연습문제 7번 8번 발표를 시작하겠습니다

7번 문제는 1부터 10까지의 합을 종료코드로 리턴하는 sum.c 프로그램을 자식 프로세스에서 실행시키고 부모 프로세스에서 값을 받아 출력하는 프로그램입니다.

(클릭). 프랙3\_7 프로그램의 플로우 차트입니다(클릭). 프로그램이 실행되면 fork() 함수를 실행하여 자식 프로세스를 생성하고 부모 프로세스와 동시에 작동합니다(클릭). 부모 프로세스의 PID는 0보다 크기 때문에 wait() 함수를 실행하여 자식 프로세스가 종료될 때 까지 기다립니다(클릭) (클릭). 자식 프로세스의 PID는 0이므로(클릭) sum.c 프로그램을 실행하여 1부터 10까지의 합을 종료코드로 받은 후 종료됩니다(클릭)(클릭). 부모 프로세스는 55의 종료코드를 받아 화면에 출력합니다.(클릭)(클릭)

화면에 코드는 prac3\_7.c 의 코드와 sum.c 의 코드입니다. 프로세스 아이디를 나타내는 pid 변수와 자식 프로세스의 종료코드를 받기 위한 status 변수를 선언 후 fork() 함수로 자식 프로세스를 생성합니다. 자식 프로세스는 execlp() 함수를 통해 미리 작성된 sum.c 프로그램을 실행하여 리턴 값을 부모 프로세스에게 전달하며 종료됩니다. 부모 프로세스는 종료코드를 받아 화면에 출력하며 종료됩니다.

8번 문제는 부모 프로세스의 n과 m 변수를 자식 프로세스에 전달하여 n부터 m 까지의 합을 계산하고 부모 프로세스에게 전달하여 1000보다 큰지 작은지를 판별하여 출력하는 프로그램입니다.

(클릭)프랙3\_8.c프로그램의 플로우 차트입니다. (클릭)n은 10, m은 100 두 변수를 선언하고 fork() 함수를 사용하여 자식 프로세스를 생성합니다(클릭). 부모 프로세스가 자식 프로세스의 종료를 기다리고(클릭) (클릭), 자식 프로세스의 PID가 0이므로(클릭) 가우스 공식을 사용하여 n부터 m까지의 합을 구하는 과정을 실행하며(클릭) (클릭) 1000 보다 작을 경우 0, 1000이면 1, 1000보다 클 경우 2를 종료코드로 리턴하여 부모 프로세스에 전달합니다(클릭) (클릭). 10부터 100까지의 합은 1000보다 큼을 보여주며 프로그램은 종료됩니다.

화면에 코드는 prac3\_8.c 의 코드입니다. 프로세스 아이디를 나타내는 pid 변수와 n 과 m, 합을 받기 위한 변수 sum, 자식 프로세스의 종료코드를 받기 위한 status 변수를 선언 후 fork() 함수를 사용하여 자식 프로세스를 생성합니다.

자식 프로세스는 n부터 m까지의 합을 구하여 1000보다 큰지, 같은지, 작은지 에 따라 종료코드를 부모 프로세스에게 전달하고 부모 프로세스는 종료코드에 따라 각 해당되는 내용을 출력 후 프로그램이 종료됩니다.

이상으로 발표를 마치겠습니다.