

* 이 문서의 내용은 C++ 프로그램 기준이고, Python과 Qt의 경우에도 비슷한 기준을 적용한다.

1. 제출 기한

- 제출 기한은 프로그래밍 과제 문서에 명시되어 있다.
- 제출 기한은 일반적으로 연장되지 않으니, 과제를 일찍 시작하도록 한다.

2. 제출물

- 보고서와 프로그램을 제출한다.
- 보고서
 - 중요한 내용을 중심으로 간결하게 작성하고, 프로그램은 첨부하지 않는다.
 - 보고서는 PDF 파일로 만들어서 제출한다.
 - 기본적으로 아래의 내용들을 포함하여야 한다. 과제에 따라서, 내용을 추가하거나 삭제할 수 있다.
 - 표지: 과제 번호/제목, 명예서약, 프로그램 제작자의 학과, 학번, 이름, povis id 등을 적는다.
 - **명예서약(Honor code):** 표지에 다음의 내용을 포함한다. "나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다." 보고서 표지에 명예서약이 없는 경우는 과제를 제출하지 않은 것으로 처리한다.
 - 프로그램 기능에 대한 개요
 - 프로그램의 class 구성 및 전체 구조와 알고리즘 등
 - 기타 프로그램을 이해하는데 필요한 내용
 - 프로그램 실행 방법: 예를 들어, 메뉴가 있다면 메뉴 사용 방법, data file이 필요하다면 data file을 읽어 들이는 방법을 설명한다.
 - 예제: 프로그램의 실행 결과 몇 개를 dump 받아 보고서에 포함시키고 간단히 설명한다.
 - 토론: 프로그램 개발 과정에서 생겼던 여러 가지 문제 또는 결정 사항에 대한 해결 방법 및 결정 과정에 대하여 설명한다.
 - 결론: 이번 과제를 통하여 배우거나 깨달은 내용들을 정리한다.
 - 개선 방향: 제작된 프로그램에 추가할 수 있는 기능, 실행 속도를 높이는 방법 등 프로그램의 성능을 높일 수 있는 방향에 대하여 설명한다.
 - 참고문헌: 인터넷 등의 다른 소스에서 참고한 내용이 있으면 출처와 내용을 명확히 기술. 숙제를 위하여 다른 소스를 참고하고 사용하는 것은 좋으나, 그 경우에 반드시 출처를 명시해야 한다. 출처가 명시되어 있지 않은 경우는 부정행위로 간주될 수 있다.
- 프로그램
 - MinGW 환경에서 g++을 이용하여 컴파일되고, 실행되어야 한다. 과목에 사용되는 MinGW 환경은 공지된 버전을 기준으로 한다.

- 프로그램 소스 코드 파일들(*.cpp, *.h)을 포함한 프로그램 디렉토리를 제출한다. 파일들을 하나의 zip 파일로 묶어서 제출한다.
- 프로그램의 구성요소들을 체계적으로 분류하여 여러 개의 파일로 나누어 작성하고, 반드시 Makefile을 만들어서 make를 이용하여 컴파일할 수 있어야 한다.
- ReadMe 파일을 만들어 프로그램 디렉토리에 있는 파일들을 간단히 설명한다.

3. 제출 방법

- LMS(lms.postech.ac.kr)의 [eClass]의 수강 과목 메뉴에서 [객체지향프로그래밍]의 [강의실입장] 버튼을 클릭하여 강의실로 이동
 → 왼쪽 메뉴에서 [학습활동]-[과제] 선택
 → 과제명 중에서 해당 ASSN{1-5}를 클릭
 → [과제제출] 버튼 클릭
- 보고서 PDF 파일을 업로드 한다. **보고서 파일 이름은 학번.pdf로 한다.**
- 소스 코드 파일들(*.cpp, *.h)과 Makefile을 묶은 zip 파일을 업로드 한다. **zip 파일 이름은 학번.zip으로 한다.**
- 소스 코드 파일 외에, 불필요한 파일들(*.obj, *.o, *.exe, a.out 등)은 zip 파일에 포함하지 않는다.

4. 채점 기준

- 프로그램 기능 (전체점수의 50%)
 - 프로그램이 요구 사항을 모두 만족하면서 올바르게 실행되는가?
 - 각 과제에서 중요하게 확인하는 사항들은 과제 문서에 명시되어 있음
 - 조교가 프로그램을 실행해 보고 채점
- 프로그램 설계 및 구현 (전체점수의 35%)
 - 요구 사항을 만족하기 위한 프로그램 및 Class 설계가 잘 되었는가?
 - 설계된 내용이 C++ 언어를 이용하여 적절히 구현되었는가?
 - 보고서에 설명된 프로그램 설계 및 구현에 관한 내용을 기준으로 채점
 - 보고서에 설명된 내용을 프로그램의 소스 코드를 참조하여 확인
- 프로그램의 가독성(readability) (전체점수의 5%)
 - 프로그램이 읽기 쉽고 이해하기 쉽게 작성되었는가?
 - 프로그램의 소스 코드를 이해하기 쉽도록 주석을 잘 붙였는가?
- 보고서 구성 및 내용, 양식 (전체점수의 10%)
 - 보고서는 적절한 내용으로 이해하기 쉽고 보기 좋게 잘 작성되었는가?
 - 보고서의 기술적인 내용과 구성이 적절한가, 설명이 간결하고 명료하게 되었는지를 기준으로 채점

- 보고서의 양식도 점수에 반영
- 추가 점수 (최대 전체점수의 10%)
 - 요구 사항 외의 추가 기능 또는 독창적으로 생각한 기능을 구현한 경우
 - 추가된 기능의 독창성과 난이도를 기준으로 채점
- 감점
 - 제출 기한이 지나면 얻은 총점의 20% 감점
 - 추가로 하루(24시간) 늦을 때마다 20%씩 감점
 - 1일 이내 지연: 20% 감점, 2일 이내 지연: 40% 감점, 5일 이상 지연: 0점
- 유의 사항
 - 조교는 LMS에 제출한 프로그램 파일들을 컴파일한 후에 test data에 대하여 실행하여 본다. MinGW 환경에서 프로그램 파일을 컴파일할 때 error가 나면 모든 test data에 대하여 실행되지 않는 것으로 본다.
 - Makefile이 없거나 make를 통해 정상적으로 컴파일이 되지 않을 경우 모든 test data에 대하여 실행되지 않는 것으로 본다.
 - 프로그램의 설계 및 구현은 프로그램의 소스 코드를 보고 확인하기는 하지만, 주로 보고서에 설명된 내용을 바탕으로 평가한다. 즉, 보고서에 설명된 프로그램의 전체 구조, 알고리즘 등을 기준으로 채점한다. 따라서, 보고서에 프로그램의 설계 및 구현을 잘 설명하여야 한다.
 - 보고서 구성 및 내용, 양식은 보고서를 얼마나 잘 작성하였는가를 기준으로 채점한다. 즉, 프로그램의 구조에 대한 설명이 잘 되어있는가, 토론, 결론, 및 개선 방향 등이 잘 정리되어 있는가, 전체적인 보고서 구성이 좋은가, 보고서 양식이 보기 좋은가 등을 기준으로 평가한다. 보고서 구성 및 내용, 양식을 채점할 때는, 프로그램의 구조가 잘 설명되어 있는가 만을 보고, 프로그램 설계 및 구현의 우수성은 점수에 반영하지 않는다.
 - 프로그램 파일을 제출하기 전에 반드시 불필요한 파일들(*.obj, *.o, *.exe, a.out 등)을 삭제하여야 한다. 불필요한 파일들을 삭제하지 않고 프로그램을 제출한 경우에는 얻은 총점의 5%를 감점할 수 있다.
 - 다른 사람의 프로그램이나 인터넷에 있는 프로그램을 복사(copy)하거나 간단히 수정해서 제출하면 학점은 무조건 'F'가 된다. 이러한 부정행위가 발견되면 학과에서 정한 기준에 따라 추가의 불이익이 있을 수 있다.
 - 보고서 작성요령에 명시된 것처럼, 인터넷 등의 다른 소스에서 참고한 내용이 있으면 보고서에 출처와 내용을 명확히 기술하여야 한다. 출처가 명시되어 있지 않은 경우 부정행위로 간주될 수 있다.
 - 과제를 하는 과정에서 담당교수나 조교 이외의 사람에게 도움을 받는 것은 부정행위에 해당한다. 수강생들 사이의 토론은 권장되나, 프로그램 소스 코드에 관하여 협력해서는 안

된다. 다른 사람이 소스 코드 작성에 도움을 준 경우에는 보고서에 명확히 기재하여야 한다. 보고서에 기재한 경우는 부정행위는 아니지만 감점이 있고, 보고서에 기재하지 않은 경우는 부정행위이다.

5. 기타 사항

- 프로그램을 하다 보면 결정해야 할 세부 사항이 많은데, 이러한 세부 사항을 처리한 방법과 이유를 보고서에 쓰십시오.
- 독창적인 아이디어와 추가 기능은 보너스 점수를 받을 수 있으므로, 보고서에 명확히 기재하십시오.
- 제출 기한이 지나면 LMS를 통하여 과제를 제출할 수 없습니다. 제출 기한이 지난 후에는 담당 조교에게 email로 보내야 합니다. 늦게 제출된 과제에 대한 감점은 담당 조교에게 email이 도착한 시간을 기준으로 합니다.
- 과제에 관한 문의 사항은 LMS의 [eClass]의 수강과목 메뉴에서 [객체지향프로그래밍]의 [강의실입장] 버튼을 클릭하여 강의실로 이동
→ 왼쪽 메뉴의 [강의실알림]-[질의응답]을 이용하여 글을 올리십시오.