# 1 Data Representation

## *1.1 Conversion & Negative Number Representation*

1. Perform the following binary subtractions by adding the 2's complement of the subtrahend. You MUST indicate whether or not overflow occurs. (FT **2003** Exam)
   01101011 - 00010110
   10011001 – 01001111

```
Answer:
01010101, valid
01001010, invalid
```

2. Complete the output when the following application is executed. (FT **2003** Test)

```
public class TestQ1 {
  public static void main(String[] args) {
    int n1 = 0x11;  //Hexadecimal
    int n2 = 011;   //Octal
    int n3 = Integer.parseInt("-11", 16);
    int n4 = Integer.parseInt("11", 2);
    int n5 = Integer.parseInt("11");
    System.out.println("n1 = " + n1); // 2 marks
    System.out.println("n2 = " + n2); // 2 marks
    System.out.println("n3 = " + n3); // 2 marks
    System.out.println("n4 = " + n4); // 2 marks
    System.out.println("n5 = " + n5); // 2 marks
  }
}
Answer =
17
9
-17
3
11
```

3. Convert the following base 8 number to base 3. Show your detailed working. (Conversion from base 8 to base 10 to base 3 is one useful approach.)[6 marks] (SC **2003** Test)
   a) $123_8$

```
Answer = 10002₃
```

4. Perform the following subtractions of signed binary numbers (where negative numbers are represented by their 2s-complement) by adding the 2's complement of the subtrahend. You MUST indicate whether or not overflow occurs. (SC **2002** Test)
   a) 10010101 – 01010111
   b) 10110111 - 11011101

```
Answer = 00111110, overflow
11011010
```

5. Perform the following subtraction of signed binary numbers (where negative numbers are represented by their 2s-complement) by adding the 2's complement of the subtrahend. You MUST indicate whether or not overflow occurs.
   a) 01010111 – 10010101 (SC **2003** Test)

```
Answer = 11000010, overflow
```

6. Convert the following unsigned 8-bit binary numbers into decimal. You must show all your working steps.
   i) 01010100
   ii) 11010110 (FT **2002** Test)



```
Answer:
a) 84
b) 214
```

7. What is the output of the following program? (FT **2002** Test)

```
public class TestQ7C {
  public static void main(String[] args) {
    int n1 = 0x12;
    int n2 = 012;
    int n3 = Integer.parseInt("-11", 16);
    int n4 = Integer.parseInt("1010", 2);
    int n5 = Integer.parseInt("1010");
    System.out.println("n1 = " + n1);
    System.out.println("n2 = " + n2);
    System.out.println("n3 = " + n3);
    System.out.println("n4 = " + n4);
    System.out.println("n5 = " + n5);
    }
}
Answer:
n1 = 18
n2 = 10
n3 = -17
n4 = 10
n5 = 1010
```

8. Perform the following binary subtractions by adding the 2's complement of the 8-bit subtrahend. You must show all your working steps and indicate the answer is valid or invalid (overflow).
   a) 00010010 – 01110110 (FT **2002** Test)

```
Answer = 10011100, valid
```

9. Perform the addition of the following 8-bit binary numbers. Given that the numbers are unsigned, state whether overflow has occurred, giving your reasons.
   01100110 + 11010010 (SC **2002** Exam)

```
Answer:
  01100110
  11010010
 100111000
Overflow, because there is a carry out.
```

10. (a) Add the following pair of unsigned binary numbers. Convert the answer into Decimal. Show your working. (FT **2001** Exam)
    i) 00100001 + 10001001
    (b) Add the following pair of signed binary numbers. Convert the answer into Decimal. Show your working.
    i) 00100001 + 10001001

```
Answer:
a) 170
b) -86
```

11. A) Convert the following unsigned 8-bit binary numbers into decimal. You must show all your working steps. [2 marks]
i) 10101010
ii) 00110101
B) Convert the following signed 8-bit 2's complement binary numbers into decimal. You must show all your working steps. [3 marks]
i) 10101010
ii) 00110101
C) Perform the following binary subtractions by adding the 2's complement of the 8-bit subtrahend. You must show all your working steps and indicate the answer is valid or invalid (overflow). [10 marks]

i) 10101010– 00110101
ii) 00110101– 10101010 (FT **2001** Test)

```
Answer:
a) 170
b) 53
a) -86
b) 53
Answer = 01110101 (2 marks) invalid/overflow (1 mark)
Answer = 10001011 (2 marks) invalid/overflow (1 mark)
```

12. Perform the following binary subtractions by adding the 2's complement of the subtrahend
a) 0111 1101 – 0010 0100
b) 0010 1010 – 0100 0101 (FT **2000** Test)

Answer:

```
2's complement 1101 1100
Answer = 0101 1001, valid
2's complement 1011 1011
Answer = 1110 0101 valid
```

13. Perform the following binary subtractions of signed 2-s complement binary numbers by adding the 2's complement of the subtrahend. You MUST indicate whether or not overflow occurs.
i) 00010101 - 10010111
ii) 10101010 – 01110110 (FT **2000** Exam)

```
Answer:
0111 1110 Valid
00110100 Invalid
```

14. Perform the following base-14 addition:
0019CA3 + 00AC136 (FT **2000** Exam)

```
Answer:
00C7DD9
```

15. Convert the following unsigned binary numbers to decimal. Show working.
i) 00100001
ii) 00101101 (FT **2000** Exam)

```
Answer:
33
45
```

16. Convert the following base 5 number to base 3. Show your detailed working (SC **2000** Test). (Conversion from base 5 to base 10 to base 3 is one useful approach.) 4321

Answer

$4321 = 4x5^3 + 3x5^2 + 2x5^1 + 1x5^0$
$= 4 \times 125 + 3 \times 25 + 2 \times 5 + 1$
$= 500 + 75 + 10 + 1$
$= 586$

```
586/3 = 195 remainder 1
195/3 = 65 remainder 0
65/3 = 21 remainder 2
21/3 = 7 remainder 0
7/3 = 2 remainder 1
2/3 = 0 remainder 2
```

586 = 210201

17. Do the following simple conversions of numbers between bases. Show your working. (SC **2000** Test).
i) $011111_{12}$ to base 10

```
Answer
result = 1000000 -1 = 2 6 -1 = 63
ii) 52810 to base 2
Answer
528 = 512 + 16 = 2 9 + 2 4
binary value = 1000000000+10000 = 1000010000
iii) 110111102 to base 16
```



```
Answer
1101, 1110 - DE16
```

18. Perform the following subtraction of 2's complement numbers by adding the 2's complement of the subtrahend. You MUST indicate whether or not overflow occurs. (SC 2000 Test)
i) 10010101 – 00010111

```
Answer =101111110 negative minus positive giving positive, therefore overflow
```

19. Convert the following unsigned binary numbers to decimal. Show your working.
i) 00100001
ii) 00101101 (SC **2000** Sample Test)

```
Answer:
33
45
```

20. Convert the following decimal number to base 6 representation
a) 555 [2 marks]
(ii) Perform the following binary subtractions by adding the 2's complement of the subtrahend. You must indicate whether or not overflow occurs. [1 mark]
10101010 – 01001010 (SC **2000** Exam)

21. Show the binary and hexadecimal representations of the following primitive int values: 100, 200, -1. Perform their addition and show their sum in binary, hexadecimal, and integer representations. Use all 32 bits for the binary representation, and all 8 hex digits for the hexadecimal representation. (FT **1999** Exam)

```
Answer:
100, 1100100, 64
200, 11001000, C8
-1, 11111111111111111111111111111111, FFFFFFFF
Sum = 100101011, 12B
```

22. Convert the following hexadecimal numbers to octal [4 marks] (SC **1999** Test)
3C6, F201

```
Answer:
a) 1706
b) 171001
```

23. Convert the following decimal numbers to binary, showing your working. Hint: you may convert first to octal or hexadecimal and thence to binary. [4 marks]
423, 234 [4 marks] (SC **1999** Test)

```
Answer:
a) 110100111
b) 11101010
```

24. Perform the following binary subtractions by adding the 2's complement of the subtrahend (SC **1999** Test)
a) 01110011 - 00001110
b) 00101100 - 01101001

```
Answer
a) 01010101 no overflow
b) 11000011 no overflow
```

25. (i) Convert the following unsigned binary numbers to decimal. (You may convert first to octal or hexadecimal and then to decimal.)
a) 110101
b) 01001101 [4 marks]
(ii) Convert the following unsigned decimal numbers to binary. (You may convert first to octal or hexadecimal and then to binary.)
a) 367
b) 902
(iii) Add the following pairs of unsigned binary numbers (some may carry into another bit)

a) 0101110 + 0001101
b) 101111 + 110010 [4 marks]

(iv) Perform the following base-16 addition
00296BA + 00FE967 (SC **1999** Exam)

```
Answer:
ia) 53
ib) 77
iia) 101101111
iib) 1110000110
iiia) 0111011
iiib) 1100001
iv) 128021
```

26. Convert the following unsigned binary numbers to decimal. (You may convert first to octal or hexadecimal and then to decimal.)
    a) 110101
    b) 01001101 [4 marks]

(ii) Add the following pairs of unsigned binary numbers (some may carry into another bit)
a) 0101110 + 0001101
b) 101111 + 110010 [4 marks]

(iii) Perform the following binary subtractions by adding the 2's complement of the subtrahend
a) 00110101 - 00010111
b) 00010101 - 00110100

(iv) Perfrom the following base-14 addition
0029CBA + 00ABD67 (SC **1998** Exam)

```
Answer:
ia) 53
ib) 77
iia) 0111011
iib) 1100001
iiia) 2's complement -> 11101001
Answer = 00011110 Valid
iiib) 2's complement -> 11001100
Answer = 11100001 Valid
iv) D7C43
```

27. Convert the following octal numbers to hexadecimal [4 marks]
    a) $7240_8$
    b) $5412_8$ (SC **1998** Test)

```
Answer
a) 1110 1010 0000 = EA0
b) 1011 0000 1010 = B0A
```

28. Convert the following decimal numbers to binary, showing your working. Hint: you may convert first to octal or hexadecimal and thence to binary. [4 marks]
    a) 387
    b) 235 (SC **1998** Test)

```
Answer
a) 603 (base 8) = 110000011
b) 353 (base 8) = 11101011
```

29. Perform the following binary subtractions by adding the 2's complement of the subtrahend. Indicate whether or not overflow occurs. [4 marks]
    a) 01101011 - 00010110
    b) 00101100 – 01101001 (SC **1998** Test)

```
Answer:
a) 01010101
b) 11000011
```

30. Convert the following octal numbers to hexadecimal



4260, 2345 (SC **1997** Test)

```
Answer:
4260  = 100 010 110 000 = 1000 1011 0000 = 8B0.
2345  = 010 011 100 101 = 0100 1110 0101 = 2E5.
```

31. Convert the following decimal numbers to binary, showing your working. Hint – you may convert first to octal or hexadecimal and thence to binary.
    483, 141 [4 marks] (SC **1997** Test)

```
Answer:
483 = 743 = 11110011
141 = 215 = 010001101
```

32. Perform the following binary subtractions by adding the 2's complement of the subtrahend
    a) 01101011 – 00010110
    b) 00101100 – 01101001 [6 marks] (SC **1997** Test)

```
Answer:
a) 01010101
b) 11000011
```

33. Convert the following unsigned binary numbers to decimal. (You may convert first to octal or hexadecimal and then to decimal.)
a) 101101
b) 01001001 [4 marks]

(ii) Add the following pairs of unsigned binary numbers (some may carry into another bit)
a) 0101110 + 0001101
b) 101111 + 110010 [4 marks]

(iii) Perform the following binary subtractions by adding the 2's complement of the subtrahend
a) 00110101 - 00010111
b) 00010101 - 00110100 (SC **1997** Exam)

```
Answer:
a) 45
b) 73
a) 0111011
b) 1100001
a) 2's complement = 11101001, answer = 00011110, valid.
b) 2's complement = 11001100, answer = 11100001, valid.
```

34. Convert the following octal numbers to hexadecimal
    426, 621, 1234, 1000 (SC **1996** Test)

```
Answer:
426  = 100 010 110  Regroup bits to 0001 0001 0110 = 116
621  = 110 010 001  Regroup bits to 0001 1001 0001 = 191
1234  = 001 010 011 100  Regroup bits to 0010 1001 1100 = 29C
1000  = 100 000 000 000  Regroup bits to 1000 0000 0000 = 800
```

35. Convert the following decimal numbers to binary, showing your working. Hint – you may convert first to octal or hexadecimal and thence to binary.
    527, 203 (SC **1996** Test)

```
Answer:
527  = 1017  = 001000001111
203  = 313  = 011001011
```

36. What would be the result of trying to represent the following values in 2's complement form, to 8 bit precision?
    +26, –17, 144, –0 [6 marks] (SC **1996** Test)

```
Answer:
+26 = 16 + 8 + 2 = 10000 + 1000 + 10 = 11010 = 00011010 to 8 bits
-17 = -(16 + 1) = -(01000 + 1) = -00010001 to 8 bits. To complement invert all the
bits and add 1 giving 11101110 + 1 = 11101111.
```

144 is larger than 127, the largest signed value which can be represented to 8 bits, so impossible
–0 has the same representation as +0 = 00000000, OR there is no representation for –0, so the conversion is impossible

37. Perform the following binary subtractions by adding the 2's complement of the subtrahend
    a) 01101011 –00010110
    b) 10010011 –01101011
    c) 00101100 –01101001
    d) 01100110 –01101010 (SC **1996** Test)

Answer:
a) 01010101
b) 00101000
c) 11000011
d) 11111100

38. Convert the following unsigned binary numbers to decimal. (You may convert first to octal or hexadecimal and then to decimal.)
a) 110101
b) 00101001 [4 marks]
(ii) Add the following pairs of unsigned binary numbers (some may carry into another bit)
a) 011010 +001101
b) 101111 +111010 [4 marks]
(iii) Perform the following binary subtractions by adding the 2's complement of the subtrahend
a)00110101 –00011011
b) 00011110 –00110100 (SC **1996** Exam)

Answer:
a) 53
b) 41
a) 100111
b) 1101001
a) 2's complement = 11100101, answer = 00011010, valid
b) 2's complement = 11001100, answer = 11101010, valid.

39. In hexadecimal do the two following 32 bit two's complement machine arithmetic calculations: (**1995**)
    a) 0x0098ba11 – 0x0098ba12
    b) 0x0098ba11 – 0x005265ec

40. Perform the following logical operations on the hexadecimal data given: (**1995**)
    a) 0xff00ffff AND 0x34a3295f = 0x3400295f
    b) 0x00ff0000 OR 0x34a3295f
    c) 0x00ff0000 XOR 0x34a3295f

## *1.2 Unicode*

41. Given the following paint() method defined for a Frame: (FT **2003** Test)

```
public void paint(Graphics g) {
        g.setFont(new Font("Arial Unicode MS", Font.BOLD, 30));
        String s = "\u0040\u03B8";
        g.drawString(s, 50, 50);
}
```

Draw the Frame as it appears on the screen. Position (50, 50) inside the Frame is shown by the dot.

Answer=

@θ

α

42. Convert the UCS-2 codes in the left hand column to UTF-8 codes. Write the UTF-8 as hexadecimal bytes. (FT **2003** Test)



0040 03B8

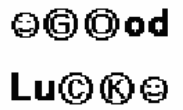
Answer = 40 CE B8

43. The following bytes are encountered in a UTF8 stream. What is the corresponding UCS2 code? (Express your answer in hexadecimal.) (SC **2003** Test)
    78 C8 BF
    Answer = 00 78 02 3F

44. What is the output of the following program? (Refer to the table given in the appendix) FT **2002** Exam

```
import java.applet.*;
import java.awt.*;
public class ExamUnicode extends Applet {
 public void paint(Graphics g) {
  Font font = new Font("Arial Unicode MS", Font.BOLD, 18);
  g.setFont(font);
  String line = "\u263a\u24bc\u24c4\u006f\u0064";
  char c1 = 0x004c, c2=0x0075, c3=0x24b8, c4=0x24c0, c5=0x263a;
  g.drawString(line, 50, 60);
  g.drawString("" + c1 + c2 + c3 + c4 + c5, 50, 90);
 }
}
```

Answer:



45. Give the representation of the string "cat" in ASCII, in Unicode, and in UTF8. Express you answers in hexadecimal. (3 marks) (SC **2002** Test)

Answer = From the table "cat" is 63 61 74 in ASCII This is obviously 00 63 00 61 00 74 in Unicode and 63 61 74 in UTF8.

46. The following bytes are encountered in a UTF8 stream. What is the corresponding UCS2 code? (Express your answer in hexadecimal.) (SC **2002** Test)
EF 80 8F

Answer = F00F

47. Convert the following unicode character to UTF8 form. Give your answer in hexadecimal. (SC **2002** Exam)
    0x0308

Answer:
0x0308 = 0000001100001000 = 00000 01100 001000
In UTF8: 110 01100 10 001111 = 11001100 10001000
= 0xCC88

48. What is the output of the following program? Please circle your answer. You can check the Unicode table from the Appendix at the end. (FT **2001** Exam)

```
public void paint(Graphics g) {
        g.setFont(new Font("Arial Unicode MS", Font.BOLD, 30));
        g.drawString("\u03b1\u0063\u263a\u24b8", 100, 100);
}
```

49. What is the output of the following program? Please circle your answer. You can check the Unicode table from the Appendix at the end. (FT **2001** Test)

```
public void paint(Graphics g) {
      g.setFont(new Font("Arial Unicode MS", Font.BOLD, 30));
      g.drawString("\u03b1\u0063\u263a\u24b8", 100, 100);
}
```

α c ☺ ©

Answer:

50. The following is a series of bytes (each in hexadecimal) representing some text in Unicode UTF-8 coding. (FT **2001** Exam)
CE B7 21 E2 99 A9 E2 92 B6
1. Convert the UTF-8 coding to UCS-2 coding.
2. Give the text corresponding to the UCS-2 coding. (Refer to the table given at Appendix)

Answer:
03B7 0021 2669 24B6

51. The following is a series of bytes (each in hexadecimal) representing some text in Unicode UTF-8 coding. (FT **2000** Test)
21 E8 A2 93 C7 8E
a) Convert the UTF-8 to UCS-2 code, writing the UCS-2 as individual hexadecimal bytes.

Answer:
21 E8 A2 93 C7 8E
21 =>0021
E8 A2 93 =>8893
C7 8E =>01CE

52. The following is a series of bytes (each in hexadecimal) representing some text in Unicode UCS-2 coding.
27 80 22 15 00 2C 00 20 00 2F 00 20 00 61 00 6E 00 64 20 44
a) Convert the UCS-2 to UTF-8 code, writing the UTF-8 as individual hexadecimal bytes. [10 marks]
b) As far as possible, give the text corresponding to the given UCS-2. [3 marks] (SC **1999** Test)

Answer:
2780->E2 9E 80 E2 E2 88 95 2C 20 2F 20 61 64 E2 81 84

53. Translate the following UCS-2 code into UTF-8
2262 0042 0391 (SC **1999** Exam)

Answer
E2 89 A2 42 CE 91

54. Translate the following UCS-2 code into UTF-8
0074 0079 0070 0065 0020 0214 0020 3092 002C 0020
As far as possible, what ASCII characters are represented? (SC **1998** Exam)

Answer:
74 79 70 65 20 C8 94 20 E3 82 92 2C 20

55. The following is an example of text in Unicode UTF-8 coding.
20 E6 98 AF 20 43 61 6C 69 73 20 C7 9A 44 E5 BB B6
a) Convert the UTF-8 to UCS-2 code, representing each UCS-2 character as a group of 4 hexadecimal digits, with characters separated by spaces.[10 marks]
b) As far as possible, give the text corresponding to the UTF-8 string. [2 marks] (SC **1998** Test)

Answer:
a) 0020 662F 0020 0043 0061 006C 0069 0073 0020 01DA 0044 5EF6
sp ?? sp C a l I s sp ?? D ??

## *1.3 Bit Operations*

56. What is the output of the following program? (SC **2003** Test)

```
public class BitOperators {
  public static void main(String[] args) {
    int x = 3; int y = 7;
    System.out.println( "x & y = " + (x & y)); //and
    System.out.println( "x | y = " + (x | y)); //or
    System.out.println( "y >> 2 = " + (y >> 2));//shift right
  }
}
```



Answer
x & y = 3
x | y = 7
x >> 2 = 1

57. What is the output of the following program? (FT **2002** Test)

```
public class TestQ8{
    public static void main(String[] args) {
        int x = 0x12;
        int y = 5;
        System.out.println( x );
        System.out.println( y );
        System.out.println( x & y );
        System.out.println( x | y );
        System.out.println( x ^ y );
        System.out.println( x >> 2 );
        System.out.println( y << 2 );
    }
}
```

Answer:
18
5
0
23
23
4
20

58. An int variable x in Java is intialised to the hexadecimal value 0xCAFEBABE. Write down the result of each of the following expressions (express your answers in hexadecimal or boolean as appropriate): (SC **2002** Test)
(i) x & 0xFFFF0000
(ii) x | 0x0000FFFF
(iii) x >> 16
(iv) x ^ 0xFFFF0000
(v) x ^ x ^ x
(vi) x < 16

Answer
0xCAFE0000
0xCAFEFFFF
0xFFFFCAFE
0x3501BABE
0xCAFEBABE
true

59. When the following piece of java code is executed: (SC **2002** Exam)
(i) What is the value held in variable b (in hexadecimal)?
(ii) What is the output from the program?
(ii) What is the purpose of the cast (i.e. (byte))?

```
short s = 0x00ff;
byte b = (byte)s;
System.out.println("s = " + s + ", b = " + b);
```

Answer:
i. b is: 0xff
ii. Output : s = 255 , b = -1
iii. The reason for the cast is:
To tell the compiler that you really mean to reduce the short to a byte, that it is not a mistake.

a) Complete the following Java method that switches the left and right halves of an int variable. (For example, if x is 0xCAFEBABE, the result will be 0xBABECAFE.) (SC 2002 Exam)

```
public static int switch (int x) {
  short left =
 short right =
 int result =
 return result;
}
```

```
Answer:
public static int switch (int x) {
short left = (short) (x >> 16);
short right = (short) x;
int  result = (right << 16) | (left & 0x0000FFFF);
return result;
}
```

60. What is the output of the following program? (105FT **2001** Test)

```
public class BitOperatos {
        public static void main(String[] args) {
                int x = 10;
                int y = 15;
                System.out.println( x & y );
                System.out.println( x | y );
                System.out.println( x ^ y );
                System.out.println( x >> 2 );
                System.out.println( y << 2 );
        }
}
```

```
Answer:
10
15
5
2
60
```

61. What is the output of the following program? (FT **2000** Exam)

```
public class BitOperators {
  public static void main(String[] args) {
    int x = 61; // Hint: What is the value in binary?
    int y = 31; // Hint: What is the value in binary?
    System.out.println( "x = " + x + " y = " + y);
    System.out.println( "x & y = " + (x & y)); //and
    System.out.println( "x | y = " + (x | y)); //or
    System.out.println( "x << 2 = " + (x << 2));//l. shft
    System.out.println( "x >> 2 = " + (x >> 2));//r. shft
  }
}
```

```
Answer:
 x=61 y=31
x & y = 29
x | y = 63
x <<2 = 244
x >> 2 = 15
```

62. What is the output of the following program? (SC 2000 Test)

```
public class BitOperators {
  public static void main(String[] args) {
        int x = -1; // Hint: What is the bit pattern in binary?
        int y = 15; // Hint: What is the bit pattern in binary?
```



```
        System.out.println( "x & y = " + (x & y)); //and
        System.out.println( "x | y = " + (x | y)); //or
        System.out.println( "x ^ y = " + (x ^ y)); //exclusive or
        System.out.println( "x << 2 = " + (x << 2));//left shift
        System.out.println( "x >> 2 = " + (x >> 2));//right shift
        //sign fill
  }
}
```

```
Answer
x & y = 15 (00000000000000000000000000001111)
x | y = -1 (11111111111111111111111111111111)
x ^ y = -16 (11111111111111111111111111110000)
x << 2 = -4 (11111111111111111111111111111100)
x >> 2 = -1 (11111111111111111111111111111111)
```

63. What is the output of the following program? (SC **2000** Exam)

```
public class BitOperators {
  public static void main(String[] args) {
    int x = 3;
    int y = 12;
    System.out.println( "x & y = " + (x & y)); //and
    System.out.println( "x | y = " + (x | y)); //or
    System.out.println( "x ^y = " + (x ^ y)); //or
    System.out.println( "x << 2 = " + (x << 2));//l. shft
    System.out.println( "x >> 2 = " + (x >> 2));//r. shft
  }
}
```