

## Lisp in Summer Projects Submission

|                                    |  |
|------------------------------------|--|
| <b>Submission Date</b>             | 2013-10-08 17:42:37  |
| <b>Full Name</b>                   | Sergii Dymchenko   |
| <b>Country</b>                     | USA  |
| <b>Project Name</b>                | Solving Google Code Jam problems in Common Lisp  |
| <b>Type of software</b>            | command-line/terminal app  |
| <b>General category</b>            | other  |
| <b>LISP dialect</b>                | Common Lisp  |
| <b>GitHub URL</b>                  | <a href="https://github.com/kit1980/cl-google-code-jam">https://github.com/kit1980/cl-google-code-jam</a>  |
| <b>Did you start this project?</b> | Yes, all the code is written by me   |
| <b>Project Description</b>         | I want to describe my project in this form.  |
| <b>Purpose</b>                     | Article that presents Common Lisp solutions to several Google Code Jam problems.   |
| <b>Function</b>                    | Demonstrate various aspects of Common Lisp usage in context of coding competitions and highlight some language features and libraries useful for programming contests.   |
| <b>Motivation</b>                  | Goals of the article include persuading more programming contests participants try Common Lisp and also luring more seasoned Lisp programmers into programming contests.   |
| <b>Audience</b>                    | Programming contest participants, who are new to Lisp. Common Lisp programmers, who are new to programming contests.   |
| <b>Methodology</b>                 | Lisp already has some limited presence in Code Jam and almost all of the problems from this article have correct submitted solutions in Lisp ( <a href="http://www.go-hero.net/jam">http://www.go-hero.net/jam</a> site can be used to find submitted solutions in different programming languages). I tried to make my solutions a bit better engineered than solutions done in a hurry of the real contest: by using modern features like Quicklisp, using |

standard input/output instead of hard-coded file names, more granular code, etc. I provide links to other peoples' Lisp solutions through the article.

I tried to make coding style of the presented solutions reflect specifics of programming contests (without sacrificing readability too much) and multi-paradigm nature of Common Lisp. For example, LOOP macro is used extensively in every program except for one, which is meant to be an illustration of a more functional programming style.

I chose one particular Common Lisp implementation - Steel Bank Common Lisp, which is popular, stable and fast. All programs were tested using SBCL 1.0.29, but should also work with other versions and standard-compliant Common Lisp implementations.

## Conclusion

Demonstrating that Lisp can be successfully used in programming contests in general, and in Google Code Jam in particular.

Future directions for the project include solving more Google Code Jam problems in Common Lisp using more techniques and paradigms: local search, linear programming, using interaction with Maxima computer algebra system, dynamic programming / memoization, etc.

## Official

I have read rules and have abided by them.

I am 18 years of age or older.

I am not living in Brazil, Quebec, Saudi Arabia, Cuba, Iran, Myanmar (Burma), North Korea, Sudan, or Syria.

Text