

고급객체지향프로그래밍 실습과제 #06

조용주
ycho@smu.ac.kr

실습 과제 #06

- C언어 컴파일과정은 보통 세 단계로 이루어짐
 - Preprocess(프리프로세스)
 - Preprocessor(프리프로세서)가 C언어 소스코드를 읽고 매크로 (#define, #include 등)를 치환한 새로운 C 소스코드를 생성
 - Compile(컴파일)
 - C 코드를 바이너리 형태의 중간 코드인 오브젝트 파일로 변환
 - Link(링크)
 - 컴파일된 오브젝트 파일들을 합치고 기존에 컴파일되어 있던 라이브러리(수많은 오브젝트 파일들을 합쳐서 한 개의 파일로 만들어놓은 것)등과 연결해서 실행 파일을 생성

실습 과제 #06

- 본 과제에서는 주어진 Preprocessor, Compiler, Linker 인터페이스를 구현하는 C 프리프로세서, C 컴파일러, C 링커를 구현해야 함
- 또한 이들 C 프리프로세서, C 컴파일러, C 링커 등을 컴포넌트로 포함하고 한 번에 빌드할 수 있도록 **퍼사드(Façade)** 패턴으로 구현된 Integrated Development Environment(IDE, 통합개발환경)을 구현해야 함
 - IDE 클래스는 Builder 인터페이스를 구현

실습 과제 #06

- 주어진 파일 중에서 다음 파일들은 수정불가
 - Executable 클래스
 - Linker 인터페이스 (자세한 내용은 Linker.java의 주석 참조)
- 주어진 파일 중에서 다음 파일들은 수정해야 함
 - **Main** 클래스
 - **Builder** 인터페이스 (자세한 내용은 Builder.java의 주석 참조)
- Main 클래스를 실행했을 때 화면에 출력되는 내용은 "실행 결과.txt"파일에 있음

실습 과제 #06 - 구현해야 하는 클래스

□ SourceCode 클래스

- C 코드 파일 이름(확장자 .c)을 멤버 변수로 포함
- 파일 이름은 생성자의 인자로 전달됨
 - 예: new SourceCode("a.c")
- 필요한 get/set 함수 추가할 것

□ ObjectCode 클래스

- Obj 파일 이름(확장자 .obj)을 멤버 변수로 포함
- 파일 이름은 생성자의 인자로 전달됨
 - 예: new ObjectCode("a.obj")
- 필요한 get/set 함수 추가할 것

실습 과제 #06 - 구현해야 하는 클래스

□ Preprocessor 인터페이스

- SourceCode를 입력으로 받고 새로운 SourceCode 객체를 반환하는 preprocess() 함수 포함
- preprocess()에서 하는 일
 - "Preprocessing C code: 파일_이름"을 화면에 출력
 - 파일_이름은 인자로 주어진 SourceCode 파일 이름
 - 주어진 SourceCode 파일 이름 앞에 "preprocessed_"를 붙여서 새로운 파일 이름을 가진 SourceCode 객체 생성
 - "Generating a new C code: 파일_이름"을 화면에 출력
 - 파일_이름은 새로 생성된 소스 코드의 파일 이름 (preprocessed_가 붙은 c 파일 이름)
 - 새로 생성된 SourceCode 객체 반환

실습 과제 #06 - 구현해야 하는 클래스

□ C언어 프리프로세서 클래스(이름은 본인이 정할 것)

- Preprocessor를 구현 (.c 확장자를 가진 파일 이름을 받아서 preprocessed_를 붙인 소스 코드를 생성)

□ 예

- 인자로 전달된 SourceCode가 a.c라는 파일 이름을 가지고 있다면 preprocess() 함수가 호출되면 다음 내용이 화면에 출력되고 preprocessed_a.c 파일 이름을 가진 새로운 SourceCode 객체가 반환됨
- 화면 출력 내용

```
Preprocessing code: a.c
```

```
Generating a new code: preprocessed_a.c
```

실습 과제 #06 - 구현해야 하는 클래스

□ Compiler 인터페이스

- SourceCode를 입력으로 받고 ObjectCode를 반환하는 compile() 함수 포함
- Compile()에서 하는 일
 - "Compiling code: 파일_이름"을 화면에 출력(파일_이름은 SourceCode에 주어진 파일 이름)
 - SourceCode에 있는 파일 이름의 확장자를 원본에서(예: .c) 오브젝트 파일의 확장자(예: .obj)로 변환한 후에 ObjectCode 객체 생성
 - "Generating object code: 오브젝트_파일_이름"을 화면에 출력
 - ObjectCode 객체 반환

실습 과제 #06 - 구현해야 하는 클래스

□ C 컴파일러 클래스(이름은 본인이 지정)

■ Compiler 인터페이스 구현

- 인자로 전달된 SourceCode가 a.c라는 파일 이름을 가지고 있다면 compile() 함수가 호출될 때 다음 내용이 화면에 출력됨
- 화면 출력 내용

```
Compiling code: a.c  
Generating object code: a.obj
```

□ C 링커 클래스(이름은 본인이 지정)

- Linker 인터페이스 구현(Linker.java의 주석 참조해서 구현)

실습 과제 #06 - 구현해야 하는 클래스

□ Builder 인터페이스

- Builder.java의 주석문을 참고해서 인터페이스에 들어가야 하는 세 개 함수의 헤더를 선언

□ IDE 클래스

- Builder 인터페이스를 구현

□ Main 클래스 일부

- main() 함수 시작 부분에 Preprocessor, Compiler, Linker를 구현한 클래스 객체들을 생성하는 코드 추가
- main() 함수 마지막 부분에 IDE를 사용해서 두 개 exe 파일을 생성하고 실행하는 코드 추가