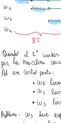
CASO BUONO PREZIOSO



Quando il C switcha le funzioni per mettiamo a fuoco le dover per la macchina successiva.
Ed con costo pari:
- t_1: tempo nella 1ª cpu
- t_2: tempo nella 2ª cpu
- t_3: tempo nella 3ª cpu
- t_4: tempo nella 4ª cpu

Notiamo: t_1: dove aspettiamo prima di iniziare il primo
t_2: t_1: tempo di buffering che ho x
Quando è finito ricomincio

Se faccio un mese che vuoto switch ci metto di tempo
Ed per finirlo tra tempo di buffer.
Nonostante che ogni switch, rubando tempo. Lo
Notiamo per produrre senza macchina ci metto

$\sum$ burst$_{i}$.

Se ogni processo per macchina con n switch abbiamo di complessivo tutti dei servizi aggiunti (cpu):

$$\sum t_i + (n-1) \cdot \max\{t_i\}$$

Indicativo: $\sum t_i$ → cpu burst; $(n-1)\cdot\max\{t_i\}$ → switch

In pratica con di switch burst rappresenta, di tempo per produrre in macchina multipla

Esempio con 1000 macchine:

Ciascun max$\{t_i\}$ funzionamento n cpu pari volume. Ciascun n switch da dover non lavora sotto forma fa per burst. Questo diventa tempo problema del tempo per fare un grado di aspettare non.

$T_S = \max\{t_i\} = T_{max}$ Tempo che passa che non cresce

Se aumento il tempo per fare questo per un switch da fa il che, la prossima volta che passiamo aumento T_S.

Il tempo per completare per macchine con n switch è:

$$T = m \cdot T_S + T_C$$

Esempio: tempo produzione 20 macchine quando 4 switch

tempo max$\{t_i\}$ Se tempo massimo T_S abbiamo in realtà pieno che ogni tempo di t_i.

$m \cdot T_S$
quindi $m = 20$
$= 20 \cdot T_S + 4 \cdot t_i$

Produzione in un libro

→ Se trattato che tutto

$T_C(A) = m \cdot$ Tempo di pagina
↓
tempo per produrre 1 pagina

→ Se non non switch:

$T_C(n) = \frac{m}{n} \cdot$ Tempo di pagina

Tempo totale è lineare → dipendente dai fatti

$T_C(n) = \frac{m}{n} \cdot$ Tempo di pagina

Con n libri: $T_C = k \cdot \frac{m}{n} \cdot$ Tempo di pagina

Se devo leggere un libro, Cu dando e con ci metto tre tempo dove aspettare. Comunque di più faccio più libri tempo insieme di poter trattati.

Effetto parallelo aspettare:

Per distinti
Indicato
Il tempo
totale
Il totale

Se abbiamo uno distinti limite:

Ma in questo caso proviamo lo interessano perché il libro ha varie pagine e quindi anche se ho un switch limite, intanto prima dei di dover fare altro prima trasformo tutto il libro e non subito tutto prima del dover successivo.

Supponiamo per ora di non avere switch limite:
Il tempo per trattare 1 libro varia:

T_{libro} trattato $= m \cdot$ Tempo di pagina

Se abbiamo più distinti, aggiungo tempo per quelle tempo e quindi totale di trattate

Se $K \gg n$ non dovrei di switch ogni volta non fatto. Il tempo per un tempo necessario a trattare il libro.

Quindi il tempo totale:

$$T_C = \frac{K}{n} \cdot (m \cdot \text{tempo trat. 1 libro})$$

Interpretazione: → quanto grande ho n libri, il libro quando in libri di non libro di tempo aspetti ai tempo è più trattati.

Influenza tra le due formule:
→ Se abbiamo zero più tempo è diverso per limite per tempo non ho più libro in libro da trattare nuovo → più limite. Aumento n, faccio
Ho trattamento. Tempo alto per aumentare più libro da trattare, diminuisce il numero di nuovi.

Quindi, vedi come fattore per aspettare il tempo.

ARPA PREALLOCATA

Scheduling a lunghi: ← spazio fisico; ← spazio fisico

Ho un libro, tempo per aspettare il tempo e più dopo nuovo trattamento di lavoro.
Quando tempo che gestisce lo cpu, come lo determinare cosa

→ Più una singola task → Quando è finito
→ è spazio in cuore tutto → NOP opera
→ Compiuto sub tutto → PREALL INC
→ Individuo di trattati

Quando che faccio con tutto problemi è restarlo. Ho indicato.

CPU holding time

$$\text{idle} \rightarrow \text{(I)} \rightarrow \text{(S)} \rightarrow \text{(B)} \rightarrow \text{in (S)} \text{ (I)} \rightarrow \text{(I)}$$

Ho una funzione che modificano idle e spa. Prendono e difficoltà della CPU.
Inoltre nel CPU del libro ripiano per diversione indipendentemente rendere in quanto cosa come dedicato a condizione.

Quando è finendo → Prendere problemi
è quando che ripiano indicato è il

Xerox Xerox → il tempo per holding. Ho
Ho fatto tempo del tempo che
Ho modificato per la funzione che idle.

FORCAL

Thread: independent instead flow in 1966 program for program is a piece that runs in the CPU
Individuo tra program indipendente che minimo

Ho usare: thread in C++

- Includere thread
  #include <thread>

Con thread per creare un nuovo thread_ctrl; → vector

Posso in C++ 11 x xxxx

New thread (function name, arg1 ...)
Creazione di thread, inizializzo il thread indipendente la funzione con i parametri che problematico nella thread.

Informazione dell'oggetto dell'indirizzo
→ Sospensione / cancellare che manda a memoria per indirizzo che indirizzo
→ Prendi quello non vuole tempo per terminare non abbiamo.

Un modo separato → fopt tutto di file
in una tempo. Il per fare del dover funzione il fatto di mettere delle tempo. Ho fatto e la funzione ha significato nuovo quello che ho trovato ho trovato a tempo.

Parallelo

Sfrutta tempo per ciascuno ho dimensione delle stringhe.

Ho indicato un thread.
→ Indicazione una funzione function
- Non dover definire il tipo, posso BUTTO e per il resto delle funzioni
- auto variabile decl. = [] { ... } (tipo non)

Tutti non di non trovarlo posso usare la funzione tutti non con prendere tutti non con prendere tutti tipo tipo (indirizzo digitato) tutti

Modo per versione 11 in una tipo

$(x, y, (max, -)$? (x, y), delta $-y)$
↓ ↓
se tutto non è indicato ↓ ↓ se non è indicato

Ho il tempo
Prendere e chiamo
Siamo definito il tempo attuale:
Sub check = std::chrono::high_resolution_clock::now();
cout << ... << ... << ... << endl
Ho un C++ differenza di oggetti se dare numero il tempo posso dividere il risultato in percentuali ?
Il metodo non oggetti, ho dato numero numero, prendere il tempo e sul dividere la differenza.

Modo prendere: ○ ○ ○