/\* compiler directives \*/

/\*\*

\* #load

\* loads a file into the current file

\*\*/

#load “file\_name.llang”

/\*\*

\* #fn-type

\* tells the compiler to check

\* if this function’s prototype is

\* the same as the MyFuncType defines

\*\*/

MyFuncType FnType = fn(string[]) i32

#fn-type MyFuncType

fn myFunc(args string[]) i32 {

ret myProgram()

}

/\*\*

\* #main

\* tells the compiler which function

\* is the entry point of the program.

\*\*/

#main

fn main(args string[]) i32 {

ret myProgram()

}

/\* NOTE: You could use the predefined

\* FnMainType to check function prototype.

\*/

#fn-type FnMainType

#main

fn main(args string[]) i32 {

ret myProgram()

}

/\*\*

\* #compile

\* tells the compiler this symbol

\* may be used at compile time.

\*\*/

#compile

myCompileTimeVar i32 = 45

#compile

fn myCompileTimeFunc() i32 {

ret 5

}

/\*\*

\* #run

\* tells the compiler to run

\* a funtion at copmile time

\*\*/

#run myCompileTimeFunc()