1. Using C++ data types for a **machine that uses a char of 9-bits and a short of 18 bits**, convert the following into two’s complement big-endian binary and if not, then show why not:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Value</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>unsigned char</td>
<td>x = 'A';</td>
<td>00010101</td>
</tr>
<tr>
<td>unsigned char</td>
<td>x = 0x255;</td>
<td>11111101</td>
</tr>
<tr>
<td>signed char</td>
<td>x = 255;</td>
<td>01111111</td>
</tr>
<tr>
<td>unsigned char</td>
<td>x = 128;</td>
<td>10000000</td>
</tr>
<tr>
<td>unsigned char</td>
<td>x = 35;</td>
<td>00100001</td>
</tr>
<tr>
<td>signed char</td>
<td>x = 127;</td>
<td>01111111</td>
</tr>
<tr>
<td>signed char</td>
<td>x = -128;</td>
<td>11111111</td>
</tr>
<tr>
<td>signed char</td>
<td>x = -07;</td>
<td>11111111</td>
</tr>
<tr>
<td>signed short</td>
<td>x = 35;</td>
<td>00110101</td>
</tr>
<tr>
<td>signed short</td>
<td>x = 'a';</td>
<td>01110101</td>
</tr>
</tbody>
</table>
2. Using C++/C#/Java operator precedence, add the correct parenthesis (signed int a, b, ..., w, x, y, z):

\[
\begin{align*}
    a &= x \mid y ^ {\wedge} w \& z; \\
    a &= z + y * z \% w / v - c;
\end{align*}
\]

3. Using VHDL operator precedence, add the correct parenthesis:

\[
a <= b + c \text{ SRL d AND e XOR f OR NOT g MOD h} * i - j;
\]

4. Using C++ convert the following into two's complement big-endian binary that machine that uses a char of 10-bits: where unsigned char u, a=0x85, b=0x96, c=02; signed char s, w=0x80, x=0x96, y=0, z=0x15; For addition and subtraction indicate if overflow and/or carry has occurred.

\[
\begin{align*}
    u &= \neg a; \\
    u &= a \& b; \\
    u &= a ^ b; \\
    u &= a ^ 'A'; \\
    u &= a - b; \\
    u &= a << 2; \\
    s &= -w; \\
    s &= w \& x; \\
    s &= w + x; \\
    s &= x << 2; \\
\end{align*}
\]

5. Convert the 24-bit number 0x414243 to mime base64: ____________

6. Convert the base64 "T2s=" to ASCII: ____________
7. What is the parity of 0x414243 (even or odd)?

8. Write a "single" C code statement of setting both bits 5 and 2 to 1 in the variable int a.

9. Write the C code function to count the number 1 bits in an integer: unsigned int bcount(unsigned int a); (note: multiply and divide not allowed). Example: bcount(0x1a) is 3.

10. What is the hamming distance of 0xAF and 0377 (show work)?

11. Give the n-cube, k-map, SOP of the f(a,b,c) minterms for (0,1,4,5,6), then give the minimize SOP, then draw the logic gate schematic.

   Solution see wakerly Figure 4-29 and read text.
12. Minimize the $f(a,b,c,d)$ minterms for $(1,3,4,5,9,11,12,13,14,15)$. Show k-map, and label "prime implicants".

   Solution see wakerly Figure 4-32 and read text.

13. Minimize the $f(a,b,c,d)$ minterms for $(1,2,3,5,7)$ and a Don’t Care minterm of $(10,11,12,13,14,15)$. Give k-map and Minimized SOP.

   Solution see wakerly Figure 4-37 and read text.