# Project (technical computing) information review

**Language design**

(D.E.Knuth, 1984)

The ideas of Knuth have been substantially built on since this original article, but the core idea of designing languages for humans to read, rather than for computers, is here. I plan to find more recent works as well as examples of languages developed using ideas inspired by Knuth.

**Data science**

(MIT OpenCourseWare, 2014)

Reputable source and a good place to start for introduction to data science. Being a software engineer I’m not too familiar with how to approach writing efficient algorithms, and this has been helpful in how to think about this.

(mycodeschool, 2013)

The demonstrations of implementations complement the theory from both this series of videos, and the MIT lectures. In particular, I will be implementing the checking for balanced parentheses and binary tree traversal algorithms featured in these videos.

**Tools**

(Stanley B. Lippman, Josée Lajoie & Barbara E. Moo, 2012)

The chapters about the STL and templates have been particularly helpful so far. This is my go to for C++ 11 and previous features.

(Friedl, 2006)

Regular expressions are very powerful; I hope to make use of them in the lexical analysis stage of my compiler, and this book contains all the information, and more, I will need to use them effectively.

**Compiler design and implementation**

(Dick Grune, Kees van Reeuwijk, Henri E. Bal, Ceriel J.H. Jacobs, Koen Langendoen, 2012)

My primary source of information for understanding what a compiler is and does, as well as a good place to start for implementing one myself. It also has been helpful for things like grammar specification.

(Bisqwit, 2017)

Excellent series of youtube videos showcasing implementation of a compiler in C++. The parser is generated using the tool ‘GNU Bison’ and the lexer is generated with the tool ‘re2c’.

# Bibliography

Bisqwit. (2017, December 29). *Bisqwit - YouTube.* Retrieved from YouTube: https://www.youtube.com/watch?v=eF9qWbuQLuw&list=PLzLzYGEbdY5n9ITKUqOuRjXkRU5tMW2Sd&index=2

D.E.Knuth. (1984). Literate Programming. *The Computer Journal*, 97-111. Retrieved from Literate Programming.

Dick Grune, K. v. (2012). *Modern Compiler Design second edition.* New York: Science & Business Media.

Friedl, J. E. (2006). *Mastering Regular Expressions 3rd edition.* Sebastopol: O'Reilly Media.

mycodeschool. (2013, October 28). *Data structures - YouTube.* Retrieved from YouTube: https://www.youtube.com/playlist?list=PL2\_aWCzGMAwI3W\_JlcBbtYTwiQSsOTa6P

OpenCourseWare, M. (2014, July 2). *MIT 6.006 Introduction to algorithms, fall 2011 - YouTube.* Retrieved from YouTube: https://www.youtube.com/playlist?list=PLUl4u3cNGP61Oq3tWYp6V\_F-5jb5L2iHb

Stanley B. Lippman, J. L. (2012). *C++ Primer fifth edition.* Upper Saddle River: Addison Wesley.