Lisp in Summer Projects Submission

Submission Date	2013-10-23 15:58:12
Full Name	Cong-Xin Qiu
Country	USA
Project Name	Schemannian
Type of software	library
General category	library
LISP dialect	Racket
GitHub URL	https://github.com/ozooxo/Schemannian
Did you start this project?	Yes, all the code is written by me
Project Description	I want to describe my project in this form.
Purpose	"Schemannian" is a scheme/Racket based package for symbolic mathematics for physicist.
Function	"Schemannian" currently supports a realization of Euler-Lagrangian Equation is classical physics, Riemannian geometry and General Relativity calculations, and simple Grassmannian calculus. To build it from the ground, "Schemannian" also includes several supported math functions for calculus, linear algebra, etc.
Motivation	Most of the physics aimed functions "Schemannian" currently supports, are not included in major mathematical softwares (For example, for the Riemannian geometry functions, Maple does support that, but neither Mathematica nor sympy support that). Lots of really tedious physical calculations can still only be done by pencil/pen (e.g., most of the loop calculations which cannot be fixed into standard QED/QCD system, and gauge theories relate to fancy algebraic systems). "Schemannian" aims to automate them in the future.
Audience	Theoretical physicist.

Methodology	"Schemannian" starts from the symbolic calculation exercises giving in Chapter 2 of SICP. However, I expanded them to support simplification of expressions, partial derivative, equation solving, etc. The Euler-Lagrangian equation part of this package is basically a list of closures of mechanical objects with dispatching "kinetic-energy" and "potential-energy". Then, with Euler-Lagriangian equation (which is well-defined in physics) and the support math functions, it can give the equation of motion of those objects. For the Riemannian geometry part, based on SICP, I built a generic system to describe tensor algebra (basically linear algebra in higher dimension), with the ideas of some typical mathematical physical calculations (Einstein summation, etc). Then I define functions to calculate Levi-Civita connection, Riemannian curvature tensor, Ricci tensor, and Ricci scalar. Those functions are well-defined in general relativity textbooks; however, the hard part of this project is to translate them to computer science languages, and let my computer understand how to do that, and to make them correct.
Conclusion	"Schemannian" currently support some useful mathematics/physics related functions, for which people do not really have many alternative choices. However, it still have several limitations. First, to use those function, people need to be familiar with Racket, which is not true for most physicist; hence, it will be better for it to have a more user friendly interface. Second, it will be better if it can support a richer set of physics functions (e.g., the ones I talked about in the "motivation" part -- the ones which no code in the world can do).
Build Instructions	"Schemannian" is currently not included in PLaneT. You can just download all the related files, and run every single of them in Racket.
Test Instructions	You can try the test files in folder /examples . The usages/physical background of the functions are self-explained in the name of the files. Also, there is a much better user's guide (in .rst file) in Github.
Describe any bugs or caveats	The current "simplify" function is still quite weak. Sometimes you can only check by hand that the result it gives is actually correct.
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.