PH502: Scientific Programming Concepts

Irish Centre for High End Computing (ICHEC)

September 23, 2020

Overview

- Here we will discuss the process of compiling a program.
- The program you write is in a normal text file (not Word), called the source code.
- This file is not understood by the machine. This is the difference between scripts and compiled languages.
- In order to execute the the program on the machine, another program called the compiler converts the source code into a machine readable form (the program).

Compiler

- Each language has its own compiler(s).
- There are two main C compilers: gcc and icc.
- For FORTRAN they are called gfortran and ifort.
- When your source code file has been constructed you can compile it using

```
gcc -o executable my_prog.c
gfortran -o executable my_prog.f90
```

Compile Options

- The first thing to note is that the source has an extension that is related to the language in which it is written (.c or .f90)
- Compilers are complex programs in their own right and have what are called command line options. They modify the behaviour of the compiler.
- The first of these options is $-o$, this tells the compiler what name you want to give to the program.
- In this case *executable* but normally you give it a name that is related to the source code file.
- You run the program or execute it using the line below

```
./executable
```

Optimization

- One of the jobs of the compiler is to optimize the code for the specific machine.
- Source code is usually portable between different machines. But the compiled program will only run on the machine on which it is compiled.
- The programmer can have a big influence over the program's performance but the compiler does a good job automatically.
- Another compile option controls how much optimization is performed $-O\sharp$ where the $\sharp$ is a number between 0 and 3.

```
gcc -O0 -o executable my_prog.c  # no optimization
gcc -O3 -o executable my_prog.c  # aggressive optimization
```

Compile Stages

ICHEC
Irish Centre for High-End Computing

- There are different stages in compilation.
- The first step is called pre-processing.
- Code from different sources can be added to you own, plus all the comments in the code are removed.
- As we will explain, comments in your code are extremely useful to the programmer but completely unnecessary to the machine.
- The next step is to compile the code into what is called assembler code which is specific to the host machine.

Compile Stages

- Assembler code is still human readable but is close to what instructions the machine understands.
- The next stage is to generate machine instructions. The assembler converts the assembler code into object code.
- Object code is not human readable.
- The final stage is linking. We will discuss this later but pre-existing code libraries can be attached to our own code in the linking step.