# ECE 71 – Introduction to Computational Programming

Professor Kriehn – Fall 2017

**Code Due By:** Midnight on Friday, September 22, 2017
**Writeup Due By:** Class on Mon/Tue, September 25/26, 2017

## HOMEWORK #09 – Fuel Efficiency Conversion

A liter is 0.264179 gallons. Write a program that will read in the number of liters of gasoline consumed by the user's car and the number of miles traveled by the car and will then output the fuel efficiency given by the number of miles per gallon driven.

**Specifications:**
Define a function to compute the number of miles per gallon. Use the following function prototype:

```
double fuelEfficiency(double liters, double miles);
```

Use a globally defined constant for the number of gallons per liter called GPL. The fuel efficiency should print the result with one digit after the decimal point.

As an example, if you execute the program with the following underlined inputs, the output will be:

```
~> main.o
Enter the liters of gas consumed by the car: 15.2
Enter the miles driven by the car: 137.8

The fuel efficiency is 34.3 mpg.
~>
```

Develop your I/O diagram and pseudocode, debug your code, and submit to the Grader Program.

# ECE 71 – Introduction to Computational Programming

Professor Kriehn – Fall 2017

## HOMEWORK #10 – Game of Craps

Write a program that simulates the game of craps, which is played with two dice. On the first roll, the player wins if the sum of the dice is 7 or 11. The player loses if the sum is 2, 3, or 12. Any other roll is called the "point", and the game continues. On each subsequent roll, the player wins if he or she rolls the point again. The player loses by rolling a 7. Any other roll is ignored and the game continues. At the end of the each game, the program will ask the user whether or not to play again. When the user enters a response other than '`y`' or '`Y`', the program will display the total number of wins and losses and then terminate.

**Specifications:**
Write the program using three functions: `main`, `rollDice`, and `playGame`. Use the following function prototypes:

```
int rollDice(void);
bool playGame(void);
```

`rollDice()` should generate two random numbers, each between 1 and 6, and return their sum. Use the `rand()` function to generate random numbers.

`playGame()` should play one craps game (calling `rollDice()` as many times as necessary to determine the outcome of each dice roll); it will return `true` if the player wins and `false` if the player loses (t. `play_game` is also responsible for displaying messages showing the results of the player's dice rolls.

`main` will call `play_game` repeatedly, keeping track of the number of wins and losses, and displaying the "`You WIN!`" and "`You lose!`" messages. At the beginning main, after displaying a "`Welcome to Craps`" message, you should prompt the user to enter a random seed (to satisfy the grader program).

Finally, use the `exit()` function and the `EXIT_SUCCESS` constant to exit your program (which are both defined in `<cstdlib>`). From this point forward, each of your programs should use the `exit()` function to terminate. In other words, to exit the program:

```
return(EXIT_SUCCESS);
```

As an example, if you execute the program with the following underlined inputs, the output will be:

```
~> main.o
Welcome to Craps

Please enter a random seed: 23

You rolled:     4
Your point:     4
```

```
You rolled:     8
You rolled:     8
You rolled:     5
You rolled:     6
You rolled:    11
You rolled:     7

You lose!

Play again? (Y/N): n

Wins:      0
Losses:    1
~>

~> main.o
Welcome to Craps

Please enter a random seed: 63

You rolled:     7

You WIN!

Play again? (Y/N): y

You rolled:     7

You WIN!

Play again? (Y/N): y

You rolled:    10
Your point:    10

You rolled:     4
You rolled:     7

You lose!

Play again? (Y/N): n

Wins:      2
Losses:    1
~>

~> main.o
Welcome to Craps
```

```
Please enter a random seed: 152
You rolled:     5
Your point:     5

You rolled:     3
You rolled:     5

You WIN!

Play again? (Y/N): q

Wins:      1
Losses:    0
~>
```