Sample Problems for Quiz # 1
(with solutions)

Here are sample problems to help you prepare for Quiz 1 on Sept. 26.

1. Logic Functions from CMOS Transistors

Recall that we introduced the abstractions for transistors shown in Figure 1. A high voltage corresponds to logical one and a low voltage to logical zero.

Determine the logical functions implemented by the CMOS circuits in Figure 2, Figure 3, and Figure 4. Write down the truth tables. (If you’re taking EE2301, and so know something about Boolean algebra, also write down the algebraic expressions.)

Solution (a)

\[ Y = (A' + B')(C' + D') = \overline{AB} + \overline{CD} \]
Figure 2: CMOS Circuit (a).
Figure 3: CMOS Circuit (b).
Figure 4: CMOS Circuit (c).
Solution (b)

\[ Y = A + B(C + D) = A(B + CD) \]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</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>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</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>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Solution (c)
2. Logic Functions from Gates

(a) Determine the logical functions implemented by the logic gates in Figure 5. Write down the truth tables. (If you’re taking EE2301, and so know something about Boolean algebra, also write down the algebraic expressions.)

Solution (a)
\[ Y = \overline{A(B + CD)} \]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</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>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</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>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Solution (b)

\[ Y = \overline{A}B + A\overline{B} = A \oplus B \]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>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>

Solution (c)

\[ Y = \overline{A}B + \overline{C}D \]
(b) Write the truth tables for the Boolean functions implemented by the logic gates in Figures 6 and 7.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</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>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</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>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

![Figure 6: Network of Logic Gates.](image-url)
Solution:

\begin{array}{cccc|cc}
 x & y & z & c & s \\
 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 1 \\
 0 & 1 & 0 & 0 & 1 \\
 0 & 1 & 1 & 1 & 0 \\
 1 & 0 & 0 & 0 & 1 \\
 1 & 0 & 1 & 1 & 0 \\
 1 & 1 & 0 & 1 & 0 \\
 1 & 1 & 1 & 1 & 1 \\
\end{array}

Figure 7: Network of Logic Gates.

Solution:

\begin{array}{cccccccccc}
 x_1 & x_2 & x_3 & w_1 & w_2 & w_3 & w_4 & w_5 & w_6 & w_7 & w_8 & w_9 \\
 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\
 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\
 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\
\end{array}
3. Iterative and Recursive C Programs.

(a) What will the following C program print out?

```c
#include <stdio.h>

int main()
{
    int i, p, q, s, t;
    p = 1;
    q = 1;
    printf("%d, ", p);
    for (i = 1; i <= 10; ++i) {
        printf("%d", p);
        s = p;
        t = q;
        q = p;
        p = 2*s + 3*t;
        if (i < 10) printf("", ");
    }
    printf("\n");
}
```

Solution:
1, 1, 5, 13, 41, 121, 365, 1093, 3281, 9841, 29525

(b) What will the following C program print out?

```c
#include <stdio.h>
int main()
{
    int q, p, n, base, i, j, sum;
    q = 1;
    p = 2;
    n = 5;
    base = -1;
    sum = 0;
    for (i = 1; i <= n; ++i) {
        p = p * base ;
    }
    for (j = 1; j <= n; ++j) {
        q = q * j;
    }
```
for (j = 1; j <= i; ++j) {
    sum = sum + q * j;
}
printf ("%d\n", sum );

Solution:
2520

(c) What will the following C program print out?

#include <stdio.h>

int main ()
{
    int i, p, q, s, t;
    p = 1;
    q = 1;
    printf("%d, ", p);
    for (i = 1; i <= 10; ++i) {
        printf("%d", p);
        s = p;
        t = q;
        q = p;
        p = s + t;
        if (i < 10) printf("", ");
    }
    printf("\n");
}

Solution
1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89

(d) What the following program print out?

#include <stdio.h>

int mario ( int base, int n)
{
    int i, p;
    p = 1;
    for (i = 1; i <= n; ++i)
        p = p * base ;
return p;
}
int luigi (int n)
{
    int i, p;
p = 1;
for (i = 1; i <= n; ++i)
    p = p * i;
return p;
}
int main ()
{
    int i, j, sum ;
    for (i = 1; i <= 10; ++i) {
        sum = 0;
        for (j = 1; j <= i; ++j) {
            sum = sum + (mario (-1, i - j) * luigi (j ));
        }
        printf ("%d %d\n", i, sum );
    }
}
1 1
2 1
3 5
4 19
5 101
6 619
7 4421
8 35899
9 326981
10 3301819

(e) Consider the following program. It implements an algorithm that is 2309 years old (seriously). To whom is this algorithm due?

#include <stdio.h>

int gcd(int x, int y) {
    if (y == 0)
        return x;
    else
return gcd(y, x % y);
}

int main(int argc, char **argv)
{
    int x = atoi(argv[1]);
    int y = atoi(argv[2]);
    printf("%d\n", gcd(x,y));
}

Suppose that it is called with the following arguments. What will it print out?
./gcd 450 72
Answer: 18

(f) What will the following C program print out?

#include <stdio.h>

int ferrari(int n)
{
    int i, p;
    p = 1;
    for (i = 1; i <= n; ++i)
        p = p * i;
    return p;
}

int lamborghini(int n, int m) {
    int i, j, k;
    i = ferrari(n);
    j = ferrari(m);
    k = ferrari(n-m);
    return i / (j * k);
}

int main()
{
    int i, j, sum;
    for (i = 0; i <= 7; ++i) {
for (j = 0; j <= i; ++j) {
    printf("%d", lamborghini(i, j));
    if (j < i) printf("", ");
}
printf("\n");
}

Answer:
1
1, 1
1, 2, 1
1, 3, 3, 1
1, 4, 6, 4, 1
1, 5, 10, 10, 5, 1
1, 6, 15, 20, 15, 6, 1
1, 7, 21, 35, 35, 21, 7, 1