1. [5 points] What does each of the following expression evaluate to? Assume that \texttt{isPrime} is a boolean function that takes one argument and returns \texttt{True} if that argument is a \textit{prime number}; otherwise the function returns \texttt{False}. Assume that \texttt{concat} is a function that takes two arguments \texttt{a} and \texttt{b} and returns \texttt{a + b}.

(a) \texttt{map(range, range(1, 10, 3))} Ans. \([0, [2, 3, 4], [6, 7, 8, 9, 10]]\)

(b) \texttt{len(filter(isPrime, range(20)))} Ans. 8

(c) \texttt{reduce(concat, map(str, range(1, 15, 3)))} Ans. '1471013'

(d) \texttt{reduce(concat, range(1, 10, 2))} Ans. 25

(e) \texttt{reduce(concat, reduce(concat, map(range, range(5))))} Ans. 10

Turn over for Problem 2.
2. **[5 points]** Here is a partially completed function called `secondMax` that takes a list of numbers as a parameter and returns the number that is second largest in the list. For example, if the given list is \([-1, 11, 3, 8, 1, 7]\) then the function would return 8. If the given list is \([-1, 11, 3, 11, 1, 7]\) then the function would return 11. Using the built-in Python functions and methods, we can solve this problem in 3 lines of code. The idea is to find the maximum element \(m\), then find the index (position) of \(m\), and then find the maximum element in the list obtained by excluding \(m\). Your task is to supply the two missing lines of code.

```python
def secondMax(L):
    m = max(L)
    k = L[0:L.index(m)] + L[L.index(m)+1:]
    return max(k)
```

```