

(과목코드 : 063)

성명 :

5. 다음 중 프로세스가 종료되는 경우로 가장 적절하지 않은 것은?
- ① 요구 페이징(demand paging) 하에서 프로세스가 접근한 페이지가 메모리에 부재한 경우
  - ② 프로세스가 프로그램 종료 시스템콜을 호출하는 경우
  - ③ 나눗셈 명령을 실행하는 과정에서 나누는 수가 0인 경우
  - ④ 다른 프로세스의 메모리 영역에 대한 접근을 시도하는 경우

6. 리눅스나 유닉스 계열 시스템에서 파일에 대한 메타데이터를 저장하는 구조인 inode에 대한 설명으로 가장 적절하지 않은 것은?

- ① inode에는 inumber라고 불리는 inode 번호, 파일의 이름, 파일의 소유권, 접근 모드, 각종 파일 접근 시간, 파일을 저장하고 있는 장치 번호 등이 기록되어 있다.
- ② inode에 저장되어 있는 일부 파일 속성들은 stat 시스템콜을 통해 얻을 수 있다.
- ③ 과거의 유닉스 계열 파일시스템에서는 파일 시스템을 생성할 때 inode를 저장하는 공간의 크기가 고정되어 있어서, 이로 인해 파일 시스템이 수용할 수 있는 파일 수에 제약을 받기도 했다.
- ④ 사용자가 파일의 inode 번호를 확인하려면 `ls -li` 명령을 사용하면 된다.

7. 페이징 시스템에서 프로세스에게 세 개의 페이지 프레임이 할당되어 있고 프로세스의 페이지 참조 열이 2 3 2 1 5 2 4 5 3 2 5 2 라고 할 때, 다음 중 페이지 폴트(page fault) 발생 횟수가 많은 순으로 페이지 교체 기법을 나열한 것은? (단, 최초의 페이지 프레임은 모두 비어있다고 가정한다.)

- ① FIFO > LRU > Optimal
- ② FIFO > Optimal > LRU
- ③ Optimal > LRU > FIFO
- ④ Optimal > FIFO > LRU

8. 페이징 시스템에서 페이지 크기에 대한 설명으로 가장 적절하지 않은 것은?

- ① 페이지 크기가 작을수록 페이지 테이블을 저장하는데 많은 공간이 필요하게 된다.
- ② 페이지 크기가 클수록 디스크 입출력 효율은 개선된다.
- ③ 페이지 크기가 작을수록 외부 단편화가 줄어든다.
- ④ 페이지 크기가 클수록 실제 사용되지 않는 CPU 명령이나 데이터가 메모리에 적재될 수 있다.

9. 다음 중 스레드에 대한 설명으로 가장 적절하지 않은 것은?

- ① 다수의 사용자 수준 스레드를 그보다 적은 수의 커널 스레드와 연관시켜 멀티플렉스하는 스레드 모델을 다대다(many-to-many) 모델이라고 한다.
- ② Pthreads 라이브러리를 사용하는 경우, 프로세스 내의 한 스레드가 전역 변수의 값을 변경하면 다른 스레드가 이를 접근할 때 변경된 결과를 얻는다.
- ③ 사용자 수준에서 구현된 스레드는 생성 시 커널의 시스템 호출 서비스를 받지 않으므로 속도가 빠르다.
- ④ 한 프로세스 내의 여러 스레드들 간에 스택과 같은 실행 환경을 공유하여 메모리의 낭비를 줄인다.

10. 레디 큐에 세 프로세스가 P1, P2, P3 순으로 대기하고 있다. 이들을 스케줄링하기 위해 시간 할당량(time quantum)의 크기를 달리해가며 라운드 로빈(Round Robin) 스케줄링을 적용하는 경우, 모든 프로세스가 실행을 마칠 때까지 발생하는 문맥 전환(context switch) 횟수가 나머지 셋과 다른 하나는 무엇인가?

프로세스	CPU 버스트
P1	18
P2	4
P3	5

- ① 시간 할당량이 4 ms인 경우
- ② 시간 할당량이 5 ms인 경우
- ③ 시간 할당량이 6 ms인 경우
- ④ 시간 할당량이 7 ms인 경우

11. 다음 중 메모리 관리 기법인 스와핑(swapping)과 가장 밀접한 관련이 있는 스케줄러는?

- ① 단기(short-term) 스케줄러
- ② 중기(medium-term) 스케줄러
- ③ 장기(long-term) 스케줄러
- ④ CPU 스케줄러

12. fork()를 사용하여 프로세스 A가 사용하던 공간을 복사하여 프로세스 B가 사용할 수 있는 공간을 만들었다. 그러나 실제로 물리 메모리는 두 개의 프로세스에서 공유되는 상태로 유지되다가 프로세스 B가 사용하는 데이터에 수정이 일어났을 때 프로세스 B의 데이터 영역에 대한 페이지가 새로 확보된다. 이런 방식을 무엇이라 하는가?
- ① 요구 페이징(demand paging)
  - ② 역 페이지 테이블(invert page table)
  - ③ 다단계 페이지 테이블(multi-level page table)
  - ④ 쓰기 시점 복사(copy on write)

13. 다음은 고전적인 동기화 문제 중 하나인 readers-writers 문제에 대한 해결책을 표현한 의사코드(pseudo code)이다. 이 해결책에 대한 설명으로 가장 적절하지 않은 것은?

<pre> /* writer */ do {     wait (rw_mutex);     ...     /* WRITING */     ...     signal(rw_mutex); } while (true); </pre>	<pre> /* reader */ do {     wait(mutex);     read_count++;     if (read_count == 1)         wait(rw_mutex);     signal(mutex);     ...     /* READING */     ...     wait(mutex);     read_count--;     if (read_count == 0)         signal(rw_mutex);     signal(mutex); } while (true); </pre>
---	--

- ① 세마포어 rw\_mutex는 reader 프로세스와 writer 프로세스 간 공유 데이터 접근 시의 상호 배제를 위해 사용된다.
- ② 세마포어 rw\_mutex는 writer 프로세스와 또 다른 writer 프로세스 간 공유 데이터 접근 시의 상호 배제를 위해 사용된다.
- ③ 세마포어 rw\_mutex는 변수 read\_count 값의 갱신 시의 상호 배제를 위해 사용된다.
- ④ 이 해결책은 기아(starvation) 현상의 가능성이 있다.

14. 프로세스 제어 블록에 있는 정보와 가장 거리가 먼 것은?

- ① 각 프로세스를 구분하는 구분자(ID)
- ② 실행하려는 프로세스가 메모리의 어디에 저장되어 있는지에 대한 정보
- ③ 프로세스가 사용하는 파일에 대한 소유자와 접근 권한의 정보
- ④ 프로그램 카운터와 같은 프로세스가 사용했던 중간값

15. 유닉스 프로세스 관리에 대한 설명으로 가장 적절하지 않은 것은?

- ① swapper 프로세스는 전체 프로세스의 출발점이 되고 나머지 프로세스는 swapper의 자식으로 만든다.
- ② 모든 프로세스를 부모-자식 관계로 만들어서 자식 프로세스가 작업을 마쳤을 때 사용하던 자원을 부모 프로세스가 회수한다.
- ③ 부모 프로세스가 자식 프로세스보다 먼저 종료 되면 자식 프로세스는 고아가 되고 반환되지 못한 자원으로 시스템의 자원이 낭비된다.
- ④ wait() 함수를 사용하면 부모 프로세스는 자식 프로세스가 끝날 때까지 다음 명령을 처리하지 않는다.

16. 캐시 매핑에 대한 설명으로 가장 적절하지 않은 것은?

- ① 캐시 직접 매핑은 메모리 블록이 캐시의 정해진 위치에만 들어갈 수 있기 때문에 자리다툼이 발생하지 않는다.
- ② 캐시 연관 매핑은 캐시 히트인지 캐시 미스인지 확인하기 위해 캐시의 모든 주소를 검색해야 한다.
- ③ 캐시 집합-연관 매핑은 캐시 직접 매핑과 캐시 연관 매핑의 장점만 취한 방식이다.
- ④ 캐시 태그 메모리의 크기는 연관 매핑이 직접 매핑보다 더 크다.

17. 다음 설명에 해당하는 디스크 스케줄링 기법으로 가장 적절한 것은?

헤드가 한 방향으로만 움직이면서 서비스를 한다. 더 이상 서비스할 트랙이 없으면 헤드가 끝까지 가지 않고 중간에서 방향을 바꾼다. 중간 쪽에 있는 트랙은 안쪽으로 들어가면서 한 번, 바깥쪽으로 나오면서 한 번 방문하기 때문에 바깥쪽 트랙이 상대적으로 불이익을 받는다.

- ① FCFS(First Come First Serve)
- ② 블록 SSTF(Block Shortest Seek Time First)
- ③ C-SCAN(Circular SCAN)
- ④ LOOK

18. 파일 구조에 대한 설명으로 가장 적절하지 않은 것은?

- ① 순차 파일(sequential file)은 초기 파일 시스템에서 사용하던 구조이고 구조가 단순하나 삽입이나 삭제에 시간이 많이 걸린다.
- ② 인덱스 파일(index file)은 데이터 접근이 빠르지만 모든 주소에 일정량의 데이터가 고르게 저장되지 않고 빈 공간이 생겨서 공간 낭비가 생긴다.
- ③ 직접 파일(direct file)은 해시 함수(hash function)를 이용해서 키 값을 찾는다.
- ④ 인덱스 파일 구조는 순차 접근과 직접 접근이 모두 가능하다. 순차 파일 구조로 파일을 저장하고, 인덱스 테이블을 보고 원하는 파일에 직접 접근한다.

19. 다음은 인터럽트 발생 직후 이루어지는 동작들이다. 이들 중 순서상 세 번째에 해당하는 것은?

- ① 앞서 저장된 상태를 복원하고 하던 일을 이어나간다.
- ② 운영체제의 인터럽트 처리 함수를 실행한다.
- ③ CPU는 하던 일을 즉시 중지한다.
- ④ CPU는 현재 상태를 저장한다.

20. 네트워크 저장장치에 대한 설명으로 가장 적절하지 않은 것은?

- ① 기존 저장장치를 LAN이나 WAN에 붙여서 사용하는 방식을 NAS라 한다.
- ② 서버와 같은 컴퓨터에 직접 연결된 저장장치를 DAS라고 한다.
- ③ DAS는 운영체제와 상관없이 사용할 수 있으며, 데이터 공유, 백업, 보안 등이 서버를 통해 자동으로 이루어지므로 사용자가 특별히 신경 쓸 일이 없다.
- ④ 데이터 서버, 백업 서버, RAID 등의 장치를 네트워크로 묶고 데이터 접근을 위한 서버를 두는 방식을 SAN이라 한다.

21. 분산 시스템의 구조에 대한 설명으로 가장 적절한 것은?

- ① 약결합 시스템(loosely coupled system)은 네트워크로 연결된 모든 컴퓨터의 프로세서가 하나의 메모리를 공유하는 방식이다.
- ② 클라이언트/서버 환경에서 서버의 과부하를 해결하기 위해서 만들어진 시스템이 P2P 시스템이다.
- ③ 언제 어디서나 컴퓨팅 파워나 소프트웨어에 접근할 수 있는 유연한 컴퓨팅 환경을 가리키는 용어는 사물 인터넷(IoT)이다.
- ④ 여러 곳에 떨어져 있는 컴퓨팅 파워나 소프트웨어를 하나로 묶어 하나의 컴퓨터처럼 사용하는 기술이 클라우드 컴퓨팅이다.

