CSCI 356 Fall 2017 : Practice Exam I

DO NOT OPEN EXAM PACKET UNTIL INSTRUCTED TO DO SO

YOU MAY FILL IN INFORMATION ON THE FRONT NOW

PLEASE TURN OFF ALL ELECTRONIC DEVICES

ID#:	
Name:	

- This exam is closed book. You are allowed one (1) 8.5" x 11" handwritten note sheet
- You will have eighty (80) minutes to complete this exam.
- Answer the questions only in the spaces provided on the question sheets.
- If you give multiple solutions to a problem without indicating which one you want graded, the grader may select one to grade.
- Your answers do not need to be complete, grammatically correct sentences.

Problem	Points	Possible
1		8
2		4
3		4
4		6
5		4
6		8
7		6
8		7
Total		47

1. Solve the following project one style problem.

```
/*
 * isAsciiDigit – return 1 if 0x30 <= x <= 0x39 (ASCII codes for characters '0' to '9')
 *   Example: isAsciiDigit(0x35) = 1.
 *            isAsciiDigit(0x3a) = 0.
 *            isAsciiDigit(0x05) = 0.
 *   Legal ops: ! ~ & ^ | + << >>
 *   Max ops: 15
 */

int isAsciiDigit(int x) {
```

2. Write the following base-10 integers in **eight-bit two's complement**. Express your answer in both binary and hex (base-16).

    a. 53

    b. -75

3. Interpret the following as hex representations of **two's complement integers (eight bits each)**. Write them both in binary and in base-10.

    a. 0xCF

    b. 0x49

4. Consider the **eight-bit floating point format**. In eight-bit floating point, there is one sign bit, three exponent bits, and four fractional bits. The exponent bias is 3.

    a. What number is 0110 1100 in base 10?

    b. How would 3.3125 (= 3 + 5/16) be represented in eight-bit floating point?

5. Give a value that makes each following expressions false, and explain why it makes the expression false. If there is no value for x and y that would make the expression false, indicate that. In each case, x and y are of type int.

    a. ((x^y) < 0)

    b. ((x >> 31) + 1) >= 0

6. I have a C function with the following signature:
   `int practice_exam_problem(int a, int b);`

Here is the assembly code for it:
```
<+0>:cmp    %esi,%edi
<+2>: jle      0x4005be <practice_exam_problem+12>
<+4>: lea      0x5(%rsi,%rsi,1),%eax
<+8>: cmp    %eax,%edi
<+10>: je      0x4005d4 <practice_exam_problem+34>
<+12>:cmp    %esi,%edi
<+14>:jge      0x4005ca <practice_exam_problem+24>
<+16>:lea      0x4(%rdi,%rdi,2),%eax
<+20>: cmp   %eax,%esi
<+22>: je      0x4005da <practice_exam_problem+40>
<+24>: cmp   %esi,%edi
<+26>: jne      0x4005e0 <practice_exam_problem+46>
<+28>: mov   $0x4,%eax
<+33>: retq
<+34>: mov   $0x3,%eax
<+39>: retq
<+40>: mov   $0xa,%eax
<+45>: retq
<+46>: mov   $0x2,%eax
<+51>:retq
```

   a. Give a value for parameters to make it return 2.

   b. Give a value for parameters to make it return 3.

   c. Give a value for parameters to make it return 4.

   d. Give a value for parameters to make it return 10.

7. Consider the following struct on an x86-64 Linux machine:

```
struct my_struct {
    char a;
    long b;
    short c;
    float *d[2];
    unsigned char e[3];
    float f;
};
```

    a. Please lay out the struct in memory below. Clearly indicate:

        i.   any bytes used for padding.
        ii.   The boundaries of each type
        iii.   the end of the struct.

    b. How many bytes will the struct occupy if our compiler optimizes for access time?

    c. How many bytes will the struct occupy if our compiler optimizes for space?

8. Draw the stack frames of test and getbuf, given that the Instruction Pointer is currently at **0x004017c7** and the stack pointer is at **0x5561dcac** at the start of test. Indicate where the stack pointer is and the addresses and the content of the stack frames (variable names are ok).

C code:

```
void test() {                                      unsigned getbuf() {
    int val;                                           char buf[BUFFER_SIZE];
    val = getbuf();                                    gets(buf);
    printf("No exploit. Getbuf returned 0x%x\n", val);  return 1;
}                                                  }
```

Assembly Code: Stack

test:		
0x00401984	sub	$0x8,%rsp
0x00401988	mov	$0x0,%eax
0x0040198d	callq	4017c3<getbuf>
0x00401992	mov	%eax,%edx
0x00401994	mov	$0x4031d8,%esi
0x00401999	mov	$0x1,%edi
0x0040199e	mov	$0x0,%eax
0x004019a3	callq	400e00<__printf_chk@plt>
0x004019a8	add	$0x8,%rsp
0x004019ac	retq	
getbuf:	sub	$0x28,%rsp
0x004017c3	mov	%rsp,%rdi
0x004017c7	callq	401a4d <Gets>
0x004017ca	mov	$0x1,%eax
0x004017cf	add	$0x28,%rsp
0x004017d4	retq	
0x004017d8		

Stack

Address	Contents
test stack frame	
getbuf stack frame	

Instruction Pointer	0x004017c7
Stack Pointer	