

COP 3223 Program #5: Weather

Part A: Rain and Temperature (rain.c)

You have noticed a very strong correlation between the temperature outside and whether or not it rains. If the temperature is 90 degrees Fahrenheit or higher, it rains. Otherwise, it does not. Furthermore, you have a handy formula for calculating WHEN it'll rain if the temperature is above 90 degrees Fahrenheit:

$$\text{Time} = 5 - (\text{temp} - 90)/5$$

The formula uses regular division (not integer division) and outputs a real number in between 1 and 5, assuming that the temperature is in between 90 and 110 degrees. (It never gets above 110 degrees, luckily!) This number indicates the number of hours past noon that the rain will begin.

For your program, you must ask the user for the temperature in Fahrenheit, **which is guaranteed to be an integer**. If this temperature is 90 or higher, output a statement with the following format:

It will rain at approximately X hours and Y minutes after noon.

where X represents the hour in the afternoon and Y represents the nearest minute, according to the formula given, that it will rain.

If the temperature is NOT 90 or higher (in Fahrenheit), then output the following:

It will not rain today.

Sample Program Run #1

What is the temperature in Fahrenheit?

95

It will rain at approximately 4 hours and 0 minutes after noon.

Sample Program Run #2

What is the temperature in Fahrenheit?

70

It will not rain today.

Part B: Randomized Weather (rain2.c)

Weather is often random! No matter how much work meteorologists do, they generally can only provide probabilities for certain outcomes. In this program, we'll acknowledge this reality and write a small simulator for whether or not it rains. This, use the following formula for calculating the probability it will rain on a certain day, given the temperature in Fahrenheit:

Probability = 0, if the temperature is less than 32
= $|65 - \text{temp}|/100$, if the temperature is in between 32 and 80, inclusive
= $2 * (\text{temp} - 50)/120$, if the temperature is greater than 80

We assume that the temperature will never exceed 110 degrees Fahrenheit.

For your program, you must ask the user for the temperature in Fahrenheit, **which is guaranteed to be an integer less than or equal to 110**. Use this to calculate the probability of rain using the formula above. Print out this probability as a percentage to two decimal places. Finally, use random numbers to “simulate” whether or not it rained on a day with that temperature and print out the outcome. (Note: The correctness of this portion of the code will be ascertained by inspecting the code and NOT just running it, since either outcome is possible in running nearly any scenario. Also, don't forget to seed the random number generator once at the beginning of your program.)

Your first line of output should have the following format:

The probability of rain is X%.

where X is a real number in between 0 and 100 to 2 decimal places indicating the appropriate percentage. For the second line of output, run a simulation using a random number to determine if it rained using this percentage and print out one of the two following lines, depending on the outcome:

In our simulation, it rained!
In our simulation, it did not rain!

Sample Program Run #1

What is the temperature in Fahrenheit?
95
The probability of rain is 75.00%.
In our simulation, it rained!

Sample Program Run #2

What is the temperature in Fahrenheit?
25
The probability of rain is 0.00%.
In our simulation, it did not rain!

Deliverables

Two source files: *rain.c*, for your solution to problem A and *rain2.c* for your solution to problem B. All files are to be submitted over WebCourses.

Restrictions

Although you may use other compilers, your program must compile in gcc and run in the Code::Blocks environment. Each of your two programs should include a header comment with the following information: your name, course number, section number, assignment title, and date. Also, make sure you include comments throughout your code describing the major steps in solving the problem. **Make sure to use good programming style, including use of appropriate constants, good variable names and good use of white space. A significant portion of your grade will be based upon programming style and not correctness. Of course, a significant portion of your grade will also be based upon correctness.**