CSCI 3320 Last Name: ________________
Fall 2016
Exam 3 First Name: ________________

Write all answers on the answer sheets provided. You *may* use the following resources during the exam: *x86 Assembly Quick Reference*, `iomacros.asm` source code, calculator.

1. (4 pts) Name two common applications of bit manipulations in programming.

2. (8 pts) Show how the following section of Java code could be implemented in assembly language. Be sure to show declarations of labels as part of your answer.

```
int x;
float y;
double z,a;
...
a= z*4.5 + x/5.0 + y-3;
```

3. Consider the following Java function (recall that a variable to type `long` is a 64-bit integer).

```
public static long calc(long x, long y)
{
   if (x < 0) return y;
   if (x==0) return 10;
   return calc(x-4,y) + calc(x/2,y);
}
```

   (a) (8 pts) Write this function in assembly language assuming a stack-based calling convention in which registers are saved by the callee. Be sure to make proper use of RBP to track the call frame.

   (b) (8 pts) Write this function in assembly language assuming a register-based calling convention in which registers are saved by the caller.

4. Suppose a text file named `donors.txt` contains a list donors along with the (dollar/integer) amount each donor has given:

```
Fred Smith
3000
Alice Jones
200
Amy Franks
2000
Joe Blow
10
...
```

(a) (10 pts) Write an assembly function called `readDonations` that accepts the address of an array as a parameter and will read the entries from the file into the provided array so that the name and donation amount are stored as a record with the null-terminated name having 32 bytes of space and the donation amount being stored as a 64-bit integer. Use either of the two standard calling conventions. The function should return the number of entries read from the file.

(b) (6 pts) Write the assembly language code necessary to call an externally linked C-language subroutine with the following prototype: `void displayDonations(Donor [] d, int n)` (where `n` is the number of donors stored as a 32-bit integer value).

(c) (10 pts) Write an assembly language function called `sumDonations` that given the address of the donor array along with the number of donations in the array will return the sum of the donations.

5. (12 pts) The Communists of America political party has issued a questionnaire containing 26 yes or no questions. An individual's answers to those question are encoded as a bit string so that a "no" answer is stored as a zero and a "yes" answer is stored as a 1. The questions are numbered 0 to 25 and correspond to the bits having the same number. The Communists of America, besides being against everything, also like to keep things simple. For that reason, all questions in the survey are worded so that a negative answer ("no") indicates an opinion that is in line with the party dogma. A positive answer ("yes") indicates an "incorrect" viewpoint. Suppose that the register RBX contains a bit string representing the answers of a person who has completed the survey. Your job is to write a section of code that will count and display the number of "yes" answers. In this way a party official can immediately view the score and determine whether or not the survey represents a friend or foe.

6. (2 pts each) Define each of the following terms:

   (a) ripple-carry adder

   (b) multiplexor

   (c) ALU

   item (6 pts) Draw a half-adder circuit using AND, OR, and NOT gates.

7. Suppose a "friend" asks you to design an efficient circuit that given three binary inputs, `a`, `b`, and `c`, will produce the result, `r`, portrayed in the following chart:

a	b	c	r
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

(a) (4 pts) Represent this chart as a boolean algebra expression.

(b) (2 pts) How many gates are represented by the expression you generated in problem 7a?

(c) (4 pts) Apply various rules of boolean algebras to simplify the expression as much as you can.

(d) (4 pts) Represent the simplified expression as a circuit diagram.

(e) (2 pts) How many gates are represented by your simplified diagram?

NOTE: In case you find it helpful, the five properties of a Boolean algebra (along with two additional properties) are listed below:

Commutative	$x + y = y + x$	$x \cdot y = y \cdot x$
Associative	$(x + y) + z = x + (y + z)$	$(x \cdot y) \cdot z = x \cdot (y \cdot z)$
Distributive	$x + (y \cdot z) = (x + y) \cdot (x + z)$	$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$
Identity	$x + 0 = x$	$x \cdot 1 = x$
Complement	$x + \overline{x} = 1$	$x \cdot \overline{x} = 0$
DeMorgan's	$\overline{x + y} = \overline{x}\overline{y}$	$\overline{xy} = \overline{x} + \overline{y}$
Involution	$\overline{\overline{x}} = x$	