

uniprocessor scheduling esercizi – B (soluzione)

VRR + Feedback

- considera la seguente politica
 - VRR (processi in ready e in aux, $q=10\text{ms}$)
 - sia ready che aux sono sostituite con piu' code
 - le code ready sono gestite con una politica di feedback (priorità scalata ad ogni quanto)
 - running \rightarrow blocco: ricorda la sua priorità p e la rimanente porzione di quanto
 - blocco \rightarrow aux: $p = p - \text{floor}(t/50\text{ms})$
 - aux \rightarrow ready: quanto finisce il suo quanto va in ready con priorità invariata
 - p rimane sempre tra 0 e 9, 0=priorità massima

VRR + Feedback

- considera tre processi con un comportamento regolare che alterna cpu-burst a I/O-busrt
 - A: 1005ms di cpu e 25ms di I/O
 - B: 25ms di cpu e 1005ms di I/O
 - C: 505ms di cpu e 205ms di I/O
- descrivi l'evoluzione di ciascun processo
- supponi che ciascun processo sia solo nel sistema

VRR + Feedback

- A: 1005ms di cpu e 25ms di I/O
 - 1 p viene incrementato per 9 volte, dopo 90ms A e' in ready 9
 - 2 running da ready 9 per 915ms.
 - 3 blocked per 25ms con un debito di quanto di 5ms.
 - 4 running da aux 9 per 5ms
 - 5 running da ready 9 per 1000ms (supponiamo che l'operazione di I/O avvenga subito prima dello scadere del quanto)
 - 6 blocked per 25ms con un debito di quanto di 0ms.
 - 7 running da aux 9 per 0ms
 - 8 running ready 9 per 1005ms
 - 9 goto 3

VRR + Feedback

- B: 25ms di cpu e 1005ms di I/O
 - 1 running da ready 0 per 10ms
 - 2 running da ready 1 per 10ms
 - 3 running da ready 2 per 5ms.
 - 4 blocked per 1005ms con un debito di quanto di 5ms.
 - $\text{floor}(1005/50) > 2 \rightarrow p=0$
 - 5 running da aux 0 per 5ms
 - 6 running da ready 0 per 10ms
 - 7 running da ready 1 per 10ms (supponiamo che l'operazione di I/O avvenga subito prima dello scadere del quanto)
 - 8 blocked per 1005ms con un debito di quanto di 0ms.
 - 9 running da aux 0 per 0ms
 - 10 goto 1

VRR + Feedback

- C: 505ms di cpu e 205ms di I/O
 - 1 p viene incrementato per 9 volte, dopo 90ms C e' in ready 9
 - 2 running da ready 9 per 415ms
 - 3 blocked per 205ms con un debito di quanto di 5ms.
 - all'uscita: $p = p - \text{floor}(205/50) = 5$
 - 4 running da aux 5 per 5ms
 - 5 running da ready n per 10ms ($n=5\dots 8$)
 - 6 running da ready 9 per 460ms (supponiamo che l'operazione di I/O avvenga subito prima dello scadere del quanto)
 - 7 blocked per 205ms con un debito di quanto di 0ms
 - all'uscita: $p = p - \text{floor}(205/50) = 5$
 - 8 running da aux 5 per 0ms
 - 9 running da ready n per 10ms ($n=5\dots 8$)
 - 10 running da ready 9 per 465ms
 - 11 goto 3