Interactive Theorem Proving - A. Chlipala (Notes)

Satyendra Kumar Banjare
January 3, 2019

1 Lecture 1

This is a fair introduction to what is the basic idea of interactive theorem proving. We are given some pre conditions , post conditions and the core idea/theorem to check. Examples include :

- Solving a simple linear equation for x, eg: $y = m * x + b$. The pre conditions are that $m \neq 0$. Post conditions involve the value of x obtained actually satisfies the original equation.
- Alias analysis involves determination of optimum strategy to find the number of ways a particular memory address can be accessed. using ITP techniques we can assertain this by checking and case elimination of redundant pointers.
- Anderson's Analysis Often called Anderson-Style Pointer analysis involves the flow sensitive pointer analysis and pointer mutation. It follows assigning a set-notation to pointers bounded by given constraints. It treats all allocations done by one instruction as if they are being done to only 1 object. We define $PT(x)$ as a set that approximats all the locations that can be ppointed by the variable x. Different constraints are generated according to the type of modifications done. We then case-by-case analize them. for better examples : `https://www.seas.harvard.edu/courses/cs252/2011sp/slides/Lec06-PointerAnalysis.pdf`.

2 Lecture 2

The lecture 2 introduces the idea of first order logic, propositional logic and the deduction system. The proofs are merely a set of ad-hoc ideas that use high level argument techniques to finally achieve the assumed equational equivalence. An example of proving by induction is explained in the lecture. We should be able to reason every step we take to achieve our next small goal.

- **Propositional Logic** is like SAT problem or rather we should say that SAT problem is an example of propositional logic. The variables used are assumed boolean and equations formed from them are assumed to be a deductive property. The outcome is desired only if the initial conditions are met. Hence the deduction system is defined and is expressed as :

$$\frac{A\ B}{A \land B} \rightarrow I$$

  for a derived property I.

  Natural Deduction uses base coditions as some inital properties that may themselves be derived from some other.

- **First Order Logic** can be understood as a general mathematical statement made that may or may not have a constraint domain and range. A infinite possible cases may be generalized and hence ’Truth Table’ analogue is diffficult to produce. In this case thus we can use Natural Deduction techniques to develop a more concrete system to analize things.

3 LECTURE 3

The Lecture 3 introduces the idea of Peano’s Axioms and contradicts the set theory approach and argues the approach of using some inductive defiitions and basic data structure to be the more fundamental foundational mathematics theory - the basic idea of Type Theory. The Coq Proof system is also dependent on Type theory fundamentals and has some very basic data structure ideas and possibility of recursive definitions.In Coq Terminology, Inductive Definitions and Fixpoint Definitions (for recursive definitions) are defined to get the readers an idea of doing theing using type theory fundamentals. Proof state Reductions are also explained with a small example. Coq Tactic : Reflexivity is also introduced.