CSE003 Individual Project
Coursework Assessment Specification

2019/20 Semester 1
Bachelor Degree – Year 1

Module Code	Module Leader	Module Title
CSE003	**Hai-Ning Liang**	**Fundamentals of Computer Programming**

Coursework Assessment Number:	**2 of 4**
Method of Working:	**Individual**
Coursework Title:	**Number Systems and Algorithms**
Percentage (%) Weighting:	**20% of the overall module marks**
Workload Guidance:	**Typically, you should expect to spend between 25 and 40 hours on this individual assessment**
Date and time of publication:	**6 pm on Friday, Week 10 (15 November 2019)**
Date and time for submission:	**9 pm on Tuesday, Week 12 (26 November 2019)**

General Instructions

1. One copy of this assessment should be handed via the module **ICE** page at http://www.ice.xjtlu.edu.cn no later than the time and date shown above, unless an extension has been authorised by the module leader.

2. Before submission, each student must complete module coursework submission form obtainable from the module ICE page. This assessment is being marked by student name and id, please ensure that you complete the correct submission form.

3. Format of the Coursework Assessment Submission:
   A **ZIP** file submitted via the ICE module page containing the deliverables outlined in the "**What to Submit**" section of the coursework assessment specification.

4. Use of Unfair Means:
   You are reminded of the University's Code of Practice on the Use of Unfair Means and that the work you submit for assessment should contain no section copied in whole or in part from any other source unless where explicitly acknowledged by means of proper citation.

5. Late penalties:
   For work submitted late the penalty is loss of **5%** marks per day. Work that is **5** or more days late will automatically be graded as **FAIL**, and no re-submission will be allowed.

Part - A (24 Marks)

Number Systems

A1. Covert the following decimal numbers to their binary equivalents using repeated division-by-2 method. You must demonstrate step-by-step implementation of your answers to obtain full marks.

(4 Marks)

(i) 59_{10}

(ii) 2473_{10}

A2. Covert the following binary numbers to their decimal equivalents. You must demonstrate step-by-step implementation of your answers to obtain full marks.

(4 Marks)

(i) 11011_2

(ii) 1001101110011101_2

A3. Complete the following binary addition problems. You must demonstrate step-by-step implementation of your answers to obtain full marks.

(4 Marks)

(i) $110_2 + 101_2$

(ii) $10101011_2 + 11011100_2$

A4. Complete the following binary substraction problems. You must demonstrate step-by-step implementation of your answers to obtain full marks.

(4 Marks)

(i) $100011_2 - 11010_2$

(ii) $100100101_2 – 111010_2$

A5. Complete the following octal numbers to their hexadecimal eqivalents using binary system method. You must demonstrate step-by-step implementation of your answers to obtain full marks.

(4 Marks)

(i) 361_8

(ii) 715_8

A6. Complete the following hexadecimal numbers to their octal eqivalents using binary system method. You must demonstrate step-by-step implementation of your answers to obtain full marks.

(4 Marks)

(i) $4E7_{16}$

(ii) $3F9B05D_{16}$?

Part - B (16 Marks)

Logical Operators

Assume that A = False, B = False, and C = True. Find whether the following expressions are false or true. You must demonstrate step-by-step implementation of your answers to obtain full marks.

B1. C && !((C && B) || A) && A && !B (4 Marks)

B2. !(B && A) || (A && !B) && (!C || A) && !(!A) && !(B || C) (4 Marks)

B3. !(!(A && (B || C && (!A && (B && !C))))) (4 Marks)

B4. !B && !A && (C || A) || !B || (A && C || (!A && !C)) (4 Marks)

Part - C (60 Marks)

Algorithm, Pseudocode and Flowchart

C1. Draw a flowchart for your algorithm to print the count of digits of a given positive number.

(5 Marks)

For example:
Enter the number 1674
Digit Count of 1674 is: 4

C2. Write an algorithm to print the factorial number of given terms. (5 Marks)

For example:
Enter Number of Terms: 5
Factorial of 5: 5! = 5*4*3*2*1 = 120

C3A. Write an algorithm to print a Fibonacci series of given terms.

(5 Marks)

For example:
Enter Number of Terms: 7
Fibonacci series of 7: 0, 1, 1, 2, 3, 5, 8

C3B. Draw a flowchart for your algorithm to print a Fibinocci series of given terms.

(5 Marks)

C4A. Write an algorithm to print a Padovan sequence of given terms.

(5 Marks)

For example:
Enter Number of Terms: 12
Fibonacci series of 12: 1， 1， 1， 2， 2， 3， 4， 5， 7， 9， 12， 16

C4B. Draw a flowchart for your algorithm to print a Padovan sequence of given terms.

(5 Marks)

C5A. Gross pay depends on the pay rate and the number of hours worked per week. However, if you work more than 40 hours, you get paid time-and-a-half for all hours worked over 40. Write a pseudocode to compute gross pay for given pay rate and hours worked.

(5 Marks)

C5B. Draw a flowchart for your pseudocode to display the weekly wages.

(5 Marks)

C6A. Write your own pseudocode to calculate your average quiz marks.

(5 Marks)

C6B. Draw a flowchart for your pseudocode to display the average quiz marks.

(5 Marks)

C7. In this section you need to write a pseudocode and draw a flowchart for a robot that can turn itself in the desired direction.

Let's assume you have given a control of a robot that can measure its distance to the wall (see Fig. 1), and it can rotate-in-place clockwise or counter-clockwise 1° each time. Now your robot is close to the wall and needs to turn left towards exactly East in the Figure 1.

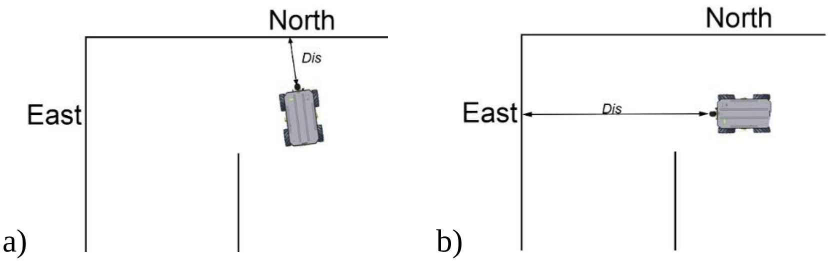

Figure 1: the robot car is performing in-place-turning towards left where a) is some previous state of the robot and b) is target state of the robot.

The robot can call the following commands:

- **RotateLeft**, which rotates the robot counter-clockwise 1° .
- **RotateRight**, which rotates the car clockwise by 1°.
- **GetDistance**, which finds the car's distance to the wall.

You can use all commands to help the robot rotate itself towards exact East direction.

Note: The robot has just started rotating and it is now not exactly facing North (see Fig.1a), and you have no accessible data of how many degrees are the offset.

C7A. Write a pseudocode to let your robot turn left facing towards exact East.

(5 Marks)

C7B. Draw a flowchart for your pseudocode to rotate your robot towards East.

(5 Marks)

What to Submit

Your **ZIP** file should include these files:

- Your completed submission form (properly completed and with your signature). The submission form will be made available on ICE (**submission form with incorrect information certainly will affect your marks, so carefully complete the submission form**). The submission form should be properly named as mentioned below.
- One **PDF** file with all your answers. Make sure it can be open and runs properly on a **Windows** computer. The **PDF** file should be properly named as mentioned below.
- Use your **First Name**, **Last Name**, and **Student Number** to name your PDF file, Submission Form, and ZIP file—for example **Liang_Haining_999999** will be the name of the files module leader would be submitting, with 999999 being as his student number (any submissions with improper or incomplete file names certainly will affect your marks, so carefully name your files).

NOTE: To create ZIP files in Windows, see these two links:

- http://windows.microsoft.com/en-US/windows-8/zip-unzip-files
- https://support.microsoft.com/en-hk/help/14200/windows-compress-uncompress-zip-files

The end of the document
***