CSC236 tutorial exercises, Week #8
best before Friday afternoon

For each of the algorithms below, prove termination. If proving termination requires a loop invariant, you may state it without proof. (Though you should be confident that your invariant actually holds, and comfortable with how it *could* be proved, if necessary.)

1.

```
def ssum(A):
    """Pre: A is a list of non-negative ints
    Post: return the sum of A
    WARNING: A may be irrerversibly altered!
    """
    i = 0
    s = 0
    while i < len(A):
        if A[i] == 0:
            i += 1
        else:
            s += 1
            A[i] -= 1
    return s
```

2.

```
def binsearch(A, x):
    """Pre: A is a sorted list of numbers. x is a number.
    Post: return i such that A[i] = x, or -1 if x is
          not an element of A.
    """
    lo = 0
    hi = len(A)-1
    while lo <= hi:
        m = lo + (hi - lo) // 2
        mid = A[m]
        if mid == x:
            return m
        elif mid < x:
            lo = m + 1
        else:
            hi = m - 1
    return -1
```

3.

```python
def perambulate(A):
    """Pre: A is a non-empty list of non-negative ints
    """
    seen = []
    curr = A[0]
    i = 0
    while curr not in seen:
        i = (i + curr) % len(A)
        seen.append(curr)
        curr = A[i]
    return curr
```