**서술형 과제 (2주차)**

**1. 다음과 같이 스케쥴링 정책이 적용되어 있는 General Purpose 운영체제가 있습니다. 다음 정책이 실제 어떻게 동작할지를 가능한 상세히 기술해주세요.**

* Round Robin 스케쥴링 정책
  + 100ms 마다 다음 프로세스로 교체
  + Ready Queue, Running Queue, Block Queue 존재
  + 각 Queue 는 FIFO 정책으로 동작함
* 인터럽트로 선점 가능 (선점형 스케쥴링 기능 지원)
* 타이머 인터럽트를 지원하며, 타이머 칩에서는 1ms 마다 인터럽트를 발생시킴
* CPU만 실행하는 A, B, C 프로세스가 Ready Queue 에 기술한 순서대로 들어간 상태임
  + 각 프로세스가 총 실행해야 하는 시간은 다음과 같음
  + A 는 200ms, B 는 500ms, C 는 300ms

➡ 위에서 기술한 사항 외에 추가 가정이 꼭 필요한 경우에는 각자 가정한 상황을 기술하고, 동작 방식을 기술하세요.

처음 상태는

Ready Queue ABC

Running Queue

Block Queue

이렇게 들어있다.

ready queue 상태에 따라 A 가 100ms 실행되고 인터럽트가 발생되어 B가 실행되며

ready queue 제일 뒤로 들어간다.

이 때 상태는 ready queue C A

running queue B 이다. 이러한 방식으로 100ms 마다 각 프로세스는 교체 되며 최종 적으로

CPU에서 실행된 결과는

ABCABCBCBB 이다.

(A종료)

(C종료)

**2. 다음과 같이 메모리 정책이 적용되어 있는 General Purpose 운영체제가 있습니다. 다음 정책이 실제 내부적으로 어떻게 동작할지를 가능한 상세히 기술해주세요.**

* 페이징 시스템을 기반으로 한 가상 메모리 시스템 지원
* 요구 페이징을 지원하며, 하드웨어에서는 MMU와 TLB 칩을 지원함

➡ 위에서 기술한 상황에서 특정 프로세스가 실행 후, CPU 상에서 가상 주소로 메모리에서 데이터를 가져와야 할때, 이 동작이 어떻게 진행될지를 가능한 상세히 기술해주세요

CPU가 **처음** 가상 메모리를 요청하면 MMU가 CR3 레지스터에 값을 가지고 Page Table에 들어간다.

물리 주소를 가져오면 MMU가 물리주소에 접근한다. 그러면 CPU에 가져온 물리 주소를 반환한다.

한 번 접근해서 가상주소를 입력해서 반환된 물리주소는 TLB에 저장된게 된다.

**다시** 똑같은 가상 메모리를 요청하게 되면 MMU를 통하는게 아니라 TLB에 해당 물리주소가 있다면 처음에 이루어졌던 작업은 하지 않고 물리주소를 반환한다.

**3. 다음과 같은 운영체제가 설치된 컴퓨터가 있을 때, 사용자가 컴퓨터를 켰을 때부터, 운영체제가 프로세스를 실행하고 쉘을 실행할 때까지 어떻게 동작할지를 가능한 상세히 기술해주세요.**

* 컴퓨터는 BIOS를 지원하고, 부팅을 지원하는 bootstrap loader 가 별도로 설치되어 있음
* 운영체제 커널은 실행후, 최초 프로세스(init)를 운영체제 코드상에서 바로 실행시키며,
* 이후에는 fork() 기능을 지원하며, 쉘 프로그램을 실행시킴
* 쉘 프로그램을 통해 사용자는 각 프로그램을 실행시킬 수 있음

➡ 위에서 기술한 사항 외에 추가 가정이 꼭 필요한 경우에는 각자 가정한 상황을 기술하고, 동작 방식을 기술하세요.

CPU는 ROM에 있는 BIOS가 있는 주소(FFFF0H)로 이동을 한다. BIOS 일부 실행이 시작되는데 BIOS프로그램을 RAM에 올리는 역할을 한다. 메모리에 올라간 프로그램은 하드웨어를 초기화하고 MBR이라는 곳으로 들어가서 boostrap loader를 메모리에 올린다.

그러면 부트로더는 파티션 테이블의 정보를 가지고 메인 파티션을 찾아낸다.

부트로더는 메인 파티션에 부트섹터로 가고 부트 코드를 읽는다.

부트 코더는 커널 이미지가 있는 곳에서 운영체제 실행 파일을 읽어낸다. 그런 다음 운영체제로 옮기면 우리가 보는 리눅스화면이 나오게 된다.

이때 fork를 이용하면 사용자별로 리눅스를 제공할 수 있고 쉘프로그램을 실행하여 사용자는 각 프로그램을 이용할 수 있다.

**4. 다음과 같은 General Purpose 운영체제에서, 사용자가 커멘드창을 오픈한 후, ls (dir) 커멘드를 키보드로 작성할 때, 실제 내부적으로 어떻게 동작하는지 가능한 상세히 기술해주세요**

* 사용자가 커멘드창을 오픈하면, 자동으로 쉘프로그램이 실행됨
* ls 명령은 윈도우의 dir 과 마찬가지로 해당 디렉토리의 정보를 보여주는 프로그램임
* 해당 쉘에서는 키보드를 입력하면, 해당 키보드 문자가 화면에 표시하며, 엔터를 누르면 그동안 입력받은 문자열을 명령으로 인식하고, 해당 명령을 실행함
* 스케쥴링 방식은 선점형을 지원하며, 기본적으로 멀티 태스킹을 지원함
* 인터럽트를 지원하며, 키보드 인터럽트와 타이머 인터럽트를 지원함
* 타이머 인터럽트는 1ms 마다 발생함
* 사용자는 키보드로 l 과 s 그리고 엔터키를 누를 때 각기 인터럽트가 발생함
* ls 프로그램이 실행되면, 내부적으로 필요시 시스템콜을 호출할 수 있음

➡ 위에서 기술한 사항 외에 추가 가정이 꼭 필요한 경우에는 각자 가정한 상황을 기술하고, 동작 방식을 기술하세요.

사용자가 커맨드창을 오픈하여 쉘프로그램을 실행한다. 쉘프로그램이 실행되는 중 l 과 s 를 누르면 인터럽트가 발생하여 순간적으로 입력 프로세스가 진행되게 한다.

ls는 위와 같은 방식으로 입력한 후 enter 를 눌렀을 때 쉘 프로그램은 ls 가 있는 위치를 찾게 된다.

/ 경로에서 ls 명령어가 있는지 찾아보게 되며 ls 위치로 이동하여 프로그램을 실행되면 해당 디렉토리의 정보를 보여주게 된다.