

Література:

1. Соломатин Д.И., Копытин А.В., Другалев А.И. Основы синтаксического разбора, построение синтаксических анализаторов. Воронеж, 2014. — 57 с.
2. Орлов С.А. Теория и практика языков программирования. Учебник для вузов. — СПб.: Питер, 2013. — 688 с.
3. Niklaus Wirth - Compiler Construction. URL: <https://people.inf.ethz.ch/wirth/CompilerConstruction/>
4. Compiler Construction. The Art of Niklaus Wirth. URL: https://www.researchgate.net/profile/Hanspeter_Moessenboeck/publication/221350529_Compiler_Construction_-_The_Art_of_Niklaus_Wirth/links/0deec5213875d84735000000.pdf
5. Windows Assembly Programming Tutorial. URL: <https://doc.lagout.org/operating%20system%20Windows/winasmtut.pdf>

Додаткова література:

6. Abdulaziz Ghuloum. An Incremental Approach to Compiler Construction. URL: <http://scheme2006.cs.uchicago.edu/11-ghuloum.pdf>
7. Making Compiler Design Relevant for Students who will (Most Likely) Never Design a Compiler. URL: https://www2.cs.arizona.edu/classes/cs453/fall20/DOCS/teaching_compilers.pdf
8. Regular expressions in lexing and parsing. URL: <https://commandcenter.blogspot.com/2011/08/regular-expressions-in-lexing-and.html>
9. Backus–Naur form. URL: https://en.wikipedia.org/wiki/Backus–Naur_form
10. Extended Backus–Naur form. URL: https://en.wikipedia.org/wiki/Extended_Backus–Naur_form
11. Parsing expressions by precedence climbing. URL: <https://eli.thegreenplace.net/2012/08/02/parsing-expressions-by-precedence-climbing>
12. A recursive descent parser with an infix expression evaluator. URL: <https://eli.thegreenplace.net/2009/03/20/a-recursive-descent-parser-with-an-infix-expression-evaluator/>
13. Top-Down operator precedence parsing. URL: <https://eli.thegreenplace.net/2010/01/02/top-down-operator-precedence-parsing/>
14. Kinga Baxter – Clean Code. URL: <https://kingadesign.com/clean-code-poster-free-download?fbclid=IwAR2p9y7Rf5gyH586sXYxp2dgoA3kWvRIL05jfgeHlj77k1Y94qM1grjdW84>
15. Hints for Computer System Design. URL: <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/acrobat-17.pdf>
16. GitHub. Sean Barrett. URL: https://github.com/nothings/stb/blob/master/stretchy_buffer.h

17. International Standard ISO/IEC 9899:201x. Programming languages C. 2011. — 711 c. URL: <http://www.open-std.org/jtc1/sc22/wg14/www/docs/n1570.pdf>
18. The Python Language Reference. URL: <https://docs.python.org/3/reference/>
19. A Brief And Brisk Overview of Compiler Architecture. URL: <https://felixangell.com/blogs/compilers-brief-and-brisk>.
20. The Super Tiny Compiler. URL: <https://github.com/jamiebuilds/the-super-tiny-compiler>
21. YouTube – Bitwise. URL: <https://www.youtube.com/user/pervognsen>
22. GitHub - pervognsen/bitwise. URL: <https://github.com/pervognsen/bitwise>
23. Wikipedia - Tree traversal. URL: https://en.wikipedia.org/wiki/Tree_traversal#Post-order
24. Daniel Kusswurm - Modern X86 Assembly Language Programming: 32-bit, 64-bit, SSE, and AVX, 1st Edition. URL: <https://www.amazon.com/Modern-X86-Assembly-Language-Programming/dp/1484200659>
25. Thomas H. Cormen - Introduction to Algorithms, 3rd Edition (The MIT Press). URL: <https://www.amazon.com/Introduction-Algorithms-3rd-MIT-Press/dp/0262033844>