

Problem Set #1

Assignment Learning Objectives

1. Write a program demonstrating good Fortran code structure
2. Write a program that uses mathematical operations for meteorological calculations, uses interactive read for data input and uses standard output for reporting results
3. Solve basic meteorological problems using the Fortran programming language

Due 27 August 2020 by 3:00 pm

Due Online

- Three scripts in the PSet1 repository on your GitHub account

Due Via Blackboard

- A PDF version of your code. Make sure the honor code appears in the comment block at the beginning of the script and that it has your full typed out name. This will serve as your assertion that you have upheld the honor code.

Honor Code

You must type the full honor code into each comment block to indicate that you have upheld the code. Authorized aid for this assignment is your notes and any previous programs you have written. You are highly encouraged to work with others in the class to help diagnose bugs and work out programming logic. Any copying of someone else's code IS A VIOLATION OF THE HONOR CODE, whether from a classmate, or the Internet. *Be sure to indicate with whom you have worked in the comment block of your submission.*

Problems

1. Meteorologists often need temperatures in different units for different equations and applications. Write a Fortran program that will read a temperature in Kelvin from the command line, convert that temperature to both Celsius and Fahrenheit, and print the three temperature values to the screen. Be sure to include the mathematical equations within the program and to well document your code.

Program Name: **temp_conversion.for**

Program Input: Temperature in Kelvin

Program Output: Temperature in Kelvin, Celsius, and Fahrenheit

Notes:

- Break the problem down into smaller bite size pieces. For example, first write your program to be able to read in a temperature from the command line and print it to standard output. Then submit that working code to your GitHub repository for this assignment. Now go back and add a temperature conversion and develop the appropriate output, and so on.
- Make sure to regularly submit your work to your GitHub site. Part of your grade on each assignment is on your regular submission of the program(s) as they are in progress of being developed. This is easily done when regularly submitting the bite size pieces that you create when breaking down any coding project!

- Test more than just the cases given below. An important part of code development is testing to make sure your code works for ALL cases, so think about "edge" cases that you know to be problematic and test those specifically.

Please make your program to produce output in the same fashion as the test cases below.

Test Case 1

```
>>>
Enter a Temperature in Kelvin:
273.15

Tk: 273.14999
Tc: 0.00000
Tf: 32.00000
>>>
```

Test Case 2

```
>>>
Enter a Temperature in Kelvin:
233.15

Tk: 233.14999
Tc: -40.00000
Tf: -40.00000
>>>
```

Test Case 3

```
>>>
Enter a Temperature in Kelvin:
303.15

Tk: 303.14999
Tc: 30.00000
Tf: 86.00000
>>>
```

2. There are a number of different calculations that meteorologists routinely encounter including potential temperature. Create a Fortran program that will accept input of a temperature in Celsius and the pressure level in hPa, compute the potential temperature for the input data, and print the original temperature in Kelvin and pressure with the calculated potential temperature to standard output.

Potential Temperature Equation: $\theta = T(p_0/p)^{(R_d/c_p)}$

where p_0 is a reference pressure (typically 1000 hPa), T is the temperature at a given pressure p , R_d is the dry gas constant (287 J kg^{-1}) and C_p is the specific heat of air (1004 J kg^{-1}).

Program Name: **potemp.for**

Program Input: Temperature in Celsius and Pressure in hPa

Program Output: Temperature in Kelvin, Pressure in hPa, Potential Temperature

Notes:

- If a previous program has a number of valuable pieces, use it as a starting point. Make a copy of the already completed program and rename to be used on this new problem. That way you don't have to always start a new program from scratch if you don't need too.
- Again, break the problem down into smaller bite size pieces. For example, first write your program to be able to read in a temperature from the command line and print it to standard output. Then submit that working code to your GitHub repository for this assignment. Now go back and add a temperature conversion and develop the appropriate output, and so on.
- Make sure to regularly submit your work to your GitHub site. Part of your grade on each assignment is on your regular submission of the program(s) as they are in progress of being developed. This is easily done when regularly submitting the bite size pieces that you create when breaking down any coding project!

Please make your program to produce output in the same fashion as the test cases below.

Test Case 1

```
>>>
Enter a Temperature in Celsius:
-15

Enter a Pressure in hPa:
850

Temperature (K): 258.14999
Pressure (hPa): 850.00000
Pot. Temp. (K): 270.42584
>>>
```

Test Case 2

```
>>>
Enter a Temperature in Celsius:
-25

Enter a Pressure in hPa:
500

Temperature (K): 248.14999
Pressure (hPa): 500.00000
```

```
Pot. Temp. (K): 302.52805
```

```
>>>
```

3. Having the temperature or potential temperature is great, but for many thermodynamic calculations we want to account for the fact that there is some amount of moisture in the atmosphere and that plays a crucial role in understanding the stability of the lower part of the atmosphere. Therefore we want to calculate the virtual potential temperature, θ_v , to account for the presence of moisture using the following equation, to account for the presence of moisture using the following equation

$$\theta_v = (1 + 0.61q)\theta$$

where θ is the potential temperature and q is the specific humidity, defined as

$$q = r_v / (1 + r_v)$$

where r_v is the dimensionless water vapor mixing ratio.

Write a program to compute the virtual potential temperature from a given temperature in Celsius, for a given pressure value in hPa, and mixing ratio in g/kg. Additionally, compute the difference between the potential temperature and the virtual potential temperature. The output of the program should be the temperature in Kelvin, the specific humidity, the potential temperature in Kelvin, the virtual potential temperature in Kelvin, and the difference of the potential from virtual potential temperature (i.e., $\theta_v - \theta$).

Program Name: **virtual_potemp.for**

Program Input: Temperature in Celsius, Pressure in hPa, Mixing Ratio in g/kg

Program Output: Temperature in Kelvin, Specific Humidity, Potential Temperature in Kelvin, Virtual Potential Temperature in Kelvin, the difference of the Potential from the Virtual Potential Temperature in Kelvin

Notes:

- If a previous program has a number of valuable pieces, use it as a starting point. Make a copy of the already completed program and rename to be used on this new problem. That way you don't have to always start a new program from scratch if you don't need too.
- Again, break the problem down into smaller bite size pieces. For example, first write your program to be able to read in a temperature from the command line and print it to standard output. Then submit that working code to your GitHub repository for this assignment. Now go back and add a temperature conversion and develop the appropriate output, and so on.
- Pay particularly close attention to units of variables in this case. Remember that mixing ratio when used in the virtual potential temperature equation must be *dimensionless*.
- Make sure to regularly submit your work to your GitHub site. Part of your grade on each assignment is on your regular submission of the program(s) as they are in progress of being developed. This is easily done when regularly submitting the bite size pieces that you create when breaking down any coding project!

Test Case 1

```
>>>
```

```
Enter a Temperature in Celsius:
```

```
30

Enter a Pressure in hPa:
1004

Enter a Mixing Ratio in g/kg:
8

Temperature (K): 303.14999
Specific Humidity: 7.93650839E-03
Pot. Temp. (K): 302.80426
Virtual Pot. Temp. (K): 304.27023
Diff. Theta from Theta_v (K): 1.4659729
>>>
```

Evaluation Criteria

Each of the following criteria will be rated from not present/completed to exemplary, having all of the elements will yield at least a 7/10 for a particular criterion. The assignment is out of 50 points.

- Required elements completed, coded efficiently, and styled well as listed in assignment description for Problem 1 (10 points)
- Required elements completed coded efficiently, and styled as listed in assignment description for Problem 2 (10 points)
- Required elements completed coded efficiently, and styled as listed in assignment description for Problem 3 (10 points)
- Code is well documented (informative comments/descriptions of code blocks; 10 points)
- Code compiles and runs without error (5 points)
- Code was regularly updated and committed to GitHub repository (5 points)