**서술형 과제 (4주차)**

1. **fork() 함수와 달리 exec() 함수는 다양한 기능들을 제공하는 exec 계열 함수들이 존재합니다. execl, execlp, execle, execv, execv, execvp, execvpe 함수들의 기능을 설명하고 각 함수들의 차이점을 표로 정리하세요.**
   * exec의 suffix로 붙는 문자의 의미를 반드시 설명해 주세요.

l과 v는 argv인자를 넘겨줄 때 사용한다. l일 경우는 char \* 하나 v일 경우에는 char \*[]배열로 넘겨준다.

e는 환경변수를 넘겨줄 때 사용한다. char \*[]배열로 넘겨줌

p는 환경변수 PATH를 참조하기 때문에 절대경로를 입력하지 않아도 된다.

|  |  |  |  |
| --- | --- | --- | --- |
|  | l 또는 v | e | p |
| execl | l방식 | x | x |
| execlp | l방식 | x | o |
| execle | l방식 | o | x |
| execv | v방식 | x | x |
| execvp | v방식 | x | o |
| execvpe | v방식 | o | o |

1. **프로세스간 커뮤니케이션 (Inter-process Communication) 기법 중 Pipe, Shared Memory를 포함해 3가지 이상 기법을 조사하고 각 기법들의 제약 조건, 사용 방식의 차이점에 대해 작성해주세요.**

Pipe : 기본 파이프는 단방향 통신으로 입력이나 출력 한가지만 가능하다.

메시지 큐 : FIFO 정책으로 데이터를 전송한다. 일부가 실패하여도 전체가 영향을 받지 않는다.

공유 메모리 : 노골적으로 kernel space에 메모리 공간을 만드록, 해당 공간을 변수처럼 사용한다. 공유 메모리 key를 가지고 여러 프로세스가 접근이 가능하다.

1. **Fork를 이용한 프로세스 생성시 메모리 복사 비용을 줄이기 위한 기법 중 하나로 copy-on-write 기법이 있습니다. Copy-on-write 기법이 있을 때와 없을 때 두 경우에 대해서 프로세스 A, B가 사용하고 있는 메모리 공간에 Read와 Write 수행하면 생기는 변화를 각 Process의 메모리 공간, 각 Process의 Page Table, 그리고 Physical Memory Space를 사용해 그림으로 나타내세요.**
   * 각 케이스에 대해 동작 순서를 표시하는 번호와 설명이 필요합니다.
   * 총 네가지 경우에 대해 설명해야합니다.
     1. Copy-on-write가 없이 Process A가 Process B를 fork로 생성하고 B가 Write 한 경우.
     2. Copy-on-write가 없이 Process A가 Process B를 fork로 생성하고 B가 Read 한 경우.
     3. Copy-on-write가 있는 상황에서 Process A가 Process B를 fork로 생성하고 B가 Write 한 경우.
     4. Copy-on-write가 있는 상황에서 Process A가 Process B를 fork로 생성하고 B가 Read 한 경우.

1번과 2번에 경우 fork로 생성시 Process A 를 B로 write와 read에 상관없이 완전히 복사하게 된다.



4번의 경우 fork 로 read 한 경우 똑같은 물리메모리를 보기만하고 복사만 되지는 않는다. 이 상태에서 3번의 write를 한 경우 해당하는 부분만 따로 복사하여 수정이 진행된다.



1. **파일에 접근하는 방법으로 read, write 함수들을 사용하는 방식과 mmap을 사용하는 Memory Mapped File 방식이 있습니다. 두 방식의 차이점과 장단점을 접근 방식, 사용성, OS 내부에서 처리 방식 관점에서 서술 하십시오.**

mmap은 가상메모리가 작동하는 방식과 비슷하다. mmap실행시 가상 메모리 주소에 file 주소를 매핑하고 메모리 접근시 물리 데이터가 없다면 페이지 폴트 인터럽트를 발생하여 data를 복사하여 넣어준다. read시 물리 페이지에 데이터를 읽으면 되며 write시에는 물리 데이터 수정 후 페이지 상태 flag를 변경해준다. close시 물리 페이지 데이터가 file에 업데이트된다.

장점으로는 read(), write()에 비하여 반복적인 파일 접근을 방지하여 성능이 개선되고 mapping된 영역은 간단한 포인터 조작으로 탐색 가능하다는 것이다.

단점으로는 mmap은 페이지 사이즈 단위로 매핑되기 때문에 작은 공간을 사용하더라도 남은 공간을 0으로 채워줘야하기 때문에 메모리가 낭비되는점을 들 수 있다.