CS1550: Project 3-VM Simulation (Trace File: GCC.Trace)

Page Faults vs # Frames for **gcc.trace**:

\* WS Clock: Refresh = 1,000,000. Tau = 70

\* NRU: Refresh = 100

I set NRU’s refresh rate to 100 because as I was tuning the refresh rates for NRU (shown on page 3), I found that the page fault trend resembles a parabola. So I estimated the best refresh rate with lowest amount of page fault, which is 100.

Similarly, for WS Clock, I controlled tau and changed refresh rate, and found the trend to be downward sloping, so I think best refresh rate is 1,000,000 (no refresh). Then I controlled refresh rate and changed tau, and found the tau that results in the lowest amount of page fault to be approximately 70.

From the resulting graph that compares all 4 algorithms, opt results in the lowest amount of page faults as expected. However, for real life OS, optimal won’t be unrealistic, as it will require the OS to predict the future. Therefore, out of the remaining 3 algorithms, working set consistently results in the least amount of page faults regardless of frame sizes. The implementation is also nice because it uses a circular queue. So I think the best algorithm for real OS out of the 4 algorithms will be working set clock.

Disk Writes vs Frames for **gcc.trace**:

**Tuning Parameters for NRU:**

NRU: Page Faults vs Refresh Rates for Different Frame Sizes:

NRU: Page Faults vs Disk Writes for Different Frame Sizes:

The correlation between page faults and refresh rates for NRU resembles a parabola. Page faults starts relatively high when refresh rate is low and decline when refresh rate is around 100, then it increases again. After examining the parabola, I determined the best NRU refresh rate for gcc.trace to be approximately 100.

WS Clock: Page Faults vs Refresh Rates (Frame = 16, Tau = 50)

|  |  |
| --- | --- |
| Refresh Rates | Page Faults |
| 10 | 125173 |
| 30 | 122946 |
| 50 | 121475 |
| 70 | 120245 |
| 90 | 119568 |
| 110 | 118966 |
| 130 | 118601 |
| 150 | 118459 |
| 300 | 117600 |
| 500 | 117168 |
| 1000 | 116881 |

WS Clock: Page Faults vs Disk Writes (Frame = 16, Tau = 50)

|  |  |
| --- | --- |
| Refresh Rates | Disk Writes |
| 10 | 14628 |
| 30 | 14403 |
| 50 | 14241 |
| 70 | 14342 |
| 90 | 14411 |
| 110 | 14426 |
| 130 | 14473 |
| 150 | 14467 |
| 300 | 14590 |
| 500 | 14679 |
| 1000 | 14723 |

The correlation between page faults and refresh rates for working set when I control the value of tau is a consistent downward sloping curve. Therefore I think the best refresh rate (refresh per line) for gcc.trace is 1,000,000, which mean there will be no refresh and everything is always inside the working set.

WS Clock: Page Faults vs Tau (Frame = 16, Refresh: 100)

|  |  |
| --- | --- |
| Tau | Page Faults |
| 10 | 125173 |
| 30 | 122946 |
| 50 | 121475 |
| 70 | 120245 |
| 90 | 119568 |
| 110 | 118966 |
| 130 | 118601 |
| 150 | 118459 |
| 200 | 117600 |
| 250 | 117168 |
| 300 | 116881 |

WS Clock: Disk Writes vs Tau (Frame = 16, Refresh: 100)

|  |  |
| --- | --- |
| Tau | Disk Writes |
| 10 | 15528 |
| 30 | 14722 |
| 50 | 14374 |
| 70 | 14219 |
| 90 | 14100 |
| 110 | 14027 |
| 130 | 13972 |
| 150 | 13937 |
| 200 | 13876 |
| 250 | 13837 |
| 300 | 13810 |

The correlation between page faults and the value of tau for working set algorithm shows a distinctive minimal amount of page faults when tau is around 70. Amount of disk writes gradually decreases as tau increases.