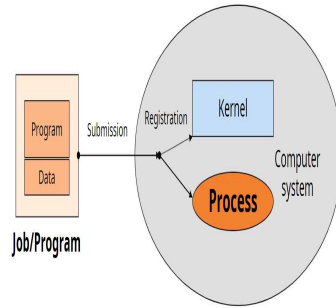
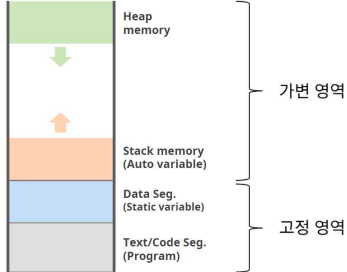


6강 Process information 요약 (System Programming, Koreatech), 8조 이상민, 배수빈, 이지영

6-1 Program vs Process				Process group			
Program (프로그램)	• 실행할 프로그램+데이터 • 컴퓨터 시스템에 실행 요청전 상태			• 하나의 job 수행을 목적으로 하는 관련된 process들을 묶은 것 ex) Shell script 내부의 명령어들 부모와 자식 프로세스 등	Process group leader		
Process (프로세스)	• Running Program • 실행을 위해 시스템(커널)에 등록된 작업 • 시스템 성능향상을 위해 커널에 의해 관리 됨			• Process group에 전달 된 signal은 그룹 내 모든 프로세스에게 전달 • job 단위 프로세스 제어를 위해 활용	• Process group에 속한 process 중 하나		
					Process Group ID (PGID)		
					• Process group에 부여된 고유번호 • PGID = group leader의 PID		
Process in memory				Getting / Setting process group ID			
가 변 영 역	Heap memory	동적 할당 시 Heap memory 사용		// pgid 가져오기 pid_t getpid (pid_t pid); pid_t getpgid (void);	pid	Group ID를 얻으려는 process의 ID 0 : current process	
	Stack memory	local variable 선언 시 stack memory 사용			Return	Process group ID, -1 : error	
고 정 영 역	Data Seg	프로그램을 실행하기 위한 데이터 미리 정의된 고정된 데이터 보관			// 프로세스 그룹 변경하기 int setpgid (pid_t pid, pid_t pgid);		• pid process를 pgid 그룹으로 옮김 • 그룹 이동은 같은 session내 group 사이만 가능 pid=0 : current process의미 pgid=0 : pgid를 pid로 설정
	Text/Code Seg	(Program) 코드 저장 영역			Session		
Shell commands for process				하나의 foreground process group과 0개 이상의 background process group 가짐			
ps	현재 실행중인 process 정보 확인			foreground process group	• 사용자의 입력을 받는 process group • Terminal을 통해 전달되는 signal을 전달 받음		
	주요 옵션	-u userid	특정 사용자의 process 확인		background process group	• foreground process group외의 process group • Output은 terminal에 출력 됨	
			j	job control format으로 출력 PID, PGID, SID확인 가능			
top	현재 시스템 내 process 정보 실시간 확인			Session Leader	세션 리더, login shell process		
kill -signal pid	process에게 signal 전달			Session ID	session leader의 PID		
	-signal	옵션, 어떤 시그널을 전달할 것인지 주요시그널 : 9, SIGKILL (프로세서 강제종료)		Getting / Setting session ID			
	pid	어떤 프로세서에게 전달할 것인지 pid 이용		pid_t getsid (pid_t pid);	session의 id를 얻어오는 system call		
6-2 프로세스 계층				pid_t setsid (void);			
PID (Process IDs)	• Process에 부여된 식별 번호 • 시스템 내 유일한 번호 할당		Parent Process ID = PPID	• 새로운 session 생성 setsid를 호출한 process가 현재 session의 리더가 아닐 시, 새로운 session을 생성하고 생성된 session의 리더가 됨 • 생성된 session은 controlling terminal을 갖지 않음			
Parent process (부모프로세스)	• 자신을 생성한 프로세스 • 모든 프로세스는 부모 프로세스가 있음 • 최상위 process = kernel (pid=0)						
pid_t getpid (void)	pid 얻어오기	pid_t getppid (void)	ppid 얻어오기				

6-3 프로세스 실행시간						
Running time of a process			Getting process running time			
Process running time	// 프로세스가 총 수행한 시간 system running time + user running time		clock_t times (struct tms * buf)		buf	Running time 정보를 저장할 tms 구조체의 포인터
System running time	• Kernel code를 수행한 시간 • System call에 소요된 시간				return	특정 시점(e.g.부팅 이후)부터 경과한 시간 - clock tick단위(software clock) - 두 시점의 return 값의 차를 이용 해 수행시간 측정 가능 - -1 : error
User running time	• 사용자 모드에서 수행한 시간 • 사용자 작성 코드를 실행하는데 걸린 시간		tms 구조체	clock_t tms_utime;	user time, 사용자 시간	
실행시간 측정 및 사용 시간에 따른 요금 부과등에 활용 가능				clock_t tms_stime;	system time, 시스템 시간	
			clock_t tms_cutime;	user time of children		
				clock_t tms_cstime;	system time of children	