

Assignment #1

Introduction to C Programming – COP 3223

Objectives

1. To give students practice at typing in, compiling and running simple programs.
2. To learn how to read in input from the user.
3. To learn how to use assignment statements and arithmetic expressions to make calculations

Introduction: Mission to Mars

Your friend has been playing a new Mars Colony simulator nonstop! They are always talking about how cool it would be if they could be a on the first real-life mission to Mars! To amuse your friend, you have decided to create a series of programs about the possible first colony on Mars.

Problem: How Much Fuel Will We Need? (marsfuel.c)

The biggest problem our mission will face is getting our supplies off Earth and on their way to Mars. Earth and Mars have different orbits, which means the distance between the two planets is always changing. Based on the current distance between these two celestial bodies our astronauts may spend varying amounts of time in transit.

You will need to ask the user for the distance between Earth and Mars on the day of the expected launch. You can then determine the number of days we expect the shuttle to be traveling using the following formula:

Distance / 261,904

If each day requires 11,000 pounds of rocket fuel, determine how much fuel will be necessary for the trip.

This may not be the most precise calculation, so our team will also add in a 20% “contingency” – an extra amount of fuel in case we run out. Add an additional 20% to the amount needed for the trip and print this final amount to the user.

Input Specification

1. The distance will be a positive integer.

Output Specification

Output the number of pounds of fuel needed as a real number rounded to two decimal places using the format below:

You will need X.XX pounds of fuel to get to Mars!

Output Sample

Below are some sample outputs of running the program. **Note that these samples are NOT a comprehensive test.** You should test your program with different data than is shown here based

on the specifications given above. In the sample run below, for clarity and ease of reading, the user input is given in *italics* while the program output is in **bold**. (Note: When you actually run your program no bold or italics should appear at all. These are simply used in this description for clarity's sake.)

Sample Run #1

How far away will Mars be on the day of the launch?

55000000

You will need 2772008.05 pounds of fuel to get to Mars!

Sample Run #2

How far away will Mars be on the day of the launch?

250000000

You will need 12600036.65 pounds of fuel to get to Mars!

Acceptable Resources

Remember, the use of online help sites is strictly prohibited. The only acceptable resources for these assignments are below:

- Course Webcourse
 - In particular: Week 1 – Introduction and Week 2 – Arithmetic in C
- Course Textbook
 - Programming Knights: An Introduction to Programming in Python and C by Arup Guha
- Professor Guha's Course Archive
 - <http://www.cs.ucf.edu/~dmarino/ucf/transparency/cop3223/>
- Course TAs and Instructor Office Hours
 - Getting Help: <https://webcourses.ucf.edu/courses/1336411/pages/getting-help>

Style Notes

Please review the course Style Guide on the webcourse, with special attention to the following notes:

- the maximum length of a line of code is 80 characters long in C
- comment major sections of code addressing: "What does this block do?" and "Why did I implement this block in this way?"
- place comments above the line(s) to which it applies
- use inline comments (//) and leave one space between // and the comment's first character
- All variables should be declared at the top of your functions (in this program, only main is needed).
- Give your variables meaningful names
- Use constants when appropriate
- Be sure to declare `main` with: `int main(void) {`
- Indent the contents of main four spaces or one tab
- leave a space on both sides of any binary operators you use in your code (i.e., operators that take two operands). For example, use `(a + b) - c` instead of `(a+b)-c`.

Deliverables

One source file – *marsfuel.c* – is to be submitted over WebCourses.

Restrictions

Although you may use other compilers, your program must compile and run using a standard C Development Environment. Your program should include a header comment with the following information: your name, assignment number or title, and date. Also, make sure you include comments throughout your code describing the major steps in solving the problem.

Grading Details

Your programs will be graded upon the following criteria:

- 1) Your correctness
- 2) Your programming style and use of white space. Even if you have a plan and your program works perfectly, if your programming style is poor or your use of white space is poor, you could get 10% or 15% deducted from your grade.
- 3) Compatibility – You must submit C source files that can be compiled and executed in a standard C Development Environment. If your program does not compile, you will get a sizable deduction from your grade.