử lý.
- ✓ Chừng nào còn có các thread với thứ tự ưu tiên cao hơn, các thread có thứ tự ưu tiên thấp hơn sẽ không được hoạt động.
- ✓ Khi nào không còn thread với mức ưu tiên cho trước, bộ lập lịch sẽ chuyển sang thứ tự ưu tiên thấp hơn kế tiếp và lên lịch để các thread ở mức ưu tiên đó được chạy.
- ✓ Nếu đột nhiên có một thread với thứ tự ưu tiên cao hơn chuyển trạng thái sang có thể chạy, các thread có thứ tự ưu tiên thấp hơn sẽ bị chiếm chỗ và lúc này thread với thứ tự ưu tiên sẽ được hoạt động.
- ✓ Hệ điều hành có thể điều chỉnh thứ tự ưu tiên một cách chủ động giống như việc 1 thread chuyển từ foreground sang background.
- ✓ Mỗi hệ điều hành có thể chọn một thuật toán lập lịch khác nhau.

Lập lịch cho luồng

✓ Ví dụ các thứ tự ưu tiên của một thread:

```
t.Priority = ThreadPriority.;
```





Nội dung tiếp theo

Tạm dừng và ngắt luồng