1.5 Write a procedure `merge(n1, n2)` which takes numbers with digits in decreasing order and returns a single number with all of the digits of the two, in decreasing order. Any number merged with 0 will be that number (treat 0 as having no digits). Use recursion.

Hint: If you can figure out which number has the smallest digit out of both, then we know that the resulting number will have that smallest digit, followed by the merge of the two numbers with the smallest digit removed.

```
def merge(n1, n2):
    """ Merges two numbers
    >>> merge(31, 42)
    4321
    >>> merge(21, 0)
    21
    >>> merge (21, 31)
    3211
    """
```

```
if n1==0:
    return n2
elif n2==0:
    return n1
elif n1%10 < n2%10:
    return 10*merge(n1//2, n2) + n1%10
else:
    10*merge(n1, n2//2)+n2%10
```

base cases

merge(31, 42)
4321
10(merge(3,42)) + 1
432
merge(3, 42)
43
10(merge(3,4)) + 2
merge(3,4)
4
10(merge(0,4)) + 3
4

1) base case
2) recursive statement
3) return?
4) anything else?

```
def func1(n):
    if (n==100):
        return True
    return func1(n+1)
```

recursion

```
def rec(n):
    if (n==0):
        return None
    return rec(n-1)
```

→ 1) base case
2) recursion statement

```
def iter(k):
    while (k>0):
        k-=1
    return none
```