Introduction

• You will learn
  – How to write C code
  – How to compile and execute C code
  – How to link the math library if the compiler doesn’t do that for you automatically

• What will you do
  – Modify the code for the exercises to embed logic in it
  – Compile and execute the code for the programs discussed in the lecture and exercises
Accessing Lab Files

- Log on to Ranger using your_login_name.

- Uncompress the tar file, trainingC.tar, that is located in the ~train00 directory into your HOME directory.

- Switch to the subdirectory Exercise within the directory trainingC

```bash
ssh <your_login_name>@ranger.tacc.utexas.edu

tar -xvf ~train00/trainingC.tar

cd trainingC/Exercise
```
Exercise 1: mathExample.c

• Objective: Learn to compile and link with both the intel compiler and gcc compiler

• Make sure you are in the directory Exercise

• Intel compiler
  – Load the intel module if it is not already loaded
  – To check the loaded modules, run the command `module list`
  – Compile the code with the intel compiler and run it

  ```
  module swap pgi intel
  icc -o myMathEx mathExample.c 
  ./myMathEx
  ```

• GCC compiler
  – Swap intel module with gcc, compile the code with the gcc compiler

  ```
  module swap intel gcc
  gcc -o myMathEx2 mathExample.c -lm
  ./myMathEx2
  ```

Note, you have to load the Math ("m") library with the gcc compiler.
Exercise 2: celToFar.c

- Modify the program for converting the temperature in Fahrenheit to Celsius
  - Formula for converting Fahrenheit to Celsius: $C = (F - 32) \times \frac{5}{9}$

- Follow the instructions in the comments of CelToFar.c

- Compile and run the code:
  ```bash
gcc -o celtoFar celToFar.c
./celtoFar
```

- Now add the code for converting Celsius to Fahrenheit
  - Formula for converting Celsius to Fahrenheit: $F = \frac{C \times 9}{5} + 32$
Exercise 3: circle.c

• Calculate the circumference and area of the circle
  – Circumference of a circle: $2\pi R$
  – Area of a circle: $\pi R^2$
    \[\pi = 3.14159265\]
    \[R = \text{radius of the circle}\]

• Modify the code in circle.c

• Save the file, compile and run
  \[\texttt{gcc -o circle circle.c}\]
  \[\texttt{./circle}\]
Exercise 4: prime.c

- Write a program to find all the prime numbers between 1 and N, N included

- Modify the code in prime.c
  - Read N from input
  - Write the code for determining the prime number in function named `prime`
  - Note that the function is returning an integer value

- Save, compile and run the code
  
gcc -o prime prime.c
  
  ./prime
Exercises 5 and 6

• If time permits modify the code in files `readInput2.c` and `readInput3.c`
• Desired modifications are described in the comments
• These exercises will give you some practice on I/O related functions