

SNU 프로그래밍 언어 특강

담당교수: 이 광근
kwangkeunyi.snu.ac.kr

강의: 화/목 14:00–15:15, 302동 309-1호

1 목표

프로그래밍언어 연구에서 다루는 개념들을 익히면서 세 가지 능력을 기른다:

- 핵심만 다루기(abstraction): 꼭 필요한 것만 드러내는 능력이다. 그래서 군더더기에 방해받지 않고 핵심만 파는 능력이다.
- 떼어놓고 다루기(modularity): 독립된 부품으로 따로 떼어 다루는 능력이다. 그래서 부품 조립으로 크고 복잡한 구조물을 만들도록 하는 능력이다.
- 정확히 다루기(precision): 애매하지 않게 각잡고 다루는 능력이다. 그래서 파생 성질을 엄밀히 확인하고 소통하는 능력이다.

이 세가지 능력은 소프트웨어 전문가의 기초 체력이다. 튼튼할수록 좋다. 이 세가지 능력이 전문가로서의 시점을 상승시켜주기 때문이다. 프로그래밍언어 분야에서 연구하거나 프로그래밍언어를 익힐때 뿐만이 아니다. 일반적으로 소프트웨어를 제작하거나 다룰때도 주요한 기초 체력이다.

또, 프로그래밍언어나 프로그래밍 관련 이야기를 접하거나 물어가며 파들어갈때도 그렇다. 소개한 개념들을 이해하고 용어를 알고 문답하면 더 깊은 내용을 효과적으로 길어올릴 수 있다. 그리고 그 문답의 바닥을 신속히 만나는데도 도움줄 수 있다.

2 내용

- 준비물: inductive definition, set, function, relation, partial order, cpo, formal logic, syntax, semantics, proof rule, soundness, completeness
- 프로그램 의미 다루기: dynamic semantics, denotational semantics, lambda calculus, domain theory, computation as continuity, least fixpoint, opera-

tional semantics, evaluation context, value, binding, environment, memory, continuation, exception, control as value, code as value, abstract semantics

- 타입으로 프로그램 다루기: static semantics, simple type, product type, sum type, curry-howard isomorphism, recursive type, polymorphic type, parametric polymorphism, system F, ad-hoc polymorphism, type class, dependent type, lambda cube, abstract data type, subtype, type checking, type inference

3 성적

숙제 90%, 출석 10%

- 강제사항: 강의실에는 모든 전자기기(핸드폰, 노트북, 노트패드 등)를 지참할 수 없다. 전자기기가 없으면 학생들의 강의 집중도를 향상시키고, 생각하게 유도하고, 수강만족도를 높여준다는 연구와 경험담이 발표되고있다. 본 강의도 그 실험에 동참한다.