

---

Subject: Re: "(national) C compiler"

Posted by [cbpporter](#) on Tue, 31 Mar 2009 18:38:45 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

I would recommend porting something. C is an extremely complex language and unfortunately you do have to implement most features to have something functional. I would recommend implementing at first two basic types: char and int. That should get you started. Combine with simple pointers, and you have something capable of most tasks that don't require structs. Programing languages are my hobby and I implemented once a "new C++", but I gave up the moment I discovered D.

Code generation is another hard part. Witting x86 assembly is relatively difficult and very difficult if you want to write an optimizing compiler. But if you leave out code generation phase and pass it on to a different tool, you could get a functional mini compiler in about 3000-5000 lines of code. With code generation and optimization you will need hundreds of thousands of lines of code, a full team and at least 10 years of development .

I would recommend checking out Clang. It is based on LLVM. It's not quite ready yet, but the code is relatively simple to understand. Or maybe D sources. They are not that long, but more difficult to understand. And there was a "simple retargetable C compiler" I played around about 5 years ago called lcc.

I hope you find something to get you started in those projects. With Clang you should be able to change the keywords in hours. But still, please don't waste too much time on C .