

2020 정보처리기사 필기 - 4.3 응용 SW 기초 기술 활용(2)

2020. 3. 25. 16:44

#2020정보처리기사 #2020정보처리기사필기정리 #2020정처기필기정리 #정보처리기사필기정리 #정처기필기정리

가상 기억장치 구현 기법 / 페이지 교체 알고리즘

가상 기억장치의 개요

- 보조기억장치의 일부를 주기억장치처럼 사용하는 것
- 용량이 작은 주기억장치를 마치 큰 용량을 가진 것처럼 사용할 수 있음
- 프로그램을 여러 작은 블록 단위로 나누어서 가상 기억장치에 보관해 놓고, 프로그램 실행 시 요구되는 블록만 주기억장치에 불연속적으로 할당하여 처리
- 주기억장치의 이용률과 다중 프로그래밍의 효율 상승
- 가상기억장치에 저장된 프로그램을 실행하려면 가상 기억장치의 주소를 주기억장치의 주소로 바꾸는 주소 변환(Mapping) 작업이 필요
- 연속 할당 방식에서 발생할 수 있는 단편화를 해결할 수 있음

페이징 기법

- 가상기억장치의 보관되어 있는 프로그램과 주기억장치의 영역을 동일한 크기로 나눈 후 나뉜 프로그램과 동 일하게 나뉜 주기억장치의 영역에 적재시켜 실행하는 기법
- 프로그램을 일정한 크기로 나눈 크기를 페이지라 하고 페이지 크기로 일정하게 나누어진 주기억장치의 단위를 페이지 프레임이라고 함
- 외부 단편화는 발생하지 않으나 내부 단편화는 발생할 수 있음
- Mapping 작업을 위해서 페이지 맵 테이블이 필요함

세그먼트 기법

- 가상기억장치에 보관되어 있는 프로그램을 다양한 크기의 논리적인 단위로 나눈 후 주기억장치에 적재시켜 실행시키는 방법
 - 프로그램을 논리적인 크기로 나눈 단위를 세그먼트라고 함
 - 내부 단편화는 발생하지 않으나 외부 단편화는 발생할 수 있음
 - Mapping 작업을 위해서 세그먼트 맵 테이블이 필요함
 - 세그먼트가 주기억장치에 적재될 때 다른 세그먼트에게 할당된 영역을 침범할 수 없으며 이를 위해 기억장치
- ≡호기가 필요

가상 메모리

- 고정크기 분할 : 페이지
- 가변크기 분할 : 세그먼트

1D1C 구독하기

ex) 5명이 펜션에 놀러갈 때

- 페이지 : 고정된 크기인 2인실 방을 3개를 주면 1명의 자리가 남게 됨 -> 내부 단편화 발생
- 세그먼트 : 5인실 방을 만들어서 주면 6명의 인원이 놀러올 때 모두가 못 들어가게 됨 -> 외부 단편화 발생

가상 메모리

- 고정크기 분할 : 페이지
- 가변크기 분할 : 세그먼트
- ex) 5명이 펜션에 놀러갈 때
- 페이지 : 고정된 크기인 2인실 방을 3개를 주면 1명의 자리가 남게 됨 -> 내부 단편화 발생
- 세그먼트 : 5인실 방을 만들어서 주면 6명의 인원이 놀러올 때 모두가 못 들어가게 됨 -> 외부 단편화 발생

페이지 교체 알고리즘

- 페이지 부재가 발생했을 때 가상 기억장치의 필요한 페이지를 주기억장치에 적재해야 하는데 이럴 경우 주기억장치의 모든 페이지 프레임이 사용 중이면 어떤 페이지 프레임을 선택하여 교체할 것인지 결정하는 기법
- OPT(OPTimal replacement, 최적 교체)
 - > 앞으로 가장 오랫동안 사용하지 않을 페이지를 교체(예상)
- FIFO(First In First Out)
 - > 페이지가 주기억장치에 적재될 때마다 그때의 시간을 기억시켜 가장 먼저 들어와서 오래 있었던 페이지를 교체
 - > 먼저 들어온 것이 먼저 나감

1. 페이지 2 참조

참조 페이지	2				
페이지 프레임					
부재 발생					

- LRU(Least Recently Used)
 - > 최근에 가장 오랫동안 사용하지 않은 페이지를 교체
 - > 페이지마다 Counter나 Stack을 두어 현시점에서 가장 오래전에 사용된 페이지 교체

1. 페이지 2 참조

1D1C 구독하기

참조 페이지	2				
페이지 프레임					
부재 발생					

- LFU(Least Frequently Used)

-> 사용 빈도가 가장 적은 페이지를 교체

- NUR(Not Used Frequency)

-> 최근에 사용하지 않은 페이지를 교체

-> 최근 사용 여부를 확인하기 위해 페이지마다 참조 비트와 변형 비트를 사용

-> 교체 우선순위

참조 비트	0	0	1	1
변형 비트	0	1	0	1
교체 순서	1	2	3	4

가상 기억장치 기타 관리 사항

페이지 크기에 따른 특징

- 페이지 크기가 작을 경우

-> 단편화와 주기억장치로 이동하는 시간 감소

-> 불필요한 내용이 적재될 확률이 낮아 워킹 셋이 효율적으로 유지

-> Locality에 더욱 일치하여 기억장치에 효율 상승

-> 페이지 맵 테이블 크기가 커지므로 매핑 속도가 늦어짐

-> 디스크 접근 횟수가 많아져 전체적인 입출력 시간이 늘어남

- 페이지 크기가 클 경우

-> 단편화와 주기억장치로 이동하는 시간이 증가

-> 프로세스 수행에 불필요한 내용까지 적재될 수 있음

-> 페이지 맵 테이블 크기가 작아지므로 매핑 속도가 빨라짐

-> 디스크 접근 횟수가 줄어들어 전체적인 입출력 시간이 줄어듦

- 프로세스가 실행되는 동안 주기억장치를 참조할 때 일부 페이지만 집중적으로 참조하는 성질이 있다는 이론
- 스래싱을 방지하기 위한 워킹 셋 이론의 기반
- 프로세스가 집중적으로 사용하는 페이지를 알아내는 방법
- 시간 구역성
 - > 프로세스가 실행되면서 하나의 페이지를 일정 시간 동안 집중적으로 액세스 하는 현상
 - > 시간 구역성이 이루어지는 기억 장소: 반복, 스택, 부프로그램, 1씩 증감, 집계에 사용되는 변수
- 공간 구역성
 - > 프로세스 실행 시 일정 위치의 페이지를 집중적으로 액세스하는 현상
 - > 공간 구역성이 이루어지는 기억 장소: 배열 순회, 순차적 코드의 실행, 프로그래머들이 관련된 변수들을 서로 근처에 선언하여 할당되는 기억 장소

워킹 셋

- 프로세스가 일정 시간 동안 자주 참조하는 페이지들의 집합
- 자주 참조되는 워킹 셋을 주기억장치에 상주시켜 페이지 부재 및 페이지 교체 현상이 줄어들어 프로세스의 기억장치 사용이 안정됨
- 워킹 셋은 시간에 따라 변화

페이지 부재 빈도 방식

- 페이지 부재 빈도는 페이지 부재가 일어나는 횟수
- 페이지 부재율에 따라 주기억장치에 있는 페이지 프레임 수를 조정하여 적정 수준으로 유지하는 방식
- 운영체제는 프로세스 실행 초기에 임의의 페이지 프레임 할당 후 페이지 부재율에 따라 프레임을 할당하거나 회수

프리 페이지징

- 처음의 과도한 페이지 부재를 방지하기 위해 필요할 것 같은 모든 페이지를 한꺼번에 페이지 프레임에 적재하는 기법

스래싱

- 프로세스의 처리 시간보다 페이지 교체에 소요되는 시간이 더 많아지는 현상
- 다중 프로그래밍 시스템이나 가상 기억장치를 사용하는 시스템에서 하나의 프로세스 수행 중 자주 페이지 부재가 발생하면서 나타나는 현상
- 전체 프로세스 선응이 저하됨
- 다중 프로그래밍의 정도가 높아짐에 따라 CPU의 이용률은 어느 특정 시점까지는 높아지지만 다중 프로그래밍의 정도가 더욱 커지면 스래싱이 나타나고 CPU의 이용률은 급격히 감소됨
- 스래싱 현상 방지 방법
 - > 다중 프로그래밍의 정도를 적정 수준으로 유지
 - > 페이지 부재 빈도를 조절하여 사용

≡> 워킹 셋 유지

- > 부족한 자원 증설, 일부 프로세스 중단
- > CPU 성능에 대한 자료의 지속적 관리 및 분석으로 임계치를 예상하여 운영

1D1C 구독하기

프로세스의 개요

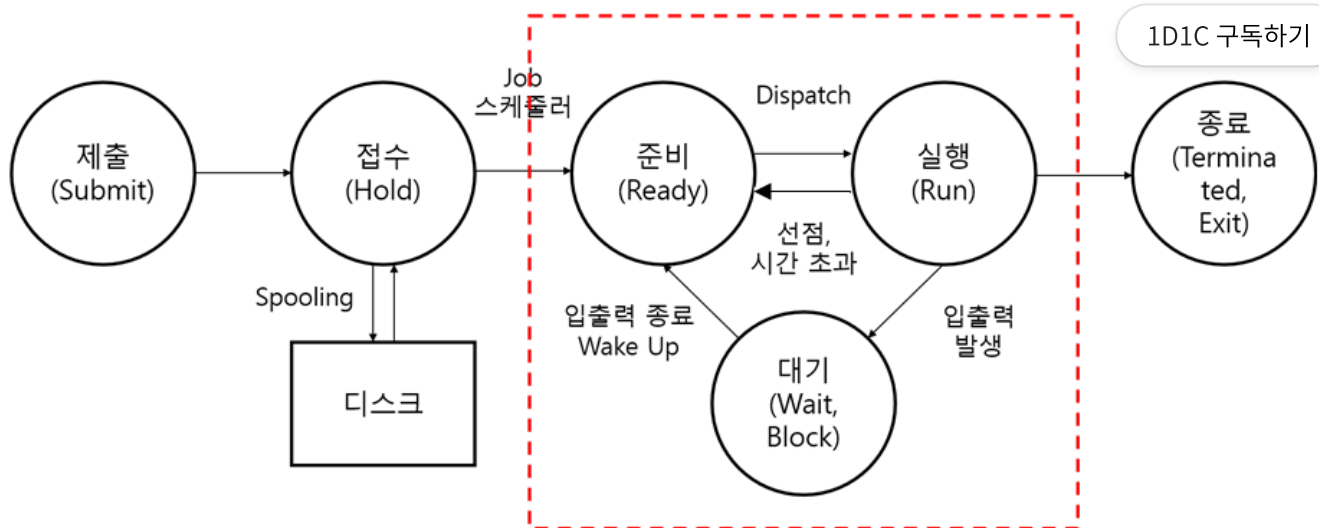
프로세스의 정의

- 프로세서에 의해 처리되는 사용자 프로그램이나 시스템 프로그램을 의미
- 실행 중인 프로그램을 의미하며 작업 혹은 태스크라고도 함
- 프로세스의 또 다른 형태
 - > PCB를 가진 프로그램
 - > 실기억 장치에 저장된 프로그램
 - > 디스패치가 가능한 단위
 - > 프로시저(부 프로그램)가 활동 중인 것
 - > 비동기적 행위(다수의 프로세스가 서로 독립적으로 실행)를 일으키는 주체
 - > 지정된 결과를 얻기 위한 일련의 계통적 동작
 - > 목적 또는 결과에 따라 발생하는 사건들의 과정
 - > 운영체제가 관리하는 실행 단위

PCB(Process Control Block)

- 운영체제가 프로세스에 대한 중요한 정보를 저장해 놓은 곳
- 프로세스가 생성될 때마다 고유의 PCB를 생성하고 프로세스 완료 시 제거됨
- PCB에 저장되는 정보
 - > 프로세스 현재 상태
 - > 포인터
 - > 프로세스 고유 식별자
 - > 스케줄링 및 프로세스의 우선순위
 - > CPU 레지스터 정보
 - > 주기억장치 관리 정보
 - > 입출력 상태 정보
 - > 계정 정보

프로세스 상태 전이



- 제출 : 작업을 처리하기 위해 사용자가 작업을 시스템에 제출한 상태
- 접수 : 제출된 작업이 스푼 공간인 디스크의 할당 위치에 저장한 상태
- 준비
 - > 프로세스가 프로세서를 할당받기 위해 기다리고 있는 상태
 - > 프로세스는 준비상태 큐에서 실행 준비
 - > Job 스케줄러에 의해 수행
- 실행
 - > 준비상태 큐에 있는 프로세스가 프로세서를 할당받아 실행되는 상태
 - > 프로세스 수행이 완료되기 전에 프로세스에게 주어진 할당 시간이 종료되면 프로세스는 준비 상태로 전이
 - > 실행 중인 프로세스에 입출력 처리가 필요하면 실행 중인 프로세스는 대기상태로 전이
 - > CPU 프로세스에 의해 수행
- 대기 : 프로세스에 입출력 처리가 필요하면 현재 실행 중인 프로세스가 중단되고, 입출력 처리가 완료될 때까지 대기하고 있는 상태
- 종료 : 프로세스의 실행이 끝나고 프로세스 할당이 해제된 상태

프로세스 상태 전이 관련 용어

- Dispatch : 준비 상태에 대기하고 있는 프로세스 중 하나가 프로세스를 할당받아 실행 상태로 전이되는 과정
- Wake up : 입출력 작업이 완료되어 프로세스가 대기 상태에서 준비 상태로 전이되는 과정
- Spooling : 입출력 장치의 공유 및 상대적으로 느린 입출력장치의 처리 속도를 보완하기 위해 입출력할 데이터를 직접 장치에 보내지 않고 나중에 한꺼번에 입출력하기 위해 디스크에 저장하는 과정
- 교통량 제어기 : 프로세스의 상태에 대한 조사와 통보를 담당

스레드(Thread)

- 프로세스 내에서의 작업 단위로서 시스템의 여러 자원을 할당받아 실행하는 프로세스 단위
- 프로세스의 일부 특성을 가지고 있어 경량 프로세스라고도 함

≡ 일 프로세스 환경에서 독립적인 다중 수행 가능

- 스레드의 분류

-> 사용자 수준의 스레드 : 사용자가 만든 라이브러리를 사용하여 운용하기 때문에 속도는 빠르다. 1D1C 구독하기

려움

-> 커널 수준의 스레드 : 운영체제의 커널에 의해 스레드를 운영하기 때문에 속도는 느리지만 구현이 쉬움

- 스레드 사용의 장점

-> 하나의 프로세스를 여러 개의 스레드로 생성하여 병행성을 증진할 수 있음

-> H/W, O/S의 성능과 응용 프로그램의 처리율을 향상시킬 수 있음

-> 응용프로그램의 응답 시간을 단축시킬 수 있음

-> 프로세스들 간의 통신이 향상

-> 공통적으로 접근 가능한 기억장치를 통해 효율적으로 통신

필기 정리

2020 정보처리기사 필기 정리

1d1cblog.tistory.com

공감

구독하기

'2020 정보처리기사 > 4과목 : 프로그래밍 언어 활용' 카테고리의 다른 글

2020 정보처리기사 필기 - 4.1 서버 프로그램 구현 (0)	2020.03.28
2020 정보처리기사 필기 - 4.3 응용 SW 기초 기술 활용(4) (0)	2020.03.26
2020 정보처리기사 필기 - 4.3 응용 SW 기초 기술 활용(3) (0)	2020.03.26
2020 정보처리기사 필기 - 4.3 응용 SW 기초 기술 활용(2) (0)	2020.03.25
2020 정보처리기사 필기 - 4.3 응용 SW 기초 기술 활용(1) (0)	2020.03.06

NAME

