CS246 Sample Exam #1
February 21, 2017

This exam is open-book/open-note. You may use any printed resources you like. You may **not** use any computing devices, such as laptops, phones, or calculators.

This sample exam has **not** been length-tested. It may be shorter or longer than the real thing.

When writing code, the logic behind your code is more important than syntactic accuracy. In other words, pay attention to getting your loops right, not your semicolons.

1. Write the following in octal notation:

    a. 0x123   _____

    b. 0xABC   _____

    c. 0xCA1   _____

    d. 13   _____

2. Consider the following pairs of bytes. When considered as `signed chars`, which one is greater? Circle the one that is considered greater.

    a. `0b00110010`

        `0b01001101`

    b. `0b00000001`

        `0b10000000`

    c. `0b00000000`

        `0b11111111`

    d. `0b00111111`

        `0b01000000`

3. What is the result of the following operations? Write your answer in the same representation (that is, decimal, hexadecimal, octal, or binary) that the question is asked in. In the case of shift operations, use the representation of the number to the left of the shift operator. Consider all values to be `unsigned chars`.

    a. `0b00110001 << 2`   _____

    b. `0b10011000 &`
       `0b10110011`   _____

    c. `0b10011000 |`
       `0b10110011`   _____

    d. `0b10011000 ^`
       `0b10110011`   _____

    e. `0x0F >> 1`   _____

    f. `13 << 1`   _____

    g. `27 & 1`   _____

    h. `~0x0F`   _____

4. What is printed when `mystery(3)` is called?

```c
int mystery(int a)
{
    for(int i = 0; i < 10; i++)
    {
        if(a % 2 == 0)
        {
            a /= 2;
        }
        else
        {
            a = a * 3 + 1;
        }

        if(a == 1)
        {
            return i;
        }

        printf("%d %d\n", a, i);
    }

    return -1;
}
```

Put program output here:

You may use this space for scratchwork:

i	a

5. Write a function to compute a difference array. A *difference array* is an array that contains the difference between every two elements in an input array. For example, if a is 5-element array containing `{3, 4, 6, 1, -10}`, then `darray(5, a, b)` will fill the array b with `{1, 2, -5, -11}` because 4 - 3 = 1, 6 - 4 = 2, 1 - 6 = -5, and -10 - 1 = -11. You may assume that the length of the input array is greater than 0.
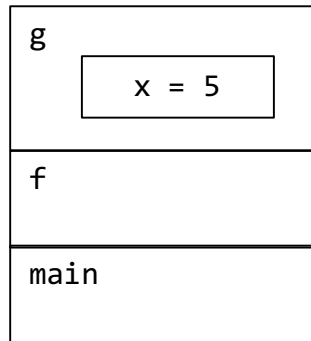
```
void darray(int n, int a[n], int b[n-1])
{



}
```

6. Consider the following functions:
   ```
   int main() { f(); }    void f() { g(5); }    void g(int x) { }
   ```
   At its deepest point, the stack frames of this program look like this:

   ```
   ┌─────────────────────────┐
   │ g                       │
   │      ┌────────────┐     │
   │      │  x = 5     │     │
   │      └────────────┘     │
   ├─────────────────────────┤
   │ f                       │
   │                         │
   ├─────────────────────────┤
   │ main                    │
   │                         │
   └─────────────────────────┘
   ```

   a. Now consider the following program. What do its stack frames look like at their deepest point? Recall that parameters are local variables, too. Note that the example above is just to set the format for the answer; it has no bearing on the code below. Draw your stack frames to the right of the code.

   ```
   int r(int a, int b)
   {
        if(b == 0)
        {
             return 1;
        }
        else
        {
             return a * r(a, b-1);
        }
   }

   int main()
   {
        printf("%d\n", r(3, 4));
   }
   ```

   b. What does the program print?

   _____