1) Number representation

a) For each binary vector below, what does it represent as an Octal number and a Hexadecimal number?

<table>
<thead>
<tr>
<th>Binary</th>
<th>Octal</th>
<th>Hexadecimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>10110011</td>
<td>263</td>
<td>B3</td>
</tr>
<tr>
<td>01111100</td>
<td>174</td>
<td>7C</td>
</tr>
<tr>
<td>11111111</td>
<td>377</td>
<td>FF</td>
</tr>
</tbody>
</table>

b) Given $X$ and $Y$ below that are 5-bit numbers in 2’s complement form, fill in the table of what $S$ should be bit-vector form (even if the result causes an overflow). Indicate whether or not there is an overflow and what $S$ is as a number (if $S$ results in an overflow, then just put an “x” under the $S$ column). (First two rows provide examples.)

<table>
<thead>
<tr>
<th>$X$</th>
<th>$Y$</th>
<th>$S = X - Y$</th>
<th>Overflow</th>
<th>$S$ as a number</th>
</tr>
</thead>
<tbody>
<tr>
<td>11101</td>
<td>00011</td>
<td>11010</td>
<td>no</td>
<td>-6</td>
</tr>
<tr>
<td>00001</td>
<td>00111</td>
<td>11010</td>
<td>no</td>
<td>-6</td>
</tr>
<tr>
<td>01000</td>
<td>10010</td>
<td>10110</td>
<td>yes</td>
<td>x</td>
</tr>
<tr>
<td>10111</td>
<td>10111</td>
<td>00000</td>
<td>no</td>
<td>0</td>
</tr>
<tr>
<td>10111</td>
<td>01011</td>
<td>01100</td>
<td>yes</td>
<td>x</td>
</tr>
<tr>
<td>01100</td>
<td>00101</td>
<td>00111</td>
<td>no</td>
<td>7</td>
</tr>
<tr>
<td>10101</td>
<td>10111</td>
<td>11110</td>
<td>no</td>
<td>-2</td>
</tr>
</tbody>
</table>
2) Multi-level logic minimization

Given following logic equations, minimize the number of literals, e.g. by using common subexpressions, Boolean rules, etc. You can introduce new immediate equations for common sub-expressions if it helps to reduce the number of literals. Put a box around your final answer, and indicate the number of literals in your final answer.

\[
\begin{align*}
S &= \overline{A}BC \\
P &= \overline{(A + B + C + D)(A + B + C + E) + (A + B + C + AD)} \\
&= \overline{A}BCD + \overline{A}BCE + \overline{A}BC(AD) \\
&= \overline{A}BCD + \overline{A}BCE + \overline{A}BC(A + D) \\
&= \overline{A}BCD + \overline{A}BCE + \overline{A}BC(AD) \\
&= \overline{A}BCD + \overline{A}BCE + \overline{A}BCD \\
&= \overline{A}BC(D + E) \\
&= S(D + E) \\
Q &= \overline{A}BCF + \overline{A}BCG + \overline{A}BCH + XYF + XYG + XYH \\
&= \overline{A}BC(F + G + H) + XY(F + G + H) \\
&= (F + G + H)(\overline{A}BC + XY) \\
&= (F + G + H)(S + XY) \\
R &= F + G + H + \overline{F}G \\
&= 1
\end{align*}
\]

Final answer (12 literals):

\[
\begin{align*}
S &= \overline{A}BC \\
P &= S(D + E) \\
Q &= (F + G + H)(S + XY) \\
R &= 1
\end{align*}
\]
3) Two-level logic minimization

\[ F(A, B, C, D) = \sum m(2, 3, 5, 7, 8, 10, 15) \]
\[ d(A, B, C, D) = \sum m(1, 12, 13) \]

\[
\begin{array}{c|cccc|}
\hline
   & 00 & 01 & 11 & 10 \\
\hline
00 & x & 1 & 1 &   \\
01 & 1 & 1 &   &   \\
11 & x & x & 1 &   \\
10 & 1 &   &   & 1 \\
\hline
\end{array}
\]

a) Identify all the prime and essential prime implicants.

<table>
<thead>
<tr>
<th>Prime</th>
<th>Essential</th>
</tr>
</thead>
<tbody>
<tr>
<td>0–1</td>
<td>( \bar{A}D )</td>
</tr>
<tr>
<td>001-</td>
<td>( \bar{A}\bar{B}C )</td>
</tr>
<tr>
<td>-1-1</td>
<td>( BD )</td>
</tr>
<tr>
<td>110-</td>
<td>( ABC )</td>
</tr>
<tr>
<td>1-00</td>
<td>( A\bar{C}\bar{D} )</td>
</tr>
<tr>
<td>10-0</td>
<td>( \bar{A}\bar{B}\bar{D} )</td>
</tr>
<tr>
<td>-010</td>
<td>( \bar{B}\bar{C}\bar{D} )</td>
</tr>
</tbody>
</table>

b) Find the minimum two-level logic implementation.

\[ F = \bar{A}\bar{B}C + BD + \bar{A}\bar{B}\bar{D} \]
4) Multiplexor/Adder/Subtractor

Implement a 3-bit adder/subtractor using only 2:1 multiplexors and inverters. When $\text{OP} = 0$, do an addition. Else (when $\text{OP} = 1$), do a subtraction.

The equations are:

- $\text{Overflow} = C_3 \oplus C_2$
- $C_0 = \text{OP}$
- $Z_i = \text{OP} \oplus Y_i$
- $C_i+1 = X_i Z_i + C_i B_i$

$\text{Overflow} = C_3 \oplus C_2$
$Z_i = \text{OP} \oplus Y_i$
$B_i = X_i \oplus Z_i$
$S_i = B_i \oplus C_i$

$C_0 = \text{OP}$
$Z_i = \text{OP} \oplus Y_i$
$B_i = X_i \oplus Z_i$
$S_i = B_i \oplus C_i$