Prob. 1	Prob. 2	Prob. 3	Prob. 4

Problem 1.

This is a paragraph

Problem 2.

Problem 3.

Problem 4.

1.

After 15 pushes followed by 7 pops the stack will be in the following state :

S_0 8
S_1 7
S_2 6 5
S_3 4 3 2 1

2.

The worst case scenario for pop is for any given integer k when S_0 contains one element and for i from 1 to k S_i is half full. In that state a pop will empty S_0 which will then take elements from S_1 emptying it in the process. S_1 will then take elements from S_2 emptying it in the process. This so empties every S_i until the last one resulting in a number of operation equal to the number of elements in the stack. We can also realize that the resulting state is the worst case scenario for a push operation.

The worst case for push is for any given integer k when for i from 0 to k S_i is completely full. In that state pushing an element in the stack will overload S_0 which will then push its elements on S_1 overloading it in the process. S_1 will then push its elements on S_2 overloading it in the process. This so overload every S_i until the last one resulting in a number of operation equal to the number of elements in the stack. We can also realize that the resulting state is the worst case scenario for a pop operation.

The worst case scenario for pop and push are both in linear $O(n)$ time.

3.

Let k be any integer and $n = 2^k - 1$.

Initial state : Empty stack

Operation sequence :

1. n pushes
2. n times a push followed by a pop
3. n pops

The second part of the proposed sequence of operations consist of n worst case pushes and n worst case pops. That makes $O(n^2)$ basic operations. We can then conclude that the two operations can't be achieved in constant amortized time.