1. From lecture: \( g_i \) in CMOS:

\[
\begin{align*}
&g_i &\rightarrow &g_{[i,i+3]} \\
\end{align*}
\]

Bubble push:

\[
\begin{align*}
&g_i &\rightarrow &g_{[i,i+3]} \\
\end{align*}
\]

Simplify:

(Showing whole circuit)

\[
\begin{align*}
g_i &\rightarrow \quad &g_{[i,i+3]} \\
\end{align*}
\]

Critical path:

\[
\begin{align*}
\text{NAND4 - NAND4} \\
\end{align*}
\]

\[
\begin{align*}
S_1 &= 1 & S_2 &\quad \text{Cout} = 12 C_{\text{ref}} \\
\end{align*}
\]

Should be \( b \). I'll explain in the next section.
1.1. **Number of stages**: \( N = 2 \)

- **Path logical effort** \( G \):
  \[
  G = g_1 \cdot g_2 = \frac{4+r}{1+r} \cdot \frac{4+r}{1+r} = \frac{6}{3} \cdot \frac{6}{3} = 4
  \]

- What I meant to say for the \( \text{Cin} \) question is that the gate is "unscaled":

[Diagram of a gate]

Sees \( \text{Cin} = 6 \)

Which would give \( \text{Cin} = 6 \) just as in the logical effort calculation.

Confusingly, I instead gave you that \( S = 1 \), and since \( \text{Cin} = S \cdot \text{Cref} \), many of you answered \( \text{Cin} = \text{Cref} \). This is not what I intended, but was marked correct.

- **Path logical effort** \( H \):
  \[
  H = \frac{C_{\text{last}}}{C_{\text{first}}} = \frac{12}{6} = 2
  \]
  (or \( H = 12 \) if you had \( \text{Cin} = \text{Cref} \))

- **Total Path effort**:
  \[
  F = GH \cdot B = 4 \cdot 2 = 8
  \]
  (or \( F = 48 \) if you had \( \text{Cin} = \text{Cref} \))

- **Optimum Stage effort**:
  \[
  \hat{f} = F^{1/N} = 8^{1/2} = \sqrt{8} = 2.83
  \]
  (or \( \hat{f} = 6.93 \) if you had \( \text{Cin} = \text{Cref} \))

- \( P = n_1 \cdot \text{Pref} + n_2 \cdot \text{Pref} = 4 + 4 = 8 \)

  \( \text{number of inputs to gate} \)

- \( D = N \cdot \hat{f} + P = 13.7 \)
  (or 17.9 if you had \( \text{Cin} = \text{Cref} \))
optimized gate sizes:

\[ f = g_i \cdot h_i = g_i \cdot \frac{C_{out}}{C_{in}} = g_i \cdot \frac{C_{out}}{S_i \cdot C_{ref}} \Rightarrow S_i = \frac{g_i \cdot C_{out}}{f \cdot C_{ref}} \]

\[ S_2 = \frac{g_2 \cdot 1.12 \cdot C_{ref}}{f \cdot C_{ref}} = \frac{2 \cdot 12}{2.83} = 8.48 \text{ (or } 3.46 = \frac{2}{6.93} \text{ if you took } C_{in} = C_{ref}) \]

\[ S_1 = \frac{g_1 \cdot 8.48 \cdot C_{ref}}{2.83 \cdot C_{ref}} = \frac{2 \cdot 8.48}{2.83} = 5.99 \leftarrow = 6 \text{ as in } C_{in} = S \cdot C_{ref} \text{ if } C_{ref} = 1 \]

So, \( S = 1 \) and \( C_{ref} = 1 \) were pretty bad assumptions to have you make, because it causes \( g = \frac{C_{in}}{C_{ref}} = 1 \) which is not true. Also, the lecture notes showed an AND4-OR4 path that reduced to a NAND4-NAND4 path, which further complicated this problem. Grading will be lenient.

For the sake of education, let's examine this problem with the more reasonable assumptions: \( C_{ref} = 3 \), \( C_{in} = 6 \) and \( S = 2 \):

- \( G = 4 \), \( C_{in} = 6 \), \( H = \frac{C_{last}}{C_{first}} = \frac{12}{6} = 2 \), \( F = 6 \cdot H = 24 \)
- \( f = F^{\frac{1}{2}} = 4.9 \), \( P = 8 \), \( D = 2 \cdot 4.9 + 8 = 17.8 \)

\[ S_2 = \frac{g_2 \cdot 1.12 \cdot C_{ref}}{f \cdot C_{ref}} = \frac{2 \cdot 12}{4.9} = 4.9 \]

\[ S_1 = \frac{g_1 \cdot 4.9}{4.9} = 2 \leftarrow \text{as required} \]
LA: NAND4 gates: 

\[ pMOS \quad W_i = rW_{ref} S_i = 2S_i \]
\[ nMOS \quad W_i = 4W_{ref} S_i = 4S_i \]
\[ \text{total LA} = 4(2S_i + 4S_i) = 24S_i \]

So, first, I gave you a wrong interpretation here of how gates are actually sized.
(The "unscaled" NAND4 gate with \( W_n = 4W_{ref} \) and \( W_p = rW_{ref} \) actually has \( S_i = 2 \)).

The correct equation should therefore be scaled by \( g_i \):

\[ pMOS \quad W_i = \frac{rW_{ref} S_i}{g_i} = \frac{2S_i}{2} = S_i \]
\[ nMOS \quad W_i = \frac{4W_{ref} S_i}{g_i} = \frac{4S_i}{2} = 2S_i \]
\[ \text{total LA} = 4(3S_i) = 12S_i \]

So here are some possible answers:

Using information as problem is written:
2 NAND4 gates with \( S_1 = 6, S_2 = 8.48 \):
\( \text{LA} = 24(6 + 8.48) \) \[ \text{LA} = \boxed{347.5} \]

2 NAND4 gates with \( S_1 = 1, S_2 = 3.46 \):
\( \text{LA} = 24(1 + 3.46) \) \[ \text{LA} = \boxed{104.3} \]

Using the corrected data and corrected LA approximation:
\( S_1 = 2, S_2 = 4.9 \) \( \Rightarrow \) \[ \text{LA} = 12(2 + 4.9) = \boxed{82.8} \]
Problem 1.2

The AND-OR path can be converted to all NAND and INV as in the first circuit (problem 1.1 above)

Full circuit:

The critical path runs through 6 NAND2 gates.
- \( N=6 \), \( g_1, g_2, g_3, g_4, g_5, g_6 = (g_{\text{NAND2}})^6 = \left(\frac{4}{3}\right)^6 = 5.62 \)
- \( C_{\text{in}} = \begin{cases} 
C_{\text{ref}} & \text{if you used } C_{\text{ref}} = S_i C_i \\
4 & \text{if you used } C_{\text{in}}=4 \text{ as in } g_{\text{NAND2}} \text{ calculation}
\end{cases} \)
- \( H = \frac{12 \cdot C_{\text{ref}}}{C_{\text{in}}} = \begin{cases} 
12 & \text{if you used } C_{\text{ref}}=C_{\text{in}} \text{ [CASE 1]} \\
3 & \text{if you used } C_{\text{in}}=4, C_{\text{ref}}=1 \text{ [CASE 2]} \\
9 & \text{if you take the more realistic } C_{\text{in}}=4, C_{\text{ref}}=3 \text{ [CASE 3]}
\end{cases} \)
- \( F = GHB = \begin{cases} 
67.44 & \text{[CASE 1]} \\
16.86 & \text{[CASE 2]} \\
50.58 & \text{[CASE 3]}
\end{cases} \)
\[ f = \begin{cases} 2.018 \quad \text{[CASE 1]} \\ 1.601 \quad \text{[CASE 2]} \\ 1.92 \quad \text{[CASE 3]} \end{cases} \]

\[ D = N \cdot f + P = \begin{cases} 24.11 \quad \text{[CASE 1]} \\ 21.61 \quad \text{[CASE 2]} \\ 23.54 \quad \text{[CASE 3]} \end{cases} \]

**Sizing:**

- **CASE 1:** (Used \( C_{ref} = C_{in} \), \( S_i = 1 \))
  
  \[ S_i = \frac{g_{ibi}}{\hat{f}} \quad \frac{C_{i+1}}{C_{ref}} = \frac{(4/3)(1)}{2.018} \quad \frac{C_{i+1}}{C_{ref}} = 0.661 \quad C_{i+1} \]

  \( S_6 = (0.661) 12 = 7.93 \)
  
  \( S_5 = (0.661) 7.93 = 5.24 \)
  
  \( S_4 = (0.661) 5.24 = 3.47 \)
  
  \( S_3 = (0.661) 3.47 = 2.29 \)
  
  \( S_2 = (0.661) 2.29 = 1.51 \)
  
  \( S_1 = (0.661) 1.51 = 0.998 \approx 1 = C_{in}/C_{ref} \)

\[ \sum_{i=1}^{6} S_i = 21.44 = S_{\text{TOTAL}} \]

- **CASE 2:** (\( C_{ref} = 1 \), \( C_{in} = 4 \))
  
  \[ S_i = \frac{(4/3)(1)}{1.601} \quad \frac{C_{i+1}}{C_{ref}} = 0.833 \quad C_{i+1} \]

  \( S_6 = (0.833) 12 = 9.99 \)
  
  \( S_5 = (0.833) 9.99 = 8.32 \)
  
  \( S_4 = (0.833) 8.32 = 6.93 \)
  
  \( S_3 = (0.833) 6.93 = 5.77 \)
  
  \( S_2 = (0.833) 5.77 = 4.81 \)
  
  \( S_1 = (0.833) 4.81 = 4.00 = C_{in}/C_{ref} \)

\[ \sum_{i=1}^{6} S_i = 39.8 = S_{\text{TOTAL}} \]
Case 3: (Cref = 3, Cin = 4) \[ S_i = \frac{g_i S_{i+1} \cdot Cref}{\hat{C} \cdot Cref} = 0.6945 S_{i+1} \]

\[
\sum_{i=1}^{6} S_i = S_{\text{TOTAL}} = 24.22
\]

Logical Area:

\[ \text{pMOS: } W_i = 2W_{\text{ref}} S_i = 2S_i \]
\[ \text{nMOS: } W_i = 2W_{\text{ref}} S_i = 2S_i \]

\[ 2 \cdot (2S_i + 2S_i) = 8S_i \]

For 6 NAND2 gates, this gives \( L_A = 8 \sum_{i=1}^{6} S_i = 8S_{\text{TOTAL}} \)

\[
\text{Case 1: } L_A = 8 \cdot (21.44) = 171.52
\]
\[
\text{Case 2: } L_A = 8 \cdot (39.8) = 318.4
\]

Case 3: If you account for the fact that an unscaled gate has \( S = 4/3 \), you can see how the LA requires the logical effort \( g \) in the calculation for the actual sizes of gates:

\[ \text{pMOS: } W_i = \frac{g W_{\text{ref}} S_i}{4/3} = \frac{2S_i}{4/3} = \frac{8}{3} S_i \Rightarrow L_A = 2(\frac{16}{3} S_i) \]
\[ \text{nMOS: } W_i = \frac{g W_{\text{ref}} S_i}{4/3} = \frac{2S_i}{4/3} = \frac{8}{3} S_i \]

\[ L_A = \frac{32}{3} S_{\text{TOTAL}} = 258.3 \]
Problem 1:
So to compare these two structures:

For CASE 1 \((C_{in} = C_{ref}, C_{ref} = 1)\): \[
\begin{array}{c}
2 \cdot \text{NAND4} \\
D = 17.9 \\
LA = 104.3
\end{array} \quad \begin{array}{c}
6 \cdot \text{NAND2} \\
D = 24.11 \\
LA = 171.52
\end{array}
\]

For CASE 2 \((C_{in} = 6 \text{ or } 4, C_{ref} = 1)\): \[
\begin{array}{c}
D = 13.7 \\
LA = 347.5
\end{array} \quad \begin{array}{c}
D = 21.61 \\
LA = 318.4
\end{array}
\]

For CASE 3 \((C_{in} = 6 \text{ or } 4, C_{ref} = 3)\): (corrected \(LA\) approx.) \[
\begin{array}{c}
D = 17.8 \\
LA = 82.8
\end{array} \quad \begin{array}{c}
D = 23.54 \\
LA = 258.3
\end{array}
\]

So, for the case of NAND4-NAND4 vs. 6 stages of NAND2's, the NAND4 implementation is faster and takes up less room. This would not necessarily be the case if the gates were not optimized, or if NOR gates were used instead of converting the paths to all NAND structures.
Problem 2

Active Low decoder: 0 on selected line, 1 on other lines:

<table>
<thead>
<tr>
<th>S2</th>
<th>S1</th>
<th>S0</th>
<th>d0</th>
<th>d1</th>
<th>d2</th>
<th>d3</th>
<th>d4</th>
<th>d5</th>
<th>d6</th>
<th>d7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

NAND gate: \[
\begin{array}{c|c|c}
  a & b & f \\
  0 & 0 & 1 \\
  0 & 1 & 1 \\
  1 & 0 & 1 \\
  1 & 1 & 0 \\
\end{array}
\]

with inverted input:

\[
\begin{array}{c|c|c}
  a & b & f \\
  0 & 0 & 0 \\
  0 & 1 & 1 \\
  1 & 0 & 1 \\
  1 & 1 & 0 \\
\end{array}
\]

which is just an OR gate, gives correct logic.

\[
\begin{array}{c}
  S_2 & d_0 \\
  S_1 & \ \ \ \ \ \ \ \\
  S_0 & \ \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  \overline{S}_2 & \overline{d}_2 \\
  \overline{S}_1 & \ \ \ \ \ \ \\
  \overline{S}_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  S_2 & d_3 \\
  S_1 & \ \ \ \ \ \ \\
  S_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  \overline{S}_2 & \overline{d}_3 \\
  \overline{S}_1 & \ \ \ \ \ \ \\
  \overline{S}_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  S_2 & d_6 \\
  S_1 & \ \ \ \ \ \ \\
  S_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  \overline{S}_2 & \overline{d}_6 \\
  \overline{S}_1 & \ \ \ \ \ \ \\
  \overline{S}_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  S_2 & d_4 \\
  S_1 & \ \ \ \ \ \ \\
  S_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  \overline{S}_2 & \overline{d}_4 \\
  \overline{S}_1 & \ \ \ \ \ \ \\
  \overline{S}_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  S_2 & d_7 \\
  S_1 & \ \ \ \ \ \ \\
  S_0 & \ \ \ \ \ \\
\end{array}
\]

\[
\begin{array}{c}
  \overline{S}_2 & \overline{d}_7 \\
  \overline{S}_1 & \ \ \ \ \ \ \\
  \overline{S}_0 & \ \ \ \ \ \\
\end{array}
\]
Problem 3

Negative-edge triggered TFF with clear:

[Diagram of a negative-edge triggered TFF with clear]
Problem 4