CS 201 – HW #2

Your Name: _____________________________________________________________

Please print this out, write your answers CLEARLY, and turn in hardcopy.

Perform these calculations without using a calculator!!!

QUESTION 1: Create a table showing the first 16 binary numbers, their decimal values, and their representations as a hex numeral.

<table>
<thead>
<tr>
<th>Binary</th>
<th>Decimal</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
QUESTION 2: Convert the following binary numbers to hex.

0110 0011 0101 1100 1001 0001 0111 1111: ________________
0000 0010 0100 0110 1000 1010 1100 1110: ________________
1111 1101 1011 1001 0111 0101 0011 0001: ________________

QUESTION 3: Convert the following hex numbers to binary.

A6C2 : ______________________________________
80 : ______________________________________
0000001 : ___________________________________
ffff : ______________________________________
87654321 : __________________________________
9ABCDEF0 : __________________________________

QUESTION 4: How many bits in a byte? ______

QUESTION 5: How many bytes in a

8-bit quantity? ______
16-bit quantity? ______
32-bit quantity? ______
64-bit quantity? ______

QUESTION 6: Show an arbitrary value in binary. (Just make up values; what counts is the number of bits.)

1 byte quantity: ______________________________________
2 byte quantity: ______________________________________
4 byte quantity: ______________________________________
QUESTION 7: Show an arbitrary value in hex. (Just make up values; what counts is the number of numerals.)

- 64-bit quantity: 
- 32-bit quantity: 
- 16-bit quantity: 
- 8-bit quantity: 

QUESTION 8: Create a table of powers of 2

- $2^0$: 
- $2^1$: 
- $2^2$: 
- $2^3$: 
- $2^4$: 
- $2^5$: 
- $2^6$: 
- $2^7$: 
- $2^8$: 
- $2^9$: 
- $2^{10}$: 
- $2^{16}$: 
- $2^{32}$: 


QUESTION 9: Convert the following binary numbers to decimal:

10 : _______
100 : _______
1000 : _______
10000 : _______
1100 : _______
10101010 : _______
100 : _______
01110 : _______

QUESTION 10: Convert the following decimal numbers to binary. Show each as a 2 byte quantity!

13: _______________________
32: _______________________
256: _______________________
486: _______________________
6,831: _______________________
89: _______________________
143: _______________________
65,535: _______________________
32,768: _______________________

QUESTION 11: How many bits in a “C” language variable of type char? ______
How many bytes? ______

QUESTION 12: How many bits in a “C” language variable of type short? ______
How many bytes? ______

QUESTION 14: How many bits in a “C” language variable of type “long long”? _____ How many bytes? _____
(HINT: “long long” is an abbreviation for “long long int”. And “long” is an abbreviation for “long int”. However, “long long int” and “long int” are not necessary the same size.)

QUESTION 15: How many bits in a “C” language variable of type float? _____ How many bytes? _____


QUESTION 17: Take the following bit strings and perform the bitwise logical AND operation.

0101 1100 1010 1111 0110 0110 0111 1011
1101 0110 0100 0011 0111 1001 1000 0011

Show the result in binary: _________________________________
Show the result in hex: _________________________________

QUESTION 18: Using the same values, perform the bitwise logical OR operation.

Show the result in binary: _________________________________
Show the result in hex: _________________________________

QUESTION 19: Using the same values, perform the bitwise logical XOR operation.

Show the result in binary: _________________________________
Show the result in hex: _________________________________
**QUESTION 20:** Add the following binary numbers.

0101 1010 0110 1011
0100 1100 0111 1010

Show the result in binary: ________________________________

**QUESTION 21:** Add the following 16-bit numbers, which are given in hex. (There are two approaches: convert the numbers to binary first or perform the addition entirely in hex. It’s a good way to check your answer.)

3D3A
5DD9

Show the result in hex: ________________

**QUESTION 22:** Assuming two’s complement representation (i.e., “signed” numbers), convert the following decimal values to 8-bit binary values and show in…

<table>
<thead>
<tr>
<th>Binary</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>0:</td>
<td></td>
</tr>
<tr>
<td>+1:</td>
<td></td>
</tr>
<tr>
<td>+2:</td>
<td></td>
</tr>
<tr>
<td>+126:</td>
<td></td>
</tr>
<tr>
<td>+127:</td>
<td></td>
</tr>
<tr>
<td>-1:</td>
<td></td>
</tr>
<tr>
<td>-2:</td>
<td></td>
</tr>
<tr>
<td>-127:</td>
<td></td>
</tr>
<tr>
<td>-128:</td>
<td></td>
</tr>
</tbody>
</table>
**QUESTION 23:** Assuming two’s complement representation (i.e., “signed” numbers), convert the following decimal values to 16-bit binary values and show in…

<table>
<thead>
<tr>
<th>Decimal Value</th>
<th>Binary</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+32766</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+32767</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-32767</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-32768</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 24:** Assuming two’s complement representation (i.e., “signed” numbers), convert the following 8-bit binary values to decimal:

- 00000000: ________
- 00000001: ________
- 00000010: ________
- 01111110: ________
- 01111111: ________
- 10000000: ________
- 10000001: ________
- 11111110: ________
- 11111111: ________
QUESTION 25: Assuming two’s complement representation (i.e., “signed” numbers), convert the following 16-bit values (shown in hex) into decimal:

- 0000:  
- 0005:  
- 0007:  
- 7ffe:  
- 7fff:  
- 8000:  
- 8001:  
- fffe:  
- ffff:  

QUESTION 26: Describe Big Endian.

........................................................................................................................................
........................................................................................................................................

QUESTION 27: Describe Little Endian.

........................................................................................................................................
........................................................................................................................................

QUESTION 28: How many bits are used for addresses (i.e., pointers) on the IA32 architecture? ________ On the x86-64 architecture? ________

QUESTION 29: What does the following x86-64 instruction do?

```
movq $1234, %rax
```

........................................................................................................................................
........................................................................................................................................
QUESTION 30: What does the following x86-64 instruction do?

\[
\text{movq 1234,} \%\text{rax}
\]

QUESTION 31: What do the following x86-64 instructions do to register \%rax? Be careful to specify what happens to all bits.

\[
\begin{align*}
\text{movq} & \quad \text{123,} \%\text{rax} \\
\text{movl} & \quad \text{123,} \%\text{eax} \\
\text{movw} & \quad \text{123,} \%\text{ax} \\
\text{movb} & \quad \text{123,} \%\text{al} \\
\text{movb} & \quad \text{123,} \%\text{ah}
\end{align*}
\]

QUESTION 32: What does the following x86-64 instruction do:

\[
\text{addq} \quad \%\text{rbx,} \%\text{rdx}
\]
QUESTION 33: Write an instruction to add the contents of 16-bit register %dx to %cx and place the result in %dx:

_____________________________________________

QUESTION 34: Write an instruction to add the contents of 64-bit register %rsi to %rdi and place the result in %rdi:

_____________________________________________

QUESTION 35: Assume j in an “int” and is stored in %edx. What instruction will perform this “C” assignment statement?

j = 123;

_____________________________________________

QUESTION 36: Assume k in a “long long” and is stored in %rbx. What instruction will perform this “C” assignment statement?

k = 123;

_____________________________________________

QUESTION 37: Consider these “C” declarations; how many bytes are used to store the variables?

int i; _______
long long k; _______
char myCh; _______
int myArry [8]; _______
long long yourArry [10]; _______
char * p; _______
int * p3; _______
char * myA [4]; _______
QUESTION 38: Here are signed numbers, shown in hex. The size of each quantity is obvious from the number of hex numerals. Negate each quantity and show the result in hex.

00000000: ____________________________
0001: ____________________________
034B: ____________________________
AB: ____________________________
7FFFFFFF: ____________________________
80000000: ____________________________
1234: ____________________________
1AC8 F203 20B4 4957: ____________________________
FF: ____________________________
00000001: ____________________________

QUESTION 39: Here are some binary fractions. What is the number in decimal. (Please write your answer in this form: 4³/₄, not 19/4 or 4.75)

0.1: ________
1.01: ________
11.001: ________
0.0001: ________
111.011: ________
110.111: ________
101.101: ________
101.1010: ________
101.1010000: ________
**QUESTION 40:** In the x86-64 architecture, there are 16 registers of 64 bits each. What are their names? (We use ATT notation, so don’t forget the %.)

__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________
__________________

Page 12
**QUESTION 41:** Show the ASCII codes for the following characters:

<table>
<thead>
<tr>
<th></th>
<th>Decimal</th>
<th>Binary</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>'a':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'A':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'j':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'J':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'0':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'3':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>')':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>' ':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'\0':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'\n':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
<tr>
<td>'\r':</td>
<td>_____</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>