EXAM PAPERS PRACTICE

5.1 Number systems

Name: ______________________

Class: ______________________

Date: ______________________

Time: **115 minutes**

Marks: **85 marks**

Comments:

Q1.

(a) Three numbers are listed in the first column of **Table 1**.

For each row in **Table 1**, shade one or more lozenges, in the appropriate column(s), to indicate which set(s) of numbers contain(s) the number on the row.

As an example, the first row has been completed for you, to indicate that π is a member of the set of irrational numbers and the set of real numbers, but is not a member of the sets of natural, integer or rational numbers.

Table 1

	Natural	Integer	Rational	Irrational	Real
π	⬭	⬭	⬭	⬬	⬬
15/23	⬭	⬭	⬭	⬭	⬭
108	⬭	⬭	⬭	⬭	⬭

(2)

(b) **Table 2** shows a list of eight numbers, stored in an array.

Table 2

Index	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
Contents	48	9	201	62	82	92	30	72

Describe what an ordinal number is **and** what an ordinal number would be used for in the context of this array.

____

____

____

____

(2)

(Total 4 marks)

Q2.

$\mathbb{R}$ denotes the set of real numbers, which includes the natural numbers, the rational numbers and the irrational numbers.

(a) Give **one** example of a natural number.

____

(1)

(b) Give **one** example of an irrational number.

____

(1)
(Total 2 marks)

Q3.

A particular computer uses a **normalised** floating point representation with an 8-bit mantissa and a 4-bit exponent, both stored using **two's complement**.

Four bit patterns that are stored in this computer's memory are listed in the figure below and are labelled **A, B, C, D**. Three of the bit patterns are valid floating point numbers and one is not.

	Mantissa	Exponent
A	0 ● 1 0 1 1 1 0 0	1 0 1 0
B	0 ● 0 0 1 1 0 1 0	0 1 0 1
C	1 ● 0 0 0 0 0 0 0	0 1 1 1
D	1 ● 0 1 1 1 1 1 1	1 0 0 0

(a) Complete the table below. In the Correct letter (**A-D**) column shade the appropriate lozenge **A, B, C or D** to indicate which bit pattern from above is an example of the type of value described in the Value description column.

Do **not** use the same letter more than once.

Value description	**Correct letter (A-D)**
A positive normalised value	A ○ B ○ C ○ D ○
The most negative value that can be represented	A ○ B ○ C ○ D ○
A value that is not valid in the representation because it is not	A ○ B ○ C ○ D ○

normalised	

(3)

(b) The following is a floating point representation of a number:

0	• 1	0	1	1	0	0	0

0	1	0	1

Mantissa Exponent

Calculate the decimal equivalent of the number. Show how you have arrived at your answer.

____

____

Answer ____________________________

(2)

(c) Write the normalised floating point representation of the negative decimal value -6.75 in the boxes below. Show how you have arrived at your answer.

____

____

____

____

Answer:

	•						

Mantissa Exponent

(3)

(d) An alternative two's complement format representation is proposed. In the alternative representation 6 bits will be used to store the mantissa and 6 bits will be used to store the exponent.

Existing Representation (8-bit mantissa, 4-bit exponent):

	•						

Mantissa Exponent

Proposed Alternative Representation (6-bit mantissa, 6-bit exponent):

	•				

Mantissa Exponent

Explain the effects of using the proposed alternative representation instead of the existing representation.

____

____

____

____

(2)

(Total 10 marks)

Q4.

A normalised floating point representation uses an 8-bit mantissa and a 4-bit exponent, both stored using **two's complement format**.

(a) In binary, write the largest positive number that can be represented using this normalised floating point system in the boxes below:

•							

Mantissa Exponent

(2)

(b) This is a floating point representation of a number:

0 •	1	0	1	1	0	0	0

0	0	1	1

Mantissa Exponent

Calculate the denary equivalent of the number. Show how you have arrived at your answer.

Working ____

____

____

(1)

Answer ____

(1)

(c) This is a floating point representation of a number:

1 •	0	1	1	0	0	0	0

Mantissa

1	1	0	0

Exponent

Calculate the denary equivalent of the number. Show how you have arrived at your answer.

Working ____

____

____

(1)

Answer ____

(1)

(d) Write the normalised floating point representation of the **negative** denary value -108 in the boxes below. Show how you have arrived at your answer.

Working ____

____

____

____

(2)

Answer

•							

Mantissa

Exponent

(1)

(e) (i) In the context of floating point representation, explain what overflow is.

____

____

____

____

(2)

(ii) The table below contains descriptions of operations which may or may not cause an overflow error when they are carried out with a floating point representation.

Place **one** tick next to the operation that may cause overflow.

Operation	May cause overflow? (Tick one box)
Subtracting a very small number from a large number.	
Dividing a large number by a very small number.	
Multiplying a large number by a very small number.	

(1)
(Total 12 marks)

Q5.

A particular computer uses a **normalised** floating point representation with an 8-bit mantissa and a 4-bit exponent, both stored using **two's complement**.

(a) Four bit patterns that are stored in this computer's memory are listed in **Figure 1** and are labelled with the letters **A** to **D**. Three of the bit patterns are valid floating point numbers and one is not.

Figure 1

A

Mantissa									Exponent			
0 •	1	0	0	0	0	0	0		1	0	0	0

B

Mantissa									Exponent			
0 •	1	0	0	0	0	0	0		1	1	1	1

C

Mantissa									Exponent			
0 •	0	1	0	1	0	0	0		0	1	1	0

D

Mantissa									Exponent			
1 •	0	1	0	1	0	0	0		0	1	0	1

Mantissa Exponent

Complete **Table 1** below. In the **Correct letter (A–D)** column write the appropriate letter from **A** to **D** to indicate which bit pattern in **Figure 1** is an example of the type of value described in the **Value description** column.

Do **not** use the same letter more than once.

Table 1

Value description	Correct letter (A-D)
A negative value.	
The smallest positive value that can be represented.	
A value that is not valid in the representation because it is not normalised.	

(3)

(b) This is a floating point representation of a number.

0 ●	1	0	1	0	0	0	0

Mantissa

0	1	1	0

Exponent

Calculate the denary equivalent of the number. Show how you have arrived at your answer.

Working: ____

____

____

(1)

Answer: ____

(1)

(c) Write the normalised floating point representation of the negative denary value –7.75 in the boxes below. Show how you have arrived at your answer.

Working: ____

____

____

____

(2)

Answer: ____

•							

Mantissa Exponent

(1)

(d) There can be a loss of precision when a denary number is stored using this floating point system.

The closest possible representation of the denary number 6.9 is shown below.

0 •	1	1	0	1	1	1	0

0	0	1	1

Mantissa Exponent

By converting this bit pattern back into denary it can be seen that the actual number stored is 6.875, not 6.9.

(i) Calculate the absolute error that has occurred.

(1)

(ii) Calculate the relative error that has occurred.

(1)

(iii) Explain how the floating point system used could be modified to allow a more accurate representation of 6.9.

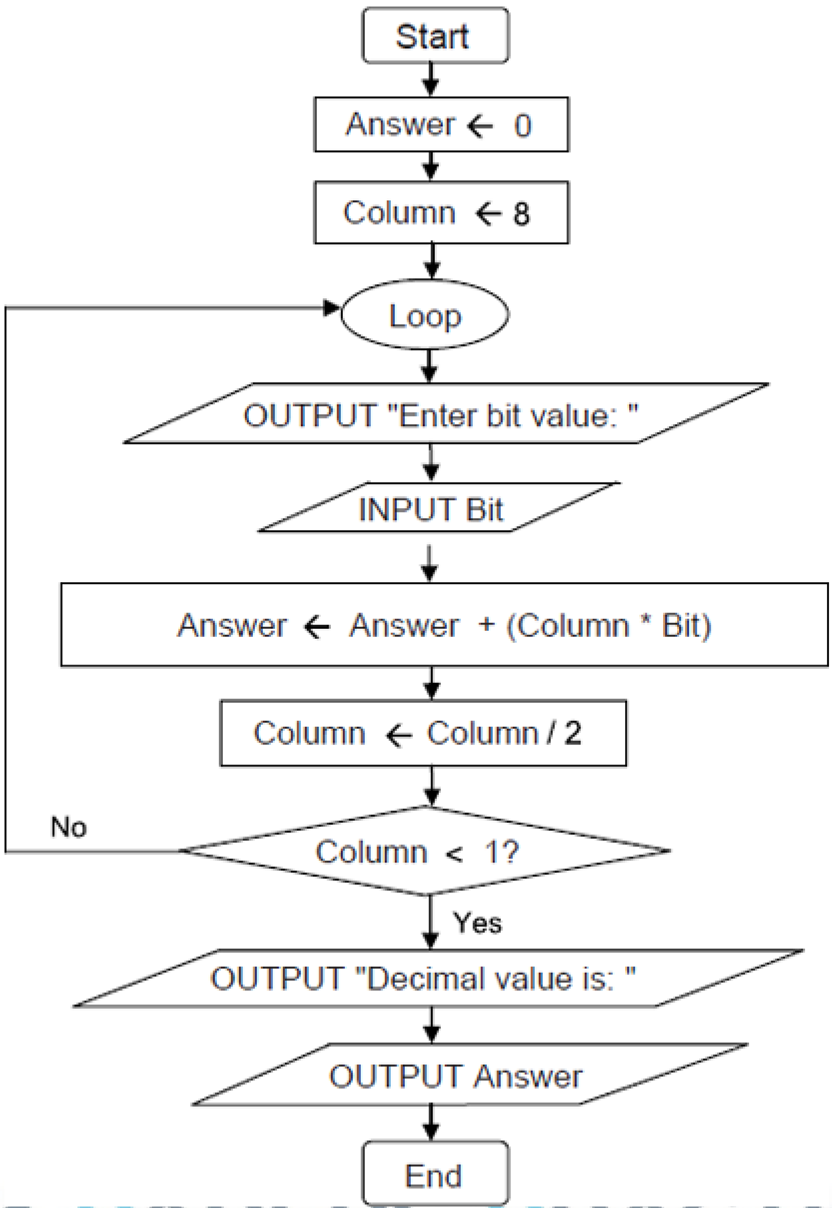
(2)

(Total 12 marks)

Q6.

Create a folder/directory for your new program.

The algorithm, represented as a flowchart below, and the variable table, describe the converting of a 4-bit binary value into denary.



Identifier	Data type	Purpose
`Column`	`Integer`	Stores the place value (column heading)
`Answer`	`Integer`	Stores the denary value equivalent to the bit pattern entered by the user
`Bit`	`Integer`	Stores a `0` or `1` entered by the user

What you need to do

Write a program for the above algorithm.

Test the program by showing the result of entering the values `1, 1, 0, 1 (in that order).`

Save the program in your new folder/directory.

Evidence that you need to provide

(a) Your PROGRAM SOURCE CODE.

(11)

(b) SCREEN CAPTURE(S) for the test described above.

(3)

(c) What is the largest denary number that could be output by the algorithm represented by the flowchart in the diagram above?

____

(1)

(d) The algorithm represented by the flowchart above can convert sixteen different bit patterns into denary.

If the symbol $\boxed{\text{Column} \longleftarrow 8}$ is changed to $\boxed{\text{Column} \longleftarrow 16}$ how many **more** bit patterns could be converted into denary?

____

(1)

(e) When developing a new system the stages of the systems development life cycle could be followed.

At which stage of the systems development life cycle would the flowchart above have been created?

____

(1)

(f) At which stage of the systems development life cycle would the algorithm represented by the flowchart above be automated using a programming language?

____

(1)

(Total 18 marks)

Q7.

(a) Represent the denary number 123 in binary using 8 bits.

Answer ____

(1)

(b) How many different denary numbers can be represented using 8-bit binary?

Answer ____

(1)

(c) What is the hexadecimal equivalent of the denary number 123?

Answer ____

(2)

(d) Why are bit patterns often displayed using hexadecimal instead of binary?

____

____

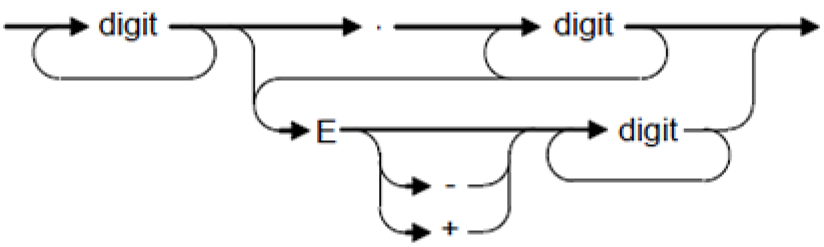
(1)

(Total 5 marks)

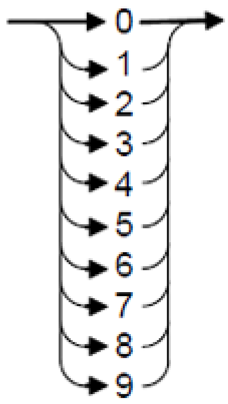
Q8.

In a particular programming language, the correct syntax for a real number is defined by the syntax diagrams in the diagram below.

real number:



digit:



(a) Write **Yes** or **No** in the spaces in the empty column of the table below to identify whether or not the numbers listed in the table are valid real numbers which conform to the correct syntax for this language.

Real number	**Valid? (Yes / No)**
203.412	
-12.87	
12.43E-12	

(3)

(b) In the same language:

A *digit* is defined as any single numeric symbol from this list: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
A *whole number* is defined as a sequence of one or more *digits*.
An *integer* is defined as a *whole number* or a + or a – symbol followed by a *whole number*.

Write Backus-Naur Form (BNF) production rules for *digit*, *whole number* and *integer*.

(3)
(Total 6 marks)

Q9.
The figure below shows three different programs which have been developed using different generations of programming language.

Program 1

```
If Sales > 10000
  Then BonusPayment :=True
etc.
etc.

Procedure InputNewData
Procedure ToOutputFile

etc ...
```

Program 2

```
Move    #0,  R1
Add     R1,  R2
Store   R1,  0197
Move    0198, R3
Add     R2,  R3
Cmp     R3,  #1662
Bne     0988

etc ...
```

Program 3

```
1000 0101
1010 1111
1010 1111
1110 0001
1010 1111

etc ...
```

The above programs were written for different tasks.

(a) What generation of programming language was used for Program 1?

(1)

(b) Indicate which program was most likely to have been written for:

(i) controlling a new hardware device.

____

(1)

(ii) a payroll application.

____

(1)

(c) Program 1, Program 2 and Program 3 may require translation before each can be executed.

	Assembler	Compiler	None
Program 1			
Program 2			
Program 3			

Put **one** tick on each row in the table above to indicate the translator software required.

(3)

(d) Describe how **interpreter** software enables a program written in a high level language to be executed.

____

____

____

(2)

(e) A friend gives you a copy of a freeware **assembler**. Why might you not be able to use this successfully on your computer?

____

____

(1)

(Total 9 marks)

Q10.

A programming language has two different data types for storing positive integers.

Data type `Integer1` uses a single byte to store data.

Data type `Integer2` uses two consecutive bytes to store data.

(a) The program statement below defines a variable `NoOfAccidents`.

```
Var NoOfAccidents : Integer1 ;
```

What is the largest value which can be assigned to `NoOfAccidents`?

____

(1)

(b) Two more program statements are:

```
Var JourneyMileageA : Integer1 ;

Var JourneyMileageB : Integer1 ;
```

Interpreter software uses address 600 for storing a value for JourneyMileageA. See **Figure 1.**

Figure 1

Address	Contents
600	0101 0001
601	1010 1010
602	1111 1100
603	
604	
~	~
~	~
700	0000 0010
701	0000 1010
702	
703	

(i) State the **denary value** for the stored binary value.

`JourneyMileageA =`____

(1)

(ii) The program statement:

```
JourneyMileageB := 138 ;
```

stores the data value for `JourneyMileageB` at address 603.

What **binary value** will be stored at location 603?

(1)

(c) Another program statement is:

```
Var TotalMileage : Integer2 ;
```

The interpreter software uses locations 700 and 701 to store a value for `TotalMileage` with the most significant byte stored at location 700. See **Figure 1.**

What is the **denary value** assigned to `TotalMileage`?

(1)

(d) Programs also work with character data.

ASCII Code Table

Character	Decimal	Character	Decimal	Character	Decimal
<space>	32	I	73	R	82
A	65	J	74	S	83
B	66	K	75	T	84
C	67	L	76	U	85
D	68	M	77	V	86
E	69	N	78	W	87
F	70	O	79	X	88
G	71	P	80	Y	89
H	72	Q	81	Z	90

(i) Using the ASCII code table shown above, what is the **7-bit binary ASCII** code for character 'B'?

(1)

(ii) When a parity bit is included, character codes are stored as 8-bit binary numbers where the most significant bit is a parity bit. This system will use **even parity.**

Describe how the parity bit is used during data transmission of a single character.

(2)
(Total 7 marks)