Virtual Memory

**Batch:** B.Tech III year

**Instructor:** Rahul Muthu

DA-IICT
The set of pages in a process’s page-frame set is maintained in the order of loading.
FIFO replacement

- The set of pages in a process’s page-frame set is maintained in the order of loading.
- If the page frames allocated to a process are filled then a fault requires moving a page out.
The set of pages in a process’s page-frame set is maintained in the order of loading.

If the page frames allocated to a process are filled then a fault requires moving a page out.

It is not very good, because it fails to take into account the fact that a page which is frequently used is likely to stay in memory longer and hence get moved out.
FIFO replacement

- The set of pages in a process’s page-frame set is maintained in the order of loading.
- If the page frames allocated to a process are filled then a fault requires moving a page out.
- It is not very good, because it fails to take into account the fact that a page which is frequently used is likely to stay in memory longer and hence get moved out.
- FIFO has what is known as Belady’s anomaly, where allocating more page-frames to a process might increase the frequency of page faults.
Special case of stack replacement algorithms.
- Special case of *stack replacement algorithms*.
- No page is ever replaced immediately, until it has been referenced.
- Special case of *stack replacement algorithms*.
- No page is ever replaced immediately, until it has been referenced.
- Allocation of *more* page frames never increases fault frequency.
LRU replacement

- Special case of stack replacement algorithms.
- No page is ever replaced immediately, until it has been referenced.
- Allocation of more page frames never increases fault frequency.
- An approximation version called clock has a referenced bit which it updates periodically to determine pages not recently used.
Allocation policies

- The higher the number of page-frames allocated to a process, the less likely the occurrence of page faults.
Allocation policies

- The higher the number of page-frames allocated to a process, the less likely the occurrence of page faults.

- However, over-allocation in an environment with limited memory leads to the presence of less number of active processes and degree of multiprogramming. This in turn leads to reduced resource utilisation.
Allocation policies

- The higher the number of page-frames allocated to a process, the less likely the occurrence of page faults.

- However, over-allocation in an environment with limited memory leads to the presence of less number of active processes and degree of multiprogramming. This in turn leads to reduced resource utilisation.

- If less than a certain minimum number of page frames are allotted, it could lead to a situation where the process cannot make progress if rollback on instruction interruptions is in place.
The higher the number of page-frames allocated to a process, the less likely the occurrence of page faults.

However, over-allocation in an environment with limited memory leads to the presence of less number of active processes and degree of multiprogramming. This in turn leads to reduced resource utilisation.

If less than a certain minimum number of page frames are allotted, it could lead to a situation when the process cannot make progress if rollback on instruction interruptions is in place.

Processes which are not faulting may have number of page-frames reduced while a high number of page faults may be handled by increasing the number of page frames.
Allocation policies

- The higher the number of page-frames allocated to a process, the less likely the occurrence of page faults.

- However, over-allocation in an environment with limited memory leads to the presence of less number of active processes and degree of multiprogramming. This in turn leads to reduced resource utilisation.

- If less than a certain minimum number of page frames are allotted, it could lead to a situation when the process cannot make progress if rollback on instruction interruptions is in place.

- Processes which are not faulting may have number of page-frames reduced while a high number of page faults may be handled by increasing the number of page frames.

- **Thrashing** is the scenario when most of the active processes are above their upper page fault rate.
This is maintained like many controlled variables.
Page fault frequency

- This is maintained like many controlled variables.
- Its variation is regulated to be bounded below and above.
- This is maintained like many controlled variables.
- Its variation is regulated to be bounded below and above.
- When a process falls below its lower page fault frequency the number of page frames allotted to it may be reduced.
This is maintained like many controlled variables. Its variation is regulated to be bounded below and above. When a process falls below its lower page fault frequency the number of page frames allotted to it may be reduced. When it goes above the upper page fault frequency it may have more page frames allotted.
Page fault frequency

- This is maintained like many controlled variables.
- Its variation is regulated to be bounded below and above.
- When a process falls below its lower page fault frequency the number of page frames allotted to it may be reduced.
- When it goes above the upper page fault frequency it may have more page frames allotted.
- This is called a page fault frequency (PFF) algorithm.
This is maintained like many controlled variables.

Its variation is regulated to be bounded below and above.

When a process falls below its lower page fault frequency the number of page frames allotted to it may be reduced.

When it goes above the upper page fault frequency it may have more page frames allotted.

This is called a page fault frequency (PFF) algorithm.

The parameter $P = 1/T$ is a parameter where $T$ is the critical interpage fault time.
During any interval of time a process favours a subset of its pages.
During any interval of time a process favours a subset of its pages.

The memory reference patterns show a high correlation between recent past and immediate future.
During any interval of time a process favours a subset of its pages.

The memory reference patterns show a high correlation between recent past and immediate future.

The frequency at which a page is referenced is a slowly changing function of time.
During any interval of time a process favours a subset of its pages.

The memory reference patterns show a high correlation between recent past and immediate future.

The frequency at which a page is referenced is a slowly changing function of time.

The working set of a process $W(t, \theta)$ is defined to be the set of pages referenced in the last $\theta$ memory references.
A program should be run if and only if its working set is in memory.
Working set principle

- A program should be run if and only if its working set is in memory.
- A page may not be removed if it is within the working set of a process.