Problem Set 09
Selection Structures and Flowcharts
Problems

Instructions

1. This problem set contains paired programming and individual programming problems. Each problem has a set of deliverables that needs to be submitted. You are responsible for following the appropriate guidelines and instructions below. Create appropriately-named files as instructed.
2. Save all files to your Purdue career account in a folder specific to PS09.
3. Compress all deliverables into one zip file named **PS09_yourlogin.zip**. Submit the zip file to the Blackboard drop box for PS09 before the due date.

Deliverables List

Item	Type	Deliverable
Problem 1: Reynolds Number	Paired	PS09_reynolds_*yourlogin1_yourlogin2*.m PS09_reynolds_*yourlogin1_yourlogin2*_report.pdf Test Cases (submitted in Answer Sheet)
Problem 2: US Standard Atmosphere, 1976	Paired/ Individual	Flowchart (paired, submitted in Answer Sheet) Test Cases (submitted in Answer Sheet) PS09_atm_temp_*yourlogin*.m PS09_atm_temp_*yourlogin*_report.pdf Any data loaded into your code
Problem 3: Contact Lens Decision	Individual	PS09_contactlens_decision_*yourlogin*.m PS09_contactlens_*yourlogin*.m PS09_contactlens_*yourlogin*_report.pdf Any data loaded into your code PS04_stats_io_*yourlogin1_yourlogin2*.m
Answer Sheet		PS09_answer_sheet_*yourlogin*.docx

Answer Sheet

You must place your flowcharts and test cases in the Answer Sheet provided in the Assignment Files. The answer sheet is named PS09_answer_sheet_template.docx. You must resave it as **PS09_answer_sheet_*yourlogin*.docx** before submitting it.

- Complete the assignment information at the top of the answer sheet
- Your answer sheet will be used for all problems in this set
- Follow any additional instructions on the Answer Sheet
- Place items in the correct locations on the Answer Sheet

Problem Set 09
Selection Structures and Flowcharts
Problems

Problem 1: Reynolds number

Paired Programming

Problem Setup

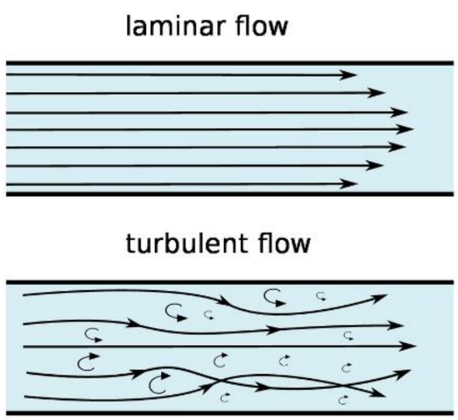

Figure 1: Flow examples

Reynolds number (Re) is a dimensionless quantity used to predict fluid flow patterns in particular situations. Process engineers use these predictions as a guide when designing a wide variety of systems. For instance, chemical and food process engineers use the Reynolds number when designing pipe systems for a production line.

For circular pipes, low Re values (Re<2300) indicate laminar flow, which is a flow regime where viscous forces dominate giving a smooth constant fluid motion. High Re values (Re>4800) indicate turbulent flow, which is a flow regime where inertial forces dominate to produce a chaotic fluid motion. Intermediate Re values indicate transitional flow, which means both laminar and turbulent flows are possible.

The formula used to calculate the Reynolds number (Re) is:

$$Re = \frac{inertial\ forces}{viscous\ forces} = \frac{\rho v D}{\mu}$$

Where:

ρ is the density of the fluid (kg/m^3)

v is the mean velocity of the fluid (m/s)

D is the diameter of the pipe (m)

μ is the dynamic viscosity of the fluid (Pa·s = N·s/m^2 = kg/(m·s))

To aid the design of piping systems for relatively low viscosity fluids at your food company, you are asked to write a user-defined function that returns the Reynolds number if given fluid density and viscosity, the fluid velocity in the pipe, and the pipe diameter. Your user-defined function should test for valid inputs and print appropriate, helpful error messages. The table below shows valid inputs for the variables.

Input	Minimum Valid Value (inclusive)	Maximum Valid Value (inclusive)
fluid density (kg/m^3)	0.5	1500 (e.g. syrup)
fluid velocity in the pipe (m/s)	0	10
pipe diameter (m)	0.05	0.2
fluid viscosity (Pa·s)	0.001	25 (e.g. syrup)

A flowchart has been approved by your supervisor for conversion to code. It is provided below.

Problem Set 09
Selection Structures and Flowcharts
Problems

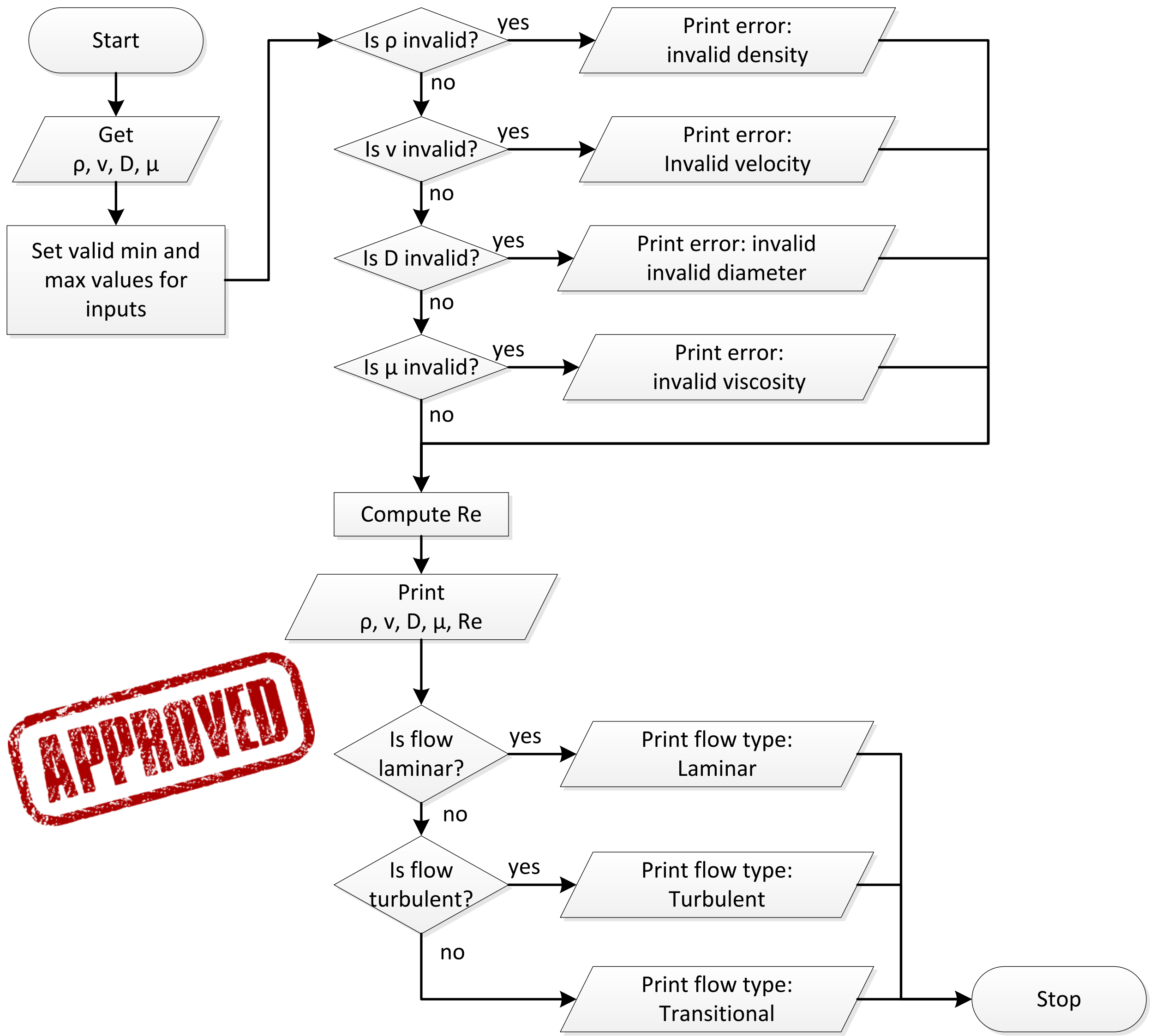

Figure 2: Reynolds number flowchart

Problem Steps

1. **Before you start to code**: Review the flowchart to understand the process for determining the Reynolds number. Note that error messages and the flow type are printed to the MATLAB Command Window.
2. In your Answer Sheet:
    a. Add a series of test cases to thoroughly examine all the possible paths in the flowchart.
    b. Record the corresponding flowchart outputs. These outputs must be determined independently of MATLAB (using the flowchart) so that they provide a means to check and debug your MATLAB code as you write the code.

Problem Set 09
Selection Structures and Flowcharts
Problems

3. Translate the flowchart to a MATLAB user-defined function. Name your UDF file **PS09_reynolds_*yourlogin1_yourlogin2*.m**. Your function must:
   a. Print useful errors to the Command Window.

      Hint: Learn to use the MATLAB built-in function `error` to display error messages. `error` is used rather than `fprintf` so that the user-defined function terminates properly when output arguments are not assigned.

   b. Return the Reynolds number when given valid values for fluid density, fluid viscosity, fluid velocity in the pipe, and pipe diameter.
   c. Print the type of flow to the Command Window when given valid values.
4. Test your function by calling it with the test cases you created in Step 2.a.
5. For each test case, paste the function call and results displayed in the Command Window as comments under the `COMMAND WINDOW OUTPUTS` section of your function file.
6. Publish your function to a PDF using any valid set of inputs and name the published file **PS09_reynolds_*yourlogin1_yourlogin2*_report.pdf**

Reference: http://www.cfdsupport.com/OpenFOAM-Training-by-CFD-Support/node263.html

Problem Set 09
Selection Structures and Flowcharts
Problems

Problem 2: US Standard Atmosphere, 1976

Paired Flowchart Creation

Individual Programming

Problem Setup

Knowledge of atmospheric temperature allows prediction of atmospheric conditions that can be valuable to many types of engineering applications, including aircraft performance, Mach number calculations, communication systems, and weather software.

The atmosphere has five main layers: the troposphere, stratosphere, mesosphere, thermosphere, and exosphere. Each main layer has an isothermic layer above it, called a pause. The temperature profile of the atmosphere is not uniform as altitude increases. Each layer and pause has its own temperature profile, and some layers have different temperature profiles within.

The US Standard Atmosphere 1976 is an idealized, steady-state atmospheric model that allows for predictions of atmospheric conditions at altitudes up to 1000 kilometers. It uses altitude ranges to distinguish between the temperature models.

For altitudes from 0 up to 86 kilometers, the temperature model for the US Standard Atmosphere is

$$T = T_b + L_b(H - H_b)$$

Where T_b is the base molecular-scale temperature in degrees Kelvin (K), H_b is the base height for the layer's altitude range in kilometers (km), and L_b is the temperature lapse rate, which is the rate of change in the temperature over a given altitude range, in K/km. The subscript b is a boundary indicator.

For this problem, you will be examining the temperature profile of the troposphere, tropopause, stratosphere, and stratopause. The coefficients for these atmospheric regions are listed in Table 1.

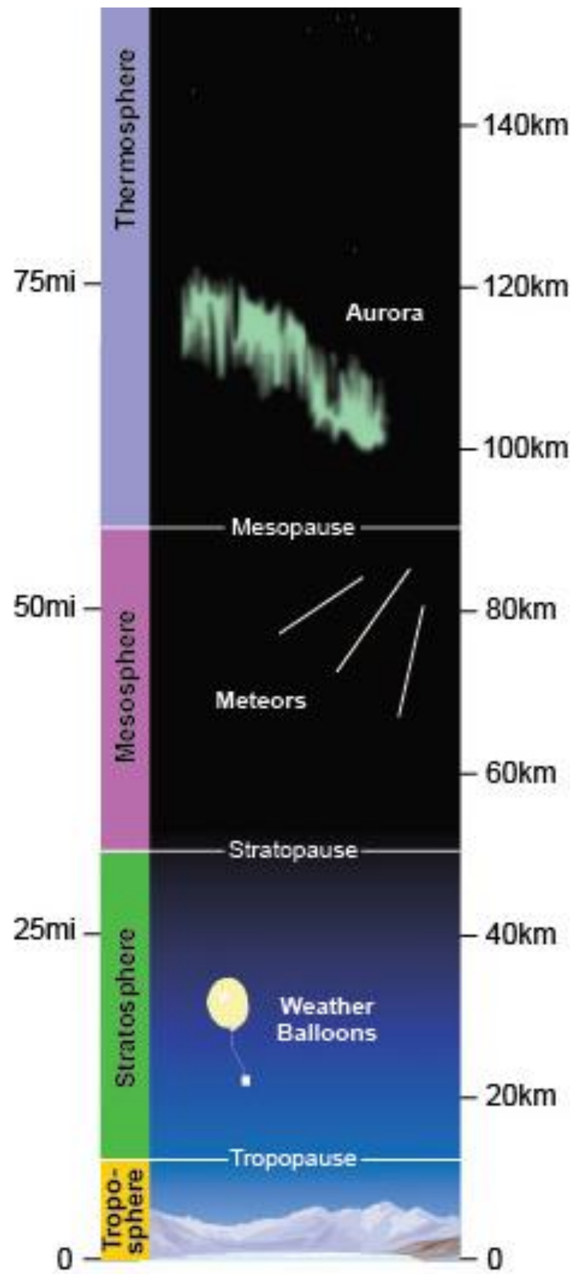

Figure 3: Atmospheric layers

Problem Set 09
Selection Structures and Flowcharts
Problems

Table 1. US Standard Atmosphere 1976 temperature-model coefficients for altitudes below 51 kilometers

Atmospheric Layer	Boundary Number	Boundary Range (km)	Base Temp, T_b (K)	Base Height, H_b (km)	Temp Lapse, L_b (K/km)
Troposphere	1	0 ≤ H < 11	288.15	0	-6.5
Tropopause	2	11 ≤ H < 20	216.65	11	0.0
Stratosphere	3	20 ≤ H < 32	216.65	20	1.0
	4	32 ≤ H < 47	228.65	32	2.8
Stratopause	5	47 ≤ H < 51	270.65	47	0.0

You have been provided with a data file that contains the base temperature (T_b), base height (H_b), and temperature lapse rate (L_b) from Table 1. The data file is named **Data_model_constants.csv**.

You need to create a MATLAB user-defined function that will use information from the US Standard Atmosphere 1976 to calculate the idealized temperature at any altitude below 51 kilometers **and** states where in the atmosphere the altitude is located. The function must:

- Accept altitude as the input argument
- Display appropriate, useful error messages to the Command Window for invalid inputs
- Return atmospheric layer (as a string) and temperature in degrees Kelvin as output arguments
- Calculate the atmospheric temperature using the coefficients provided in the data set
- Print the atmospheric layer and temperature to the Command Window for valid inputs

You will first create a flowchart and then translate it into MATLAB code.

Problem Steps

1. **Before you start to code**: Create a flowchart to outline how information should move through the code.
    a. Work with your paired partner to complete the flowchart.
    b. Draw a flowchart to:
        - Check for valid altitude inputs,
        - Correctly identify the atmospheric layer or pause, and
        - Calculate the temperature.
    c. You can draw the flowchart using any means that result in a clear image for the answer sheet. Make sure your flowchart is legible. Options include:
        - Drawing it by hand and taking a clear photo
        - Drawing it directly in the Word answer sheet using Word's drawing tools

Problem Set 09
Selection Structures and Flowcharts
Problems

- Drawing it in Microsoft's Powerpoint, Publisher, or Visio
- Using another flowchart tool

2. In your Word answer sheet:
   a. Paste the clear image of your flowchart.
   b. Select a series of test cases to thoroughly test all possible paths on your flowchart.
   c. Record the atmospheric layer or error for each test case.
3. Working individually, translate your flowchart into a one-input, two-output user-defined function named **PS09_atm_temp_*yourlogin*.m**.
4. Test your function using the test cases that you specified in Step 2.b.
5. For each test case, paste the function call and results displayed in the Command Window as comments under the `COMMAND WINDOW OUTPUTS` section of your function file.
6. Publish your function to a PDF using any valid input and name the published file **PS09_atm_temp_*yourlogin*_report.pdf**.

References
http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19770009539.pdf
http://www.srh.noaa.gov/jetstream/atmos/layers.html

Problem Set 09
Selection Structures and Flowcharts
Problems

Problem 3: Contact Lens Decision

Individual Programming

Problem Setup

In PS05 Problem 1, you were asked to create a user-defined function to determine the acceptability of contact lens designs. You were provided with a p-code file that accepted lens parameters and returned an acceptability decision. Reread the problem setup for PS05 to remind yourself of the original problem setup. Note that the problem set and the solutions are both available on Blackboard.

For this problem, you will create a UDF that performs the same job as the contactlens_decision.p code. The UDF requires six input arguments: lens design batch ID, mean base curve radius, standard deviation of base curve radius, mean diameter, standard deviation of diameter, and threshold. Output will be a binary (logical) decision variable, set to 0 if the batch of lenses is unacceptable and set to 1 if the batch of lenses is acceptable. When run, your code should print to the screen a message about the acceptability of <u>each lens</u> in this format:

```
Lens Design <batch ID> is <ACCEPTABLE/UNACCEPTABLE> at threshold ratio
<specified value>.
```

where `batch ID` refers to the lens design batch ID, and decision is either "`UNACCEPTABLE`" or "`ACCEPTABLE`" at a specified threshold as determined by your calculations for each lens design.

You have been provided with a new data set, named **Data_newlensdesigns.csv**, with four new lens designs. This is the data you will be testing with your new code. You will continue to use a threshold of ε = 0.02.

Problem Steps

1. Move all files for PS05 Problem 1 into your working folder for PS09. You will need these files to complete this problem, including the PS04_stats_io function. Run the PS05 files to remind yourself how they work. Check your code against the solution to debug it if it does not work as it should.
2. **Before you start to code:** Create a flowchart for ONLY your new contactlens_decision code (you do not need a flowchart for any other code in this problem).
    a. Draw a flowchart to indicate how the data will be processed by the code. You can draw the flowchart using any means that result in a clear image for the answer sheet. Make sure your flowchart is legible. See Problem 2 for drawing options.
3. In your Word answer sheet:
    a. Paste the clear image of your flowchart.
4. Save a new copy of your executive function from PS05 Problem 1, PS05_contactlens_*yourlogin*.m. Rename it as **PS09_contactlens_*yourlogin*.m**.
5. Write a user-defined function, named **PS09_contactlens_decision_*yourlogin*.m**, that accepts the same inputs as contactlens_decision.p and returns the same output argument and printed results.

Problem Set 09
Selection Structures and Flowcharts
Problems

6. Replace contactlens_decision.p with PS09_contactlens_decision_*yourlogin*.m in your executive function PS09_contactlens_yourlogin.m. Update any other relevant parts of the executive function.
7. Test and debug your code.
8. Publish your executive function to a PDF file **PS09_contactlens_*yourlogin*_report.pdf**.