NHN 정리

1. 자료구조 & 알고리즘
   1. 리스트

배열

연결리스트

단일연결

원형연결

이중연결

* 1. 스택 & 큐

스택 :

큐 :

* 1. 트리

??

이진트리

완전이진트리

B+tree

* 1. 그래프
     1. DFS & BFS

1. 프로그래밍 언어
   1. JAVA
   2. C
   3. Python
2. 운영체제
   1. 페이징 알고리즘

보조기억장치를 이용하여 가상 메모리를 같은 크기의 블록으로 나눈것

페이징 기법을 사용하면 연속적이지 않은 공간을 활용할 수 있으므로 메모리 외부 단편화 문제를 해결 할 수 있다.

내부 단편화는 여전히 생길 수 있다.

OPT(최적 교체)

앞으로 가장 오랫동안 사용하지 않을 페이지를 교체하는 기법

각 페이지의 호출 순서와 참조 상황을 미리 예측해야 하므로 실현 가능성이 희박

FIFO(First in First Out)

각페이지가 주 기억장치에 적재될 때 마다 시간을 기억시켜 가장 ㅓㄴ저 들어와서 가장 오래 있던 페이지를 교체하는 기법

이해하기 쉽고 프로그래밍 및 설계가 간단하며 벨레이디의 모순현상이 발생

벨레이디 모순현상 : FIFO알고리즘에서 기존 페이지 프레임의 개수를 늘리면 Page Fault 발생이 감소해야하나 오히려 증가하는 현상

LRU (Least Recently Used) 최근에 가장 오랫동안 사용하지 않은 페이지를 교체하는 기법

각페이지마다 계수기나 스택을 두어 현시점에서 가장 오랫동안 사용하지 않은 페이지를 교체함

LFU(Least Frequently Used)

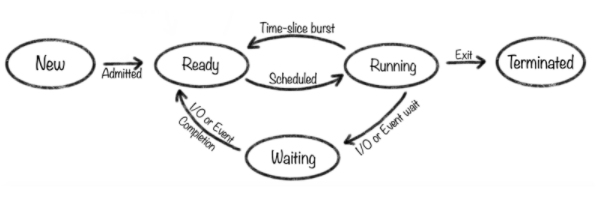
사용빈도가 가장 적은 페이지를 교체하는 기법

단점 : 프로그램 실행 초기에 많이 사용된 페이지가 그 후로 사용되지 않을 경우에도 프레임을 계속 차지하는 단점이 존재

NUR(Not Used Recently)

최근에 사용하지 않은 페이지를 교체하는 기법

SCR(Second Chance ReplaceMent) 가장 오랫동안 주 기억장치에 있던 페이지 중 자주 사용되는 페이지의 교체를 방지하기 위한 기법



스케줄링 : 프로세스가 생성되어 실행될 때 필요한 자원을 해당 프로세스에 할당하는 작업을 말합니다.

성능 측정지표 : 평균실행시간 평균대기시간 평균 반환시간

* 1. 선점 스케줄링
  2. 비선점 스케줄링

이미 할당된 자원을 다른 프로세스가 강탈 할 수 없음

응답시간의 예측이 편하며 일괄처리 방식에 적합

단점 : 덜 중요한 자원을 할등 받으면 중요한 작업이 와도 먼저 처리 될 수 없음

FCFS SJF HRN 우선순위 기한부

FCFS (선입선출)

먼저 들어온 것이 먼저 처리되는 FIFO 구조의 알고리즘

단점으로는 waiting time 이 너무 김

SJF 대기하는 작업중 실행시간이 가장 짧은 작업을 우선하는 하는 알고리즘

평균 대기 시간에 있어서는 최적이 된다.

도착시점을 고려하면 아직 도착하지 않은 프로세스는 스케줄링의 대상이 되지 않기 때문에 더 긴 프로세스가 먼저 할당 될 수 있다.

BURST TIME 이 같을경우 FCFS와 같음

HRN : 어떤 작업이 서비스 받을 시간과 그 작업이 서비스를 기다린 시간으로 결정

우선순위 대기식 = (대기시간 + 서비스 시간) / 서비스 시간

우선순위 : 준비상태 큐에서 대기하는 프로세스에게 부여된 우선순위가 가장 높은 프로세스에게 먼저 CPU를 할당하는 방식

BigEndian 방식 :

낮은 주소에 데이터의 높은 바이트부터 저장하는 방식

메모리에 저장된 순서 그대로 읽을 수 있음

Little endian 방식 : 높은 주소의 데이터에서 낮은 주소로 저장하는 방식

1. 네트워크
2. 데이터베이스