프로젝트명	Unix Shell program				
개발 인원	1명	본인 역할	일인 프로젝트	개발 기간	2016.11
개발 언어	C				
개발툴	VIM				
작품 소개(요약)	“유닉스 프로그램”이라는 강의를 수강하면서 여러 가지 shell에 대한 프로그래밍과 지식을 배웠습니다. 이에 수강했던 내용들을 복습하며 공부하고자 unix에서 이요 가능한 Shell program을 직접 구현해 보았습니다. 가능한 기능은 cd(cd path) : 현재 디렉토리 변경 killone(killone process_name) : 특정 프로세스 강제 종료 pipe(arg ... | arg ...) : 두 개의 명령을 pipe를 통해서 연결 background process(arg ... &) : 명령을 background로 실행 입니다.				
개발 내용	Shell을 구현하면서 가장 초점을 맞췄던 것은 background process의 관리였습니다. 일반 배열이 아닌 원형큐를 통해서 구현하였습니다. 이러한 구조를 사용하기 쉽고 관리하기 쉽게 하기 위해서 C언어에서의 객체지향프로그래밍을 통해서 자료구조를 형성하였습니다. 이 덕분에 많은 background process를 효율적으로 관리할 수 있고, 추가하기도 쉽습니다. 하지만 원형큐로 구현이 되어져 있기 때문에 프로세스를 큐에서 삭제하는 연산이 O(N)시간이 걸리는 단점이 있습니다.(여기서 N은 원형큐의 크기) 또한 background process를 자동으로 종료하기 위해서 SIGCHLD에 대한 handler를 등록하여 자동으로 zombie process를 종료시키도록 하였습니다.				
느낀점 / 아쉬운 점	평소에 써오던 C++이 아닌 C를 통해서 구현하였고, 또한 Visual Studio가 아닌 VIM을 통해서 구현하다보니 구현하는데 걸리는 시간이 훨씬 오래 걸렸습니다. 하지만 이 기회를 통해서 C언어에서의 객체지향프로그래밍에 대해서 공부 할 수 있었고 VIM에서의 구현도 꽤나 익숙해졌습니다. 특히나 Unix에서의 manual page를 보는 것이 익숙해지고 자연스러워졌습니다. 하지만 아쉬운 점은 오류처리가 미흡하다는 점입니다. 최소한의 오류에 대해서는 처리를 해주지만 이상적이지 않은 사용자의 입력에 대해서는 예상치 못하게 코드가 흘러갈 수 있습니다.				
참고 자료	shell.c (구현 코드)				