15-110 Midterm #1a – Fall 2018
50 minutes

Name: _____

Andrew ID: _____@andrew.cmu.edu

Section: _____

- **You may not use any books, notes, or electronic devices during this exam.**
- **You may not ask questions about the exam except for language clarifications.**
- **Show your work on the exam (not scratch paper) to receive credit.**
- **If you use scratch paper, you must submit it with your andrew id on it, and we will ignore it.**
- **All code samples run without crashing. Assume any imports are already included as required.**
- **Do not use these post-midterm1 topics: strings, lists, etc.**

DO NOT WRITE IN THIS AREA		
Part 1 (CT)	30 points	
Part 2 (Logic Circuits) (xnor)	10 points	
Part 3 (Very Short Answers)	20 points	
Part 4 (FR / CMU-A)	10 points	
Part 5 (FR / primeCount)	10 points	
Part 6 (FR / avgOfEvenDigits)	15 points	
Part 7/bonus	5 points bonus	
Total	95 points	
(Not on exam: Watch The Imitation Game)	5 points	___

1. **[30 pts; 10 pts each] Code Tracing**

Indicate what each will print. Place your answer in the boxes below each block of code. Show your work, outside the box, for partial credit.

```
def ct1(n):
    m = 2*n
    d = 10
    while (n < m):
        print(n, d)
        n += d
        d += 10
    return n
print(ct1(40)) # prints 7 values
               # on 4 lines
```

```
def ct2(lo, hi):
    result = 0
    for z in range(lo, hi):
        if (z%2 == 1):
            print(z, result)
            result += z%10
    return result
print(ct2(20,24)) # prints 5 values
                  # on 3 lines
```

```
def ct3(d, m):
    for x in range(m):
        for y in range(m):
            if (x * y == d):
                print(x, y)
print(ct3(5, 6)) # prints 5 values
                 # on 3 lines
```

2. [10 pts] Logic Circuits: xnor

The function xnor is the opposite of xor. So (x xnor y) is True if both x and y are True, or neither x nor y are True. Also, (x xnor y) is False if exactly one of x or y is False.

A. [4pts] Write the Truth Table for (x xnor y)
Hint: the table should have 4 rows in it.

B. [3 pts] Write (x xnor y) in Disjunctive Normal Form (DNF, using only And, Or, and Not)

C. [3 pts] Draw a logic circuit that computes (x xnor y) using only And, Or, and Not gates, that matches your DNF expression from the previous step (2B). Each gate should be drawn as a rectangle with the word And, Or, or Not inside.

3. [20 pts; 2.5 pts each] Very Short Answers

Answer each of the following very briefly.

A. In just a few words, state one topic that the Blums discussed in their guest lecture.

B. Multiply 23 * 37 using lattice multiplication. Show your work.

C. In just a few words, state one important technical contribution Alan Turing made.

D. When we add two 1-bit values x and y, we get a 2-bit result. The high-order bit is (x and y). What logical function of x and y describes the low-order bit (the one's digit)?

E. In the number-guessing game from our case study, the user picks a number between 0 and 100, inclusive, and the computer guesses 50. If that is too low, what will the next guess be? Show your work.

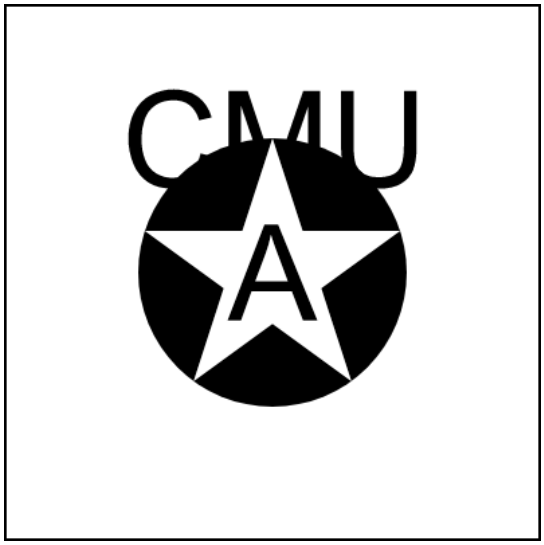
F. In just a few words, state a winning algorithm for the game of Nim.

G. In just a few words, why did we switch from ascii to unicode?

H. Multiply 9 * 6 using Egyptian Multiplication. Show your work.

4. **[10 pts] Free Response: CMU-A**

Using the CS Academy drawing functions, draw this image:



Notes:

* all numbers in your code should be either a single digit or a multiple of 100
* all colors are black or white
* the outer 400x400 rectangle is the canvas border, and not part of the drawing

5. **[10 pts] Free Response: primeCount**

Note: for this problem, assume you already have the function isPrime(n) that returns True if n is prime and False otherwise. Do not write isPrime(n) here!

Using the isPrime(n) function, write the function primeCount(n) that takes a possibly-negative integer n and returns the number of primes up to n, inclusive.

Note that the first several primes are: 2, 3, 5. So:
primeCount(1) returns 0
primeCount(2) returns 1
primeCount(3) returns 2
primeCount(4) returns 2
primeCount(5) returns 3

Also, since there are no primes smaller than 2, primeCount(-10) returns 0.

6. **[15 pts] Free Response: averageOfEvenDigits**

Without using strings, write the function averageOfEvenDigits(n) that takes an integer n and returns the average of the even digits in n. Here are some sample test cases for you:

```
assert(averageOfEvenDigits(12345) == 3)  # (2+4)/2 == 6/2 == 3
assert(averageOfEvenDigits(102201) == 1) # (0+2+2+0)/4 == 4/4 == 1
assert(averageOfEvenDigits(1) == None)   # no even digits --> None
assert(averageOfEvenDigits(0) == 0)      # handle 0!
assert(averageOfEvenDigits(-2) == 2)     # handle negatives!
```

Reminder: do not use strings.

7. Bonus/Optional: **[2.5 pts] What will this print? Clearly circle your answer.**

```
def bonusCt1(n):
    a = n%10
    b = n//10%10
    n //= 100
    m = 0
    while (n > 0):
        d = n%10
        n //= 10
        if ((d > a) and (b > d)):
            m = 10*m + d
    return m
bigNumber = 269995000794699429990000081
print(bonusCt1(bigNumber))
print(bonusCt1(bonusCt1(bigNumber)))
```

Bonus/Optional: **[2.5 pts] What will this print? Clearly circle your answer.**

```
def bonusCt2(n):
    result = 0
    sign = 1
    for i in range(n):
        for j in range(i+1):
            result += sign * j
            sign = -sign
    return result
for i in range(100):
    k = bonusCt2(i)
    if (abs(k)//2 == 10):
        print(i, k)
```