

2021-06-23

Esercizio c.1

Esercizio c.2

```
class stackprio:
    int prio;
    lista = [];
    length = 0
    semaforo_binario s;

add(int prio, semaforo_biario):
    if length == 0:
        lista.append( new stackprio

class semstack:
    semaforo_binario mutex
    lista head
    int value = 0

    void init(int val):
        value = val

    void add_stack(int prio, semaforo a) {
        if (head

    void PLP(int prio) {
        mutex.P();
        if (value > 0) {
            value--;
            mutex.V();
        } else {
            Semaforo a = semaforo(0);
            add_stack(prio, a);
            mutex.V()
            a.P();
        }
    }
}
```

### **Esercizio g.1**

### **Esercizio g.2**

- a) perché DMA viene utilizzato per le unità di memoria secondaria (es dischi) e non per terminali? DMA viene utilizzato per memoria secondaria perchè si ha necessità di trasferire interi blocchi. Mentre
- b) cosa succede in un sistema operativo quando un processo utente tenta di eseguire una operazione illegale (es. divisione per zero)? Lo standard POSIX (UNIX) cosa prevede in questo caso?

Viene generata una trap. La trap viene intercettata dal sistema operativo che sceglie come gestirla (solitamente viene ucciso il processo). Il processo potrebbe essere equipaggiato di un exception handler, in tal caso l'esecuzione potrebbe essere trasferita ad esso (tale handler sarebbe all'interno del PCB)

- c) L'algoritmo di rimpiazzamento second chance (detto anche dell'orologio) è a stack? Perché viene preferito a LRU?

Non è a stack, potrebbe soffrire dell'anomalia di Belady.

- d) Quali sono le caratteristiche negative dell'uso delle Access Control List per la memorizzazione delle informazioni di autorizzazione.