[說明]

你好~

輸入的時候在workload 後面加上 [CPU使用率][memory MB][disk使用率],例如,想要讓CPU跑在60%(與增加CPU使用率是不太一樣的,我是直接指定要跑在60%)與memory 多1000MB (都是以MB為單位)30M/S的寫入速度,就要輸入 ./workload 60 1000 30 ,可能輸入方式不是類似增加%數的方式,但還是能用來測試系統的stress,由於增加iowrite的方式是用對一個檔案寫入文字來控制,產生出的write.txt得在測試玩後刪掉,刪掉的指令也寫在makefile裡,可以用make clean來清除

[實作方法]

環境是ubuntu14.04 4顆CPU 4G RAM

CPU使用率可能會有些誤差,我自己實測在不同電腦與實機或是虛擬機跑出來的結果都不一樣QQ,我採用的方法是設置4個thread 分別配到4顆cpu 然後每次都會讀取現在的cpu使用率,如果想要跑的使用率小於目前的的使用率(例如目前是3% ,但想讓他跑到10%),則讓四條thread直接跑無窮迴圈增加各顆CPU使用率,反之,馬上讓他停下來(我在這邊利用wait 與signal)再透過signal handler每10000usec(經過N便的測試………..這個比較穩定)來刷新目前的cpu使用率,來達到讓thread去判斷目前該執行的動作,至於memory我是直接用calloc去分配char(1char=1Byte)然後設計讓使用者輸入MB的單位,iowrite也是用跟CPU一樣的概念讓他們去忙著寫入,超過目標值在用wait將他們停下來,詳細情況也都寫在程式碼的註解中,我盡量註解的清楚,如果有問題都能問我,麻煩了謝謝。

也先謝謝助教與老師辛苦的看作業了,謝謝。

[心得]

Stress我認為真的是這三者作業中最複雜的,也花了很多很多的時間在調究竟要讓他sleep多久,後來想到用signal wait,本以為這已經是最完美的方式了,可是實作依舊重重的打我臉XD,不知道為何在有些機器可行,有些機器無法,真的是看了很久很久哈哈,還請助教手下留情QQ