3.1.3 Real Numbers and Normalised Floating Point Representation

Oct/NOV 2003

6. Using an 8 bit byte for the mantissa (fraction) and another 8 bit byte for the exponent (characteristic)

(a) Show

(i) $10\frac{3}{4}$
(ii) $-10\frac{3}{4}$

as 2 byte, normalised, floating point numbers. **[4]**

(b) Show the bit pattern that represents

(i) the largest positive
(ii) the smallest magnitude negative

number that can be represented using this 2 byte normalised floating point form. **[4]**

Oct/NOV 2004

6. (c) Explain how accuracy can be improved in a floating point representation and state an effect it can have on the number represented. **[3]**

May/June 2005

7. (c) (i) Convert −63 and −94 into 2's complement, 8 bit, binary numbers. **[2]**

(ii) Add the binary values obtained in (i) together. **[2]**

(iii) Comment on the result that you obtained in (ii). **[2]**

Oct/NOV 2006

5(b) A particular computer stores numbers as 8 bit, two's complement, binary numbers.

01011101 and 11010010 are two numbers stored in the computer.

(i) Write down the decimal equivalent of 11010010. **[2]**
(ii) Add the two binary values together and comment on your answer. **[3]**

3.1.3 Real Numbers and Normalised Floating Point Representation

May/June 2007

9. A computer stores numbers in floating point form, using 8 bits for the mantissa and 8 bits for the exponent. Both the mantissa and the exponent are stored in two's complement form.

(a) Explain the effect on the

- range
- accuracy

of the numbers that can be stored if the number of bits in the exponent is reduced. **[4]**

(b) Give the denary number which would have `01000000 00000000` as its binary, floating point representation in this computer. **[2]**

(c) Explain why it is not possible to represent zero as a normalised floating point number. **[2]**

Oct/NOV 2007

10. A computer stores fractional numbers in floating point binary representation. Five bits are used for the mantissa and three bits for the exponent. All values are stored in two's complement form.

Mantissa					Exponent		

(a) By using a diagram of this representation, state the value of each of the bits. **[4]**

(b) By using $2\frac{1}{2}$ as an example, explain how real numbers can be shown in normalised form in this representation. **[3]**

(c) State the floating point binary value of $-\frac{3}{4}$ in this representation. **[2]**

May/June 2009

6. (b) The denary number $10\frac{3}{4}$ Is to be represented as a floating point binary number using 12 bits.

The first 8 bits are to be used for the mantissa and the remaining four bits are to be used for the exponent.

(i) Explain what is meant by the mantissa of a floating point number. **[2]**

(ii) Explain what is meant by the exponent of a floating point number. **[2]**

(iii) Show why `001010110101` is a floating point representation of $10\frac{3}{4}$. **[3]**

(iv) Normalise the floating point value given in (iii). **[2]**

3.1.3 Real Numbers and Normalised Floating Point Representation

Oct/NOV 2009. P32

9. (b) (i) Work out the answer to the following binary addition sum. (All values are given in two's complement form. You should show your working.)

```
 01001101
 00101011
 01000101 +
 ________
```

[2]

(ii) Explain why the binary result does not give the correct answer. [1]

Oct/NOV 2010. P31/ P32

7. A computer stores floating point numbers using 2 eight-bit bytes.

The first byte is used to store the mantissa and the second stores the exponent.

(a) The normalised form of the floating point representation of $9\frac{1}{2}$ is

0100110000000100.

(i) Explain this representation of $9\frac{1}{2}$. [4]

(ii) Give the floating point value of $22\frac{1}{4}$ using this representation. [2]

(b) It is decided to change the representation by using 10 bits for the mantissa and 6 bits for the exponent. Explain the effect of this decision on the range and accuracy of the data represented. [4]

Oct/NOV 2010. P33

7 . Floating-point numbers in a particular computer system are stored using 12 bits. The first 6 bits are used for the storage of the mantissa and the second set of 6 bits is used to store the exponent.

(a) One way to represent 6.5 as a floating-point number in this representation is

001101000100

Explain why this representation is equivalent to 6.5 [4]

(b) (i) Using the representation above to help you, write the number 6.5 as a floating-point number in normalised form. [2]

(ii) Explain the effects of changing the representation so that 8 bits are used for the mantissa and 4 bits for the exponent. [2]

(c) The numbers 011011001101 and 101100001110 are stored with 6 bits for the mantissa and 6 for the exponent. Add the exponents of the two floating-point numbers together. [2]

3.1.3 Real Numbers and Normalised Floating Point Representation

Oct/NOV 2011. P32

3 (b) A particular computer uses two 8-bit bytes to store floating-point values. One byte is used to store the mantissa and the other is used to store the exponent.

(i) Write down, in binary form, the largest positive value that can be stored using this representation. **[2]**

(ii) Write down, in binary form the smallest magnitude, negative number that can be stored in this representation. **[2]**

(iii) The value `01101000 11111101` is stored as a floating-point number in this computer.
State what denary number is being represented, explaining how you arrived at your answer. **[4]**

Oct/NOV 2011. P33

3. (c) A particular computer uses a single 10-bit word to store a floating-point representation of a number.

The first 6 bits are used to store the mantissa and the remaining 4 bits are used to store the exponent.

(i) Explain why `000101 0100 = 2`
using this notation. **[2]**

(ii) Rewrite the binary value of this floating-point representation so that it is in normalised form. **[2]**

(iii) `011001 0011` is a normalised floating-point number.
By converting each of the mantissa and the exponent into a denary number first, write this number in denary. **[3]**

May/June 2012. P31/32

2 A binary pattern can be used to represent different data used in a computer system.

(a) Consider the binary pattern: `0101 0011`
The pattern represents an integer.
What number is this in denary? **[1]**

(b) Consider the binary pattern: `0001 0101 0011`
The pattern represents a Binary Coded Decimal (BCD) number.
What number is this in denary? **[1]**

(c) Consider the binary pattern: `1001 0010`
This represents a two's complement integer.
What number is this in denary? **[1]**

(d) Floating point is to be used to represent real numbers with:

- 8 bits for the mantissa, followed by
- 4 bits for the exponent
- two's complement used for both the mantissa and the exponent

(i) Consider the binary pattern:

3.1.3 Real Numbers and Normalised Floating Point Representation

0	1	1	0	1	0	0	0		0	1	0	0

What number is this in denary? Show your working. **[3]**

(ii) The representation shown in part (d)(i) is normalised.

Explain why floating point numbers are normalised. **[1]**

(iii) Show the binary pattern for the smallest positive number which can be stored using a normalised 12-bit floating point representation.

Mantissa:

Exponent:

Work out its denary value.

Denary: .... **[3]**

(e) The developer of a new programming language decides that all real numbers will be stored using 20-bit normalised floating point representation. She cannot decide how many bits to use for the mantissa and how many for the exponent.

Explain the trade-off between using either a large number of bits for the mantissa, or a large number of bits for the exponent. **[2]**

May/June 2012. P33

2. (c) Real numbers are to be stored in floating point representation with:

- 8 bits for the mantissa, followed by
- 4 bits for the exponent
- two's complement used for both the mantissa and the exponent

(i) Consider the binary pattern:

1	0	1	0	1	0	0	0		0	1	1	1

What number is this in denary? Show your working. **[3]**

(ii) Explain how you can recognise that the above pattern is normalised. **[1]**

(iii) Show the binary pattern for the smallest negative number (negative sign and large magnitude) which can be stored using a normalised 12-bit floating point representation.

Mantissa:

Exponent:

Work out its denary value.

Denary: ………………………………………………………………… **[3]**

Oct/NOV 2012. P32

2. (b) A computer system stores real numbers using a 12-bit floating point representation.

The first 8 bits are the mantissa and the final 4 bits the exponent. Both the mantissa and the exponent use two's complement representation.

Consider the binary pattern:

1	0	0	0	1	0	0	0	0	1	1	1

(i) What is the mantissa in denary? **[1]**

(ii) What is the exponent in denary? **[1]**

(iii) What real number is being represented? Show your working. **[2]**

(iv) Show the binary pattern for the largest and smallest positive numbers which can be represented with this 12-bit floating point representation.

Largest positive number in binary:

Smallest positive number in binary:

[4]

Oct/NOV 2012. P33

2. (b) Two integers are represented as 8-bit two's complement numbers. The numbers are to be added.

1	1	0	0	1	1	0	0	
1	0	0	0	0	1	1	1	+

(i) Show the result (in binary) in the table above. **[2]**

(ii) Comment on the result. **[1]**

(c) A computer system stores real numbers in floating point format using 12 bits.

The first 8 bits are the mantissa and the final 4 bits the exponent.

Both the mantissa and the exponent use two's complement format.

3.1.3 Real Numbers and Normalised Floating Point Representation

Consider the binary pattern `0101 1000 0101`

(i) What is the exponent in denary? **[1]**

(ii) What real number is being represented? (Show your working.) **[2]**

May/ June 14.P31/P32

5. (d) Many computer systems need to store and process real numbers.

A computer uses two bytes to store a real number. The first (Byte 7) stores the mantissa and the second (Byte 8), the exponent. Both mantissa and exponent use two's complement.

(i) What denary number is represented by Byte 7 and Byte 8?

Byte 7								Byte 8							
0	1	1	0	1	0	0	0	0	0	0	0	0	0	1	1

Show your working. **[3]**

(ii) Without any working out, how can you recognise that this 16-bit pattern (Byte 7 and Byte 8) is a positive number? **[1]**

(e)(i) Without any working out, how can you recognise that this 16-bit pattern (Byte 7 and Byte 8) is normalised? **[1]**

(ii) Both of the representations shown below are not normalised. Write in the empty rows the binary for the normalised form for the same value.

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

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

[3]

(f) A change is made to use the two bytes as a 12-bit mantissa with a 4-bit exponent. Describe the effect of this change on the values that can be represented, compared with the old use of the two bytes. **[2]**

3.1.3 Real Numbers and Normalised Floating Point Representation

May/ June 14.P33

5 (d) Many computer systems need to store and process real numbers.

A computer uses two bytes to store a real number. The first (Byte 7) stores the mantissa and the second (Byte 8) the exponent. Both mantissa and exponent use two's complement.

(i) What denary number is represented by Byte 7 and Byte 8?

Byte 7								Byte 8							
0	1	1	0	1	0	0	0	0	0	0	0	0	0	1	1

Show your working. **[3]**

(ii) How can you recognise that this 16-bit pattern (Byte 7 and Byte 8) is normalised? **[1]**

(iii) The positive number 2.0 is to be represented as a normalised real number.
Show the mantissa and exponent for this value.

Mantissa								Exponent							

(iv) What is the largest positive number that can be represented? Use the same 8-bit mantissa and 8-bit exponent.
Show the mantissa and exponent.

Mantissa								Exponent							

Do not attempt to evaluate this. **[2]**

(e) An alternative representation is suggested using a 6-bit mantissa with a 10-bit exponent.

Describe the effect on the numbers which can be represented, compared to the 8-bit mantissa and 8-bit exponent used earlier. **[2]**

3.1.3 Real Numbers and Normalised Floating Point Representation

Computer Science (9608)

May/ June 15.P31/ P32

5 A gardener grows vegetables in a greenhouse. For the vegetables to grow well, the temperature needs to always be within a particular range.

The gardener is not sure about the actual temperatures in the greenhouse during the growing season. The gardener installs some equipment. This records the temperature every hour during the growing season.

(c) The equipment records temperatures in the greenhouse. It does this for seven locations. Each recording is stored as two successive bytes.
The format is shown below:

Greenhouse location (Byte 1) / Temperature reading (Byte 2)

	7	6	5	4	3	2	1	0
Byte 1								

Byte 2								

[2]

The location is indicated by the setting of one of the seven bits in byte 1. For example, location 4 is indicated by setting bit 4.
Bit 0 of byte 1 acts as a flag:

- the initial value is zero
- when the reading has been processed it is set to 1

Byte 2 contains the temperature reading (two's complement integer).

(i) Interpret the data in byte 1 shown below:

	7	6	5	4	3	2	1	0
Byte 1	0	0	1	0	0	0	0	1

Byte 2	0	0	0	1	1	0	0	0

(ii) The system receives a temperature reading of –5 degrees from sensor 6.
Complete the boxes below to show the two bytes for this recording. The reading has not yet been processed.

	7	6	5	4	3	2	1	0
Byte 1								

Byte 2								

[2]

May/ June 15.P33

3.1.3 Real Numbers and Normalised Floating Point Representation

6 A company grows vegetables in a number of large greenhouses. For the vegetables to grow well, the temperature, light level and soil moisture need to always be within certain ranges.

The company installs a computerised system to keep these three growing conditions within the best ranges. Sensors are used for collecting data about the temperature, light level, and moisture content of the soil.

Each greenhouse has eight sensors (numbered 1–8).

- The byte at address 150 is used to store eight 1-bit flags.
- A flag is set to indicate whether its associated sensor reading is waiting to be processed.
- More than one sensor reading may be waiting to be processed at any particular moment.
- Data received from the sensors is stored in a block of eight consecutive bytes (addresses 201–208).
- The data from sensor 1 is at address 201, the data from sensor 2 is at address 202, and so on.

Sensor number

	1	2	3	4	5	6	7	8
150	0	1	0	0	0	1	0	1

201	0	0	0	0	0	0	0	0
202	0	0	0	0	0	1	0	0
203	0	0	0	0	0	0	0	0
204	0	0	0	1	0	0	0	0
205	0	0	0	0	0	0	1	0
206	0	0	0	1	0	1	0	0
207	0	0	0	1	0	0	1	0
208	0	0	0	1	0	0	1	0

(d) (i) Interpret the current reading for sensor 2. **[2]**

Oct/Nov 2015.P31/P33

1 In a particular computer system, real numbers are stored using floating-point representation with:
- 8 bits for the mantissa, followed by
- 4 bits for the exponent

Two's complement form is used for both mantissa and exponent.

(a) (i) A real number is stored as the following 12-bit binary pattern:

0	1	1	0	1	0	0	0		0	0	1	1

Calculate the denary value of this number. Show your working. **[3]**

3.1.3 Real Numbers and Normalised Floating Point Representation

(ii) Give the normalised binary pattern for +3.5. Show your working. **[3]**

(iii) Give the normalised binary pattern for –3.5. Show your working. **[3]**

The number of bits available to represent a real number is increased to 16.
(b) (i) If the system were to use the extra 4 bits for the mantissa, state what the effect would be on the numbers that can be represented. **[1]**
(ii) If the system were to use the extra 4 bits for the exponent instead, state what the effect would be on the numbers that can be represented. **[1]**
(c) A student enters the following expression into an interpreter:
OUTPUT (0.1 + 0.2)
The student is surprised to see the following output:
0.30000000000000001
Explain why this output has occurred. **[3]**

Oct/Nov 2015.P32
1 In a particular computer system, real numbers are stored using floating-point representation with:

- 8 bits for the mantissa, followed by
- 8 bits for the exponent

Two's complement form is used for both mantissa and exponent.
(a) (i) A real number is stored as the following two bytes:

Mantissa

0	0	1	0	1	0	0	0

Exponent

0	0	0	0	0	0	1	1

Calculate the denary value of this number. Show your working. **[3]**

(ii) Explain why the floating-point number in **part (a)(i)** is not normalised. **[2]**
(iii) Normalise the floating-point number in **part (a)(i)**.

Mantissa

Exponent

[2]

(b) (i) Write the largest positive number that can be written as a normalised floating-point number in this format.

Mantissa

Exponent

[2]

(ii) Write the smallest positive number that can be written as a normalised floating-point number in this format.

Mantissa

Exponent

[2]

(iii) If a positive number is added to the number in **part (b)(i)** explain what will happen. **[2]**

3.1.3 Real Numbers and Normalised Floating Point Representation

(c) A student writes a program to output numbers using the following code:

```
X ← 0.0
FOR i ← 0 TO 1000
   X ← X + 0.1
   OUTPUT X
ENDFOR
```

The student is surprised to see that the program outputs the following sequence:
0.0 0.1 0.2 0.2999999 0.3999999 ……
Explain why this output has occurred. **[3]**

May/June 2018.P31/P33

1 In a computer system, real numbers are stored using normalised floating-point representation with:

- 12 bits for the mantissa
- 4 bits for the exponent
- Two's complement form for both mantissa and exponent.

(a) Find the denary value for the following binary floating-point number.

Mantissa

1	0	1	1	1	0	0	1	1	0	1	0

Exponent

0	1	0	1

[3]

(b) Calculate the normalised floating-point representation of 5.25 in this system. Show your working.

Mantissa

Exponent

[3]

(c) The size of the mantissa is decreased and the size of the exponent is increased.
State how this affects the range and precision of the numbers that the computer system can represent. **[2]**