**Jack Compiler**

This is compiler for the Jack language as described in the Nand2Tetris course. The project was very difficult and I gained an appreciation for the complexity of modern day compilers. While the compiler passed all tests supplied in the course and compiled a simple snake game, there are still probably some hidden bugs. Additionally, the error reporting is very minimal so the instructions provided for bugs in the source code are not very detailed. Anyways, this was mainly a learning exercise. For others going through the nand2tetris course, feel free to take inspiration from my compiler but try to figure it out on your own too. Its challenging, but worth it!

**Getting Started**

If you want to run this compiler on your local machine, follow the instructions below:

**Prerequisites**

* c++ compiler (I used GCC)
* Jack OS vm files (included in the repo)
* A Jack source file (file.jack)

Give examples

**Installing**

Simply pull the source from git and navigate to the directory in the command line

Run make within the directory to compile the code

make

Then execute the file with the jack source that you wish to compile as command line args

./j++ “Main.jack” “Foo.jack”

The compiler will output the compiled vm file to the same directory

**Running the tests**

If you want to test the VM code in isolation you can download the VM Emulator from the Nand2Tetris website. Otherwise you can use the VM and Assembler (see respective repos) and take the code all the way down to binary

**Author**

* **Cameron Jackson**

**Acknowledgments**

* Nand2Tetris Q&A forum was helpful (and prompt) in answering questions