Math 9 Midterm – version A

`Directions:` The exam is 50 minutes long. Please read each question carefully.

EACH QUESTION IS WORTH 20 POINTS When asked to write code, you should write working Python code that has correct syntax.
Use the backs of the pages if needed.

Last Name: _____

First Name: _____

Student ID #: _____

Question	Points	Score
1	20	
2	20	
3	20	
4	20	
5	20	
Total:	100	

1. (20 points) Write down the output of the following programs.

   1.
```
i = 97
while i >= 0:
  print(i)
  i -= 10
print i
```

   2.
```
def f(n):
  count = 0
  while n >= 1:
    n = n // 2
    count += 1
  return count

print (f(15), f(16))
```

   3.
```
def g(n):
  if n == 0:
    return []
  return [n % 10] + g(n // 10)

print g(5120)
```

2. (20 points) Write code to produce the following lists:

    1. `[1,2,3,4,5,11,12,13,14,15,21,22,23,24,25,31,32,33,34,35]`

    2. `[9,99,999,9999,99999,999999,9999999]`

    3. `[1,3,5,7,1,3,5,7,1,3,5,7]`

3. (20 points) Write down the output of the following code:

   1. (10 pts)

```
reduce(lambda x, y: x*y, [2 for i in range(5)])
```

   2. (10 pts)

```
reduce(lambda x, y: x if (x>y) else y, range(5))
```

4. (20 points) Write down a Python function `second_largest(xs)` that will return the second largest element of a list `xs`. [Hint: sort but do not use the built-in `sort()` function]

5. (20 points) Write down a Python function `base_10(binary_number)` that will convert a number in binary to its base-10 equivalent, e.g. `base_10(1101)` should return `13`. [Hint: use `digits(binary_number)` which gives you a list of the digits of the binary number in reverse order and remember that, e.g., 11 in base 2 is $2^1 + 2^0$ in base 10.]

```
def digits(n):
    '''returns a list containing digits of n in reverse order'''
    if n == 0:
        return []
    return [n % 10] + digits(n // 10)

def base_10(binary_number):
    # your code here
```