

### 3.5. Data Representation – Test 1

---

1. a) Describe the representation of unsigned denary integers in binary format. [1]

.....  
.....

b) Fill in the table below with the binary and hexadecimal representations of the given denary numbers: [4]

Denary	Binary	Hexadecimal
0		
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		

2. Calculate the result of each following binary expressions. Show your working and give the result in binary.

a)  $0111 + 0010$  [1]

b)  $1011 * 111$  [3]

3. a) Use the even-parity bit given to state whether an error has occurred during the transmission of the following numbers:
- i. 0100101011 parity 1 [1]  
.....
  - ii. 010111011 parity 0 [1]  
.....
  - iii. 11111111 parity 1 [1]  
.....
- b) Explain how a checksum is used for error detection. [3]  
.....  
.....  
.....  
.....
4. a) Explain why it is necessary to sample sounds when recording them to a computer. [2]  
.....  
.....  
.....
- b) Explain the difference between .wav, .midi and .mp3 files for storing music. [3]  
.....  
.....  
.....  
.....  
.....
- c) Explain the difference between lossy and lossless compression for sound files, commenting on the resulting file sizes and quality of the resulting file. Which form will most likely be used for streaming music over the Internet? [3]  
.....  
.....  
.....  
.....  
.....

5. a) Convert the following unsigned binary numbers into denary.

i. 01100110 [1]

ii. 10111001 [1]

iii. 11000101 [1]

b) Convert the following denary numbers into 8-bit unsigned binary.

i. 78 [1]

ii. 123 [1]

iii. 228 [1]

c) Convert the following 8-bit two's complement binary numbers into denary.

i. 00101101 [1]

ii. 10100111 [1]

iii. 11111111 [1]

d) Convert the following binary fractions into denary decimal numbers.

i. 0100.1100

[1]

ii. 0011.1110

[1]

ii. 1011.1001

[1]

6. Run-length encoding is a form of lossless compression. A simple way of representing run-length encoded data is as a series of pairs of bytes, with the first byte in each pair representing a character and the second byte in each pair representing the number of times the character is repeated.

Example: "HHHEEEELLLLLOOOOOOOO" would be encoded as ['H', 3, 'E', 4, 'L', 4, 'O', 8].

a) Encode the string "GGOOOOODDBYYYEEEE" using the format described above.

[2]

b) Run-length encoding on its own does not compress text well. Give an example of a type of compression that is better suited to compressing text and describe how it works.

[3]

.....

.....

.....

.....

.....

.....

**Total marks = /40**