

Trip Report on OPLSS 2013

Types, Logic, and Verification

July 21 – August 3, 2013, Eugene, Oregon, US

이원열 (leewy@postech.ac.kr)

머리말

정신 없었던 1학기가 끝나고, 7월 중순쯤에 미국 오레곤 대학교에서 열린 [Oregon Programming Languages Summer School \(OPLSS\)](#) 에 다녀왔습니다. 2주동안 총 8분의 교수님들께서 열정적으로 강의를 해주셨으며, PL과 logic에 대해서 기대했던 것 이상으로 많은 것들을 배울 수 있었습니다. (원래는 Greg Morrisett 교수님께서도 강의해주시기로 되어있었으나, 다른 사정 때문에 아쉽게도 참석하시지 못했습니다.)

여름학교 첫째 주에는 주로 이론적인 부분에 관한 강의를 있었고 (0, 1, 3, 6, 7), 그래서 그런지 교수님들께서 칠판을 주로 사용하셨습니다. 반면, 둘째 주에는 주로 응용적인 부분에 대해서 강의를 해주셨고 (2, 4, 5, 8), 슬라이드를 이용해서 강의를 하셔서 좀 더 여유롭게 수업을 따라갈 수 있었습니다. (첫 날에 엄청난 속도로 진행되었던 Robert Harper 교수님의 3시간 강의도 둘째 주가 좀 더 여유롭게 느껴지는 데에 한 몫 했던 것 같습니다.)

I. Coq

0. Coq Boot Camp [실습] --- Andrew Tolmach

Coq의 기본적인 것들을 가르쳐주는 수업으로, 여름학교가 시작하기 바로 전날에 진행되었으며, "[Software Foundations](#)" 책의 <Prop: Propositions and Evidence> 단원 이전까지를 배웠습니다. 이 책을 따라가면서 설명을 해주시고, 중간 중간에 몇몇 exercise는 저희가 직접 해보는 방식으로 수업이 진행되었습니다.

Coq을 미리 어느 정도 다뤄봤거나 전산논리 수업을 이미 수강했다면, Coq Boot Camp에 참가하지 않아도 될 것 같습니다. (다만, 위 책의 앞부분을 미리 한 번 훑어보고 오면 충분할 것입니다.)

1. Software Foundations in Coq [실습] --- Andrew Tolmach

Coq Boot Camp 때와 같은 교수님께서 강의를 해주셨으며, "Software Foundations" 책에서 <Prop>, <More Prop>, <Logic>, <ProofObjects>, <MoreInd>, <Imp>, <Hoare>, <Auto> 단원을 다뤘습니다. Coq을 어떻게 사용하는지 (Coq을 이용하여 수학을 어떻게 formalize하고, 어떠한

tactic들이 있는지) 에 초점이 맞춰졌습니다. Coq Boot Camp와는 다르게 강의 때 exercise를 직접 해볼 시간은 주시지 않고, 중요한 내용들만 빠르게 잡고 넘어가는 방식으로 수업이 진행되었습니다.

이 수업은 Coq을 어떻게 사용하는지를 배우는 것이 목표이므로, 다양한 exercise들을 직접 스스로 해보는 것이 중요한 것 같습니다. 그리고 이번 여름학교에서 Coq을 처음 배웠는데, "Software Foundations" 책이 정말 좋은 Coq 입문서라고 생각됩니다.

2. Verifying LLVM Optimizations in Coq --- Steve Zdancewic

Upenn PL group에서 최근에 진행되고 있는 [Vellm \(Verifying the LLVM\) project](#)에 대한 강의였습니다. LLVM IR을 단순화한 Vminus에 대해서 설명하신 뒤, 간단한 언어인 Imp에서 Vminus로 compile하는 과정이 correct하다는 것을 Coq으로 어떻게 formalize할 수 있는지를 보여주셨습니다. (이것만 하더라도 양이 결코 적지 않아서, Coq code에서 definition과 theorem의 statement들만 간단 간단하게 훑어보았습니다.) 그리고 Vminus에서 full LLVM IR로 확장을 할 때, 어떠한 issue들을 고려해야 하는지와 [CompCert](#)와의 차이점에 대해서 설명해주셨습니다.

강의 앞 부분은 이전에 수강했던 compiler 수업에서 다뤘던 내용들이라, 다른 강의들보다 오히려 좀더 쉽게 들을 수 있었습니다. 이번 강의를 듣기 전에는 "compiler가 correct하다"는 것을 정의하는 것이 별로 어렵지 않을 것이라는 생각을 가지고 있었는데, 알고 보니 이것을 정의하는데에도 여러 가지 choice가 있고, 각각을 정확히 정의하는 것이 전혀 trivial하지 않다는 것을 느꼈습니다. 그리고 이번 여름학교에서 강의하신 Stephanie 교수님과 결혼하신 사이라는 걸 나중에 알고 나서 좀 신기해 했습니다. (두 분 다 upenn CS에서 PL 분야를 연구하고 계시더군요.)

II. Dependent Type Theory

3. Type Theory Foundations --- Robert Harper

가장 기억에 남는 강의였습니다. intuitionistic dependent type theory와 최근에 활발히 연구 중인 homotopy type theory (HoTT)¹의 전반적인 내용에 대해서 다뤘습니다. 좀 더 자세하게는, dependent type, definitional equality, universe, quantifications (Π, Σ types), identity relation (extensional/intentional), weak ∞ -groupoid, functionality/functoriality, path induction, univalence axiom, proof relevant mathematics에 대해서 배웠습니다. 제한된 시간 안에 많은 내용을 다루려고 하셔서 중요한 key idea와 insight 위주로 설명을 해주셨고, 수업시간 내내 거의 실새 없이 말씀을 하셔서 개인적으로 진도가 매우 빠르게 느껴졌습니다. 자세한 내용은 "[Homotopy Type Theory](#)" 책이나 "[Programming in Martin-Lof's Type Theory](#)" 책을 참고하면 좋을 것 같습니다.

¹ 작년 가을부터 올해 봄까지 IAS에서 "Univalent Foundations of Mathematics"라는 special program이 있었을 정도로, HoTT는 요즘 상당히 hot한 분야라고 생각됩니다.

작년 여름학교에서도 Bob 교수님께서 비슷한 내용을 다루신 것 같았으나, 올해에는 더 많은 부분을 다루시면서 강의 횟수는 1번이 적어서 수업이 더욱더 빠르게 진행되었습니다. (그런데 정작 교수님께서서는 미리 계획했던 것보다 진도를 덜 나가서 아쉬워하셨습니다!) 저의 경우는 intuitionistic type theory 자체를 처음 접하는 상황이어서 수업을 따라가기가 매우 힘들었습니다. 다만, 이후에 있었던 Dan과 Stephanie 교수님의 강의를 통해, Bob 교수님이 말씀하셨던 내용이 어떤 의미였는지 대략적으로나마 파악할 수 있었습니다. 아무래도 시간을 두고 강의 내용을 다시 한 번 차근차근 살펴봐야 할 것 같습니다. 따라가기는 힘든 수업이었으나, HoTT가 정말 멋있고 아름다운 이론이라는 것을 알기에는 충분한 수업이었습니다.

5번의 강의를 통해, 교수님의 type theory와 constructivism에 대한 강한 믿음 (그리고 set theory를 별로 탐탁지 않게 생각하시는 것)을 느낄 수 있었습니다. 강의 중간 중간에 rule들을 설명하시면서 "이러한 것들이 얼마나 멋있는가! 정말 아름답다!" "type theorist에게는 이러한 rule들이 너무나 직관적이지 않은가! 반면 set theory에서의 axiom들은 너무 arbitrary하다..." 등의 말씀을 연발하셨습니다. 그리고 교수님이 어릴 적에 proof by contradiction을 이용한 증명을 처음 접하셨는데, 이것이 왜 원래 명제에 대한 증명이 되는지 도저히 납득이 되지 않으셨다고도 합니다. 수업을 듣는 내내 Bob 교수님의 강력한 포스가 느껴졌고, 정말 통찰력이 대단하신 분이라는 느낌을 받았습니다. 그 때문인지 아니면 강의 내용이 어려워서인지 (아니면 진도가 빨라서인지), 다른 강의들에 비해서 학생들의 질문 자체가 적더군요.

4. Dependently-Typed Programming in Agda [실습] --- Daniel Licata

Agda에서의 dependently-typed programming, extrinsic/intrinsic verification, programming in HoTT에 대해서 다룬 강의였으며, 제작년에 Bob 교수님 아래에서 Ph.D를 받은 학생인 Daniel이 강의를 맡았습니다. 직접 작성하신 [lecture note](#)에 설명이 정말 자세하게 되어 있어서, 내용을 이해하는데 많은 도움이 되었습니다.

Bob 교수님 강의를 들을 때, code 자체가 proof가 된다는 내용이 추상적으로만 다가왔었는데, intrinsic proof를 직접 Agda로 해보면서 좀 더 구체적으로 와 닿았습니다. Coq과 Agda를 비교해보자면, Coq의 syntax가 ML과 유사한 반면 Agda의 syntax는 Haskell과 비슷합니다. 이 때문에 ML syntax에 익숙한 저에게는 Agda code가 약간 생소했습니다. 그리고 Coq에서는 term과 proof를 쓸 때 서로 다른 언어 (Gallina와 Ltac) 를 사용하지만, Agda에서는 이 둘 사이에 구분이 없고 모든 proof를 (tactic의 사용 없이) 손으로 써야 한다는 점도 신기했습니다. Daniel은 이 둘로 증명하는 것을 video game을 하는 것에 비유하면서 "Coq은 여러 개의 key가 있고 각각의 key를 충분히 눌러야 레벨 업이 되는 게임과 비슷하고, Agda는 테트리스와 같이 5개 정도의 조각이 있고 이들을 잘 배열하는 게임과 비슷하다"고 하더군요 :)

5. Designing Dependently-Typed Programming Languages [실습] --- Stephanie Weirich

간단한 dependently-type language인 "[pi-forall](#)"을 Haskell을 이용하여 어떻게 구현할 수 있는지에 대한 강의였습니다. 완벽한 언어를 구현하는 것이 목적이 아니라, 각각의 요소들을 어떠한 방식으로 구현할 수 있는지 그 idea들을 전달하는 것에 초점이 맞춰졌습니다.

다른 강의들과는 다르게 슬라이드와 판서는 거의 없이, 몇몇 부분을 직접 구현해 보이시면서 여러 design choice들에 대해서 학생들과 discussion하는 방식으로 수업이 진행되었습니다. 이 때문에 개인적으로는 수업을 따라가기가 약간 힘들었습니다. (dependent type을 아직 완전히 이해하지 못한 상태여서 따라가기가 더욱 힘들었습니다.)

III. Others

6. Logical Relations --- Amal Ahmed

Logical relation²을 이용하여 strong normalization과 type safety (for simply typed lambda calculus + recursive types), contextual equivalence (in System F), compiler correctness를 어떻게 증명할 수 있는지에 대해서 다뤘습니다. Strong normalization 부분은 ["Type and Programming Languages"](#) 책의 12장을, contextual equivalence 부분은 ["Advanced Topics in Types and Programming Languages"](#) 책의 6장을 참고하면 좋을 것 같습니다.

수업을 듣기 전에는 "logical relation"이 무엇인지 모르고 '이게 PL 분야랑 어떤 연관이 있지?' 라는 생각이 들었는데, 알고 보니 PL 분야에서 많이 쓰이는 중요한 proof technique이어서 배우길 잘했다는 생각이 들었습니다.

7. Linear Logic and Session-based Concurrency --- Frank Pfenning

박성우 교수님의 지도교수님이신 Frank 교수님께서 강의를 맡으셨습니다. 작년까지의 여름학교에서 proof theory를 강의하신 것과는 다르게, 올해에는 linear logic을 중심으로 linear inference, sequent calculus, proof as process, session-based programming, focusing and polarity에 대해서 강의해주셨습니다. 자세한 내용은 교수님의 [Linear Logic](#) 수업을 참고하면 좋을 것 같습니다.

Frank 교수님께서 Bob 교수님과 함께 CMU CSD에 계시지만, 강의 스타일은 많이 달랐습니다. Bob 교수님은 많은 내용을 커버하는 것에 중점을 두셨다면, Frank 교수님께서 학생들 전체가 모두 따라오는 것을 목표로 하셨습니다. 첫 강의를 시작하실 때, "my ultimate goal is not as ambitious as the one that Bob sets." 라고도 말씀하셨습니다 :) 교수님의 목표대로 대부분의 수업은 따라가기가 수월했으나, linear logic을 concurrent process로 해석하는 수업에서는 monad, pi-calculus 등이 언급되면서 약간 이해하기가 어려웠습니다.

8. Adventures with Types in Haskell --- Simon Peyton-Jones

Haskell의 major contributor로 잘 알려진 MSR Cambridge에 계시는 Simon 교수님께서 이 강의를

² Strong normalization과 같은 위의 성질들을 증명하려면 induction hypothesis가 충분히 strong해야하는데, logical relation을 이용하면 strong한 induction hypothesis를 체계적으로 만들 수 있습니다.

말아주셨습니다. (이번 여름학교의 연사 분들 중 유일하게 학교에 계시지 않는 분이었습니다.) Haskell의 type과 관련하여 본인이 작성하신 여러 논문들을 핵심만 간추려서 정말 재미있게 설명해주셨습니다. 자세하게는 Haskell의 type class, kind and GADT, type directed compilation and System FC, type function, type inference and constraint solving에 대해서 다루셨습니다.

맨발로 돌아다니시면서 정말 열정적으로 강의를 해주셨습니다 :) (이전 trip report들을 살펴보니, 매년 여름학교마다 맨발로 강의하신 분이 한 분씩은 꼭 있는 것 같습니다.) 그리고 올해 연사 분들 중에 가장 재미있었던 분이라고 생각합니다. 아쉽게도 영국식 억양에 별로 익숙하지 않아서, 많은 농담과 재미있는 이야기를 이해하지 못했습니다 $\pi\pi$ 그리고 Haskell도 이번에 처음 접해보는 언어여서 수업 뒷부분은 약간 어렵게 느껴졌습니다.

맺음말

여름학교에 오기 전에는 복습하고 숙제를 해볼 시간이 많을 줄 알았는데, 시차 적응 등의 이유로 실제로는 전혀 그렇지 않았던 것 같습니다. 그리고 여름학교에 참석하기 전에 준비할 시간이 많지 않아서 예습을 많이 하지 못했는데, '기본 배경지식을 조금 더 많이 쌓고 수업을 들었으면 도움이 더 많이 되었겠다' 라는 아쉬움이 남습니다. 나중에 OPLSS에 참가하게 될 학생은 미리미리 준비를 해서, 기본적으로 읽어오라는 것들은 꼭 공부해서 가면 좋겠습니다.

여름학교 수업을 들으면서 신기했던 점은 여러 교수님들께서 다른 교수님들의 수업도 꽤나 많이 들으신다는 것입니다. 이뿐만이 아니라 다음에 진행되는 자신의 수업에서, 다른 교수님의 이전 수업 내용을 연관 지어서 설명하시는 경우가 많았습니다. 이러한 방식 덕분에 서로 다른 수업 사이의 연결 관계를 알기 쉬웠고, 내용을 이해하기가 수월했다고 생각합니다.

이번 여름학교에서는 Coq과 더불어 dependent type에 관한 내용이 많았던 것 같습니다. 원래는 Coq을 배우는 것이 OPLSS에 참석한 주 목적이었는데, 이러한 커리큘럼 덕분에 dependent type theory와 HoTT에도 많은 관심을 갖게 되었습니다.

2주동안 정말 다양하고 깊이 있는 내용들을 배울 수 있었고, PL 분야와 많이 친해질 수 있었습니다. PL 수업을 듣고 연구참여를 하면서 제가 어렵듯이 느꼈던 PL 분야는 정말 빙산의 일각이라는 생각이 저절로 들었습니다. 다만, 단기간에 너무나도 많은 새로운 지식을 배우게 되어서 아직은 완전히 정리가 되지 않은 것 같습니다. 앞으로 시간적인 여유를 두고 다시 복습해 볼 계획입니다.

더불어 영어 공부가 아직도 많이 필요하다는 것을 느꼈습니다. 우선, 빠른 영어로 강의하시는 것을 들으면서 동시에 필기를 하려니까 놓치는 부분들이 조금 있었습니다. 그리고 학생들과 교수님들이 질문/답변을 할 때에는 정말 빠르게 이야기가 오고 가고 다양한 억양이 등장하는데, 이 때 제가 이해하지 못했던 경우가 더 많았다고 생각합니다. 이러한 것들에 익숙해질 수 있도록 앞으로 계속 노력해야 할 것 같습니다.

마지막으로 느낀 점은 앞으로 제가 공부하고 배워나가야 할 것들이 정말 많다는 것입니다. 저에게는 모든 강의 내용이 다 새로웠으며, 몇몇 강의는 배경지식의 한계로 뒷부분을 잘 이해하지 못했습니다. 반면에 여러 학생들은 이미 강의 내용의 대부분을 미리 알고 있는 것 같았으며, 그보다 더 나아간 질문을 하는 경우가 빈번했습니다. 이러한 학생들과 비슷한 수준에서

연구를 하려면 앞으로 더 열심히 공부를 하는 방법 밖에는 없다고 생각합니다.

이렇게 많은 것들을 보고 배우고 느낄 수 있게 도와주신 박성우 교수님께 감사드립니다.



그림 1. 둘째 주에 강의실 건물 계단에서 찍은 단체사진.
계단 맨 아래에 Steve, Stephanie 교수님과 어린 두 자녀분,
그리고 OPLSS organizer 중 한 분이신 Zena 교수님이 보입니다.
계단 중간 왼쪽에 Daniel과 계단 윗부분 오른쪽에 Simon 교수님도 보입니다.