COMP325 - Programming Languages in Theory and Practice

Fall 2015

1 Case Studies

The C language and the Lisp family of languages are some of the most famous, influential, widely known, and used languages in computing history. The story of their early development is documented in Richie's *The Development of the C Language* (Ritchie, 1993) and McCarthy's *History of Lisp* (McCarthy, 1978). First, go look up these authors and their history. Your goal for these readings is simple: *understand the context in which the languages were built, the goals of the designers, and the problems faced in language design and implementation.* Imagine you wan to make your mark in programming languages, what can and should you learn from the stories of LISP and C?

2 Language Design and Community

Guy Steele is a big name in the Programming Languages community. Go look up him up. Back in 1998 he gave the keynote speech at OOPSLA (go look that up!) and that speech has become a bit of a rallying cry for a segment of the language design community. You can watch it here: `http://www.youtube.com/watch?v=_ahvzDzKdB0`. You've also been given a written variation of the talk to read(Steele, 1999) Think about Steele's comments and ideas in comparison to those of Ritchie and McCarthy. What does he add to the conversation? What does he reinforce?

3 Paradigms

Now read Robert Floyd's Turing Lecture *The Paradigms of Programming*(Floyd, 1979) and the first three sections of Peter Van Roy's *Programming Paradigms for Dummies: What Every Programmer Should Know*(Van Roy, 2009). Again, look up the author's and find out who they are and what their credentials are.

Now, as you read look for insight on the following questions:

1. *What is a Programming Paradigm?*
   - list concrete examples
   - list the authors' definitions. (compare and contrast)
   - What is the purpose of a paradigm?
   - From where do paradigms originate?
   - To what extent is the distinction technical vs. social?
   - **Definition in your own words**
2. *What's the relationship (in theory and practice) between programming languages and paradigms?*
3. *How can programmers and program language designers invent/discover new paradigms and programming concepts?*

4 The Assignments

1. Reading and Discussion

   (a) *For Wednesday 8/26* Read the McCarthy and Ritchie papers. Come ready to compare and contrast the evolution of these languages and the goals and challenges of Language design and implementation.

   (b) *For Friday 8/28* Read/Watch Steele's OOPSLA talk and be ready to discuss it and compare his ideas to those of Ritchie and McCarthy.

   (c) *For Monday 8/31* Read Floyd and Van Roy. Be ready to discuss the prompts about Paradigms.

2. Written Assignment *Due Wednesday 9/2*
   Write a one to two page reflection on the readings, our discussions, your experiences with programming languages, and your expectations for the remainder of the course. Try to draw some kind of consistent thread between everything you've read and done. Your goal is not to answer to the above prompts directly but look for some deeper insight that comes from the answers you've found and the comparisons you've drawn.

References

Robert W. Floyd. 1979. The Paradigms of Programming. *Commun. ACM* 22, 8 (1979), 455–460.

John McCarthy. 1978. History of LISP. *SIGPLAN Not.* 13, 8 (1978), 217–223.

Dennis M. Ritchie. 1993. The Development of the C Language. In *The Second ACM SIGPLAN Conference on History of Programming Languages* (1993) *(HOPL-II)*. ACM, 201–208.

Guy L. Steele, Jr. 1999. Growing a Language. *Higher Order Symbol. Comput.* 12, 3 (1999), 221–236.

P. Van Roy. 2009. Programming Paradigms for Dummies: What Every Programmer Should Know. *New Computational Paradigms for Computer Music* (2009).