RMIT University

School of Engineering

Advanced Mechatronics System Design – MANU2451

LabVIEW Test 2018

Student Name:

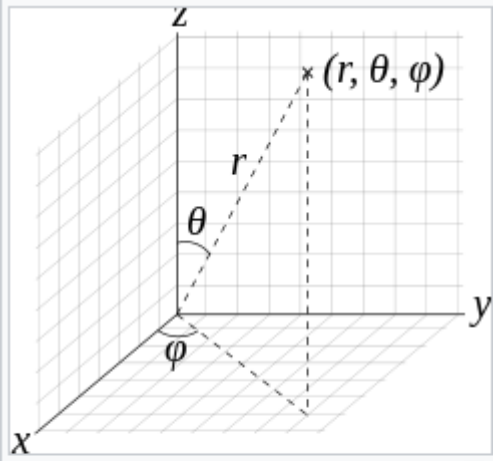
Student Number:

You have 60 minutes to finish this test and return this page at the end of the test period. If this page is not returned, you will not receive any mark for this assessment. After you finished testing your code, put all your files in a zip-folder, and then submit the zipped folder through Canvas → Assignments → Assessment 1 → LabVIEW Code Submission.

If you have any questions, please raise your hand. This is an open book assessment and you can use all available materials including Google. **However, this is an individual assignment and no form of communication with any other person (except your lecturer) is allowed during this test.**

Task 1: Please save and submit as "YourStudentNumber_Task1" (**5.5 Points**)

Apart from the Cartesian coordinate system, where the location of a point in space is represented by its (x, y, z) coordinate, we can also use the spherical coordinate system, in which the point is represented by the radial distance (r), polar angle (θ) and azimuthal angle (φ) as shown in the following figure.



Spherical coordinates (r, θ, φ) as commonly used in *physics* (ISO convention): radial distance r, polar angle θ (theta), and azimuthal angle φ (phi). The symbol ρ (rho) is often used instead of r.

Your task is to create a program to read in (x, y, z) values and then translate them into (r, θ, φ) values.

Write a program which allows you to:

a. Key in three numbers for x, y, z. (think "control") – **0.25 point**

b. Calculate the radial distance, r. – **1.5 points**

c. Calculate the polar angle, θ. – **1.5 point**

d. Calculate the azimuthal angle, φ. – **1.5 point**

e. Display r, θ, and φ. – **0.5 point**

f. The program has to run continuously until you press stop. – **0.25 points**

Task 2: Please save and submit as "YourStudentNumber_Task2" (4 **Points**)

You are going to simulate the home monitoring system in this task, by using AND or OR gates.

Imagine that your home has three rooms (A, B and C). Also imagine that the monitoring system has two simple settings:

- Setting 1: On – As long as any sensor in rooms A, B or C detects any human motion, the alarm would go off.
- Setting 2: Off – Alarm will not go off.

Here are the tasks in detail:

a. First, bring in a "structure" which will repeat the iterations until you press stop. – **0.25 points.**

b. Add in three Boolean controls (push button), and label them "room A", "room B", and "room C" respectively. – **0.25 points**

c. Add in one Boolean indicator (LED) and label it "alarm". – **0.25 point**.

d. Add in one numerical control and label it "setting". – **0.25 point**.

e. Find an icon to compare the "setting" with 1. If the setting is 1, then the alarm will turn on if either room A OR room B OR room C is switched on. – **2.5 points**

f. If the setting is 2, then the alarm shall be always off. – **0.5 point**.

Remember, you should use only AND or OR gates for tasks e and f.

Task 3: Please save and submit as "YourStudentNumber_Task3" (**4 Points**)

In this task, you will calculate the sum of all odd numbers in the range of 0 to 100.

- Bring in a structure which will run 100 iterations. **– 0.5 points**
- Find a function within the "numeric" palette which can give you a quotient and remainder from division. **– 0.5 points**
- Think of a way to use the above function to determine if the iteration number "i" is an odd number. **– 2 points**
- Cumulatively add all the odd numbers and display the sum in a numeric indicator. **– 1 points.**

Task 4: Please save and submit as "YourStudentNumber_Task4" (**6.5 Points**)

In this task, you will first create a "noisy" sine wave, by adding a sine wave with a random number. After that, you will use a "moving average" to filter the signal.

a. First, bring in a "structure" which will repeat the iterations until you press stop. – **0.25 points.**

b. Let the iteration run every 0.1s. – **0.25 points.**

c. Within this structure, create a signal y1 = sin(2π*t). **– 1 points**
Hint: you can use iteration "i" as a source of time "t". However, you may need to rescale this "i" to match the actual time. This has been discussed in class before.

d. Create a second signal y2 = random number from -0.5 to +0.5. **– 1 points**
Hint: the random number from LabVIEW gives 0 to 1. You need to do something to bring it to -0.5 to +0.5.

e. Add y1 and y2 together. **– 0.5 points.**

f. Use shift register(s) to hold values of (y1+y2) including:

- 1 step earlier.
- 2 steps earlier
- 3 steps earlier
- 4 steps earlier **– 1 point.**

g. Calculate the sum of (y1+y2) over the most current step, 1 step earlier, 2 steps earlier, 3 steps earlier and 4 steps earlier. Then divide the sum by 5 to get the average **– 1.5 point**

h. Use a chart to display the noisy signal (y1+y2) which you obtained in e, along with the filtered signal which you calculated in g. **– 1 point**