1)

a) From truth table of XOR (right)

we see that $x \oplus 1 = x'$. Hence

when $op = 1$, $B_i' = B_i \oplus 1$

is the input to each full adder.

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
<th>$x \oplus y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

b) When $A = 0$, the output of this circuit is the

two's complement negative of $B$. For example,

let $A = 0$, $B = 0101$ (decimal 5), $op = 1$. The

circuit would compute

\[
\begin{array}{c|c|c|c}
1 & (op carried in) & 1010 & (complement of 5) \\
1010 & & & \\
+ 0000 & (0) & & \\
\hline
1011 & (-5) & & \\
\end{array}
\]

This is exactly the complement-and-add-one

method of negation we defined in class.

c) When $A \neq 0$ and $op = 1$, we are simply adding

$A$ to the (two's complement) negative of $B$.

Hence, we are subtracting $B$ from $A$. 
Note we could also use gate instantiation for the wires instead of "assign" statements, as in

\[
\text{XOR}(Q0, B[0], \text{Op})
\]

Here is the output:
3) The key here is realizing the 3 address lines are shared between the first two MUX's:

![Diagram of MUX's](image)

4a) Here is the truth table (A3-A0 are inputs, C1-C0 are outputs):

<table>
<thead>
<tr>
<th>A3</th>
<th>A2</th>
<th>A1</th>
<th>A0</th>
<th>C1</th>
<th>C0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>X</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

b) This is exactly a priority encoder with the outputs negated (complemented). You would just draw a priority encoder with two inverters on its outputs.

c) The definition of the CLZ function specifies a nonzero number; this is because we essentially have 4 leading zeros if the inputs are all zero, and we cannot represent the decimal number 4 in two bits.

5)