```
void flip(StackOfT& s)
//! updates s
//! ensures s = rev(#s)
```

Name: _____

Name: _____

One CM: _____

To do: Fill out this reasoning table

S	Code	Assume		Confirm
0				
	`StackOfT t;`			
1			Unchanged	

	`while(s.length() > 0) {`
	`//! updates s, t`
	`//! maintains`
	`//! rev(t) * s =`
	`//!   rev(#t) * #s`
	`//! decreases |s|`			

2			Unchanged	
	`  T y;`			
3			Unchanged	
	`  s.pop(y);`			
4			Unchanged	
	`  t.push(y);`			
5			Unchanged	
	`}`			
6				
	`s.transferFrom(t)`			
7				

Some of the Stack operations

```
template <class T>
class Stack1
    //! is modeled by string of T
    //! exemplar self
{
public: // Standard Operations
    Stack1 ();
        //! replaces self
        //! ensures: self = <>
    void transferFrom (Stack1& source);
        //! replaces self
        //! clears source
        //! ensures: self = #source
    void push (T& x);
        //! updates self
        //! clears x
        //! ensures: self = <#x> * #self
    void pop (T& x);
        //! updates self
        //! replaces x
        //! requires: self /= <>
        //! ensures: <x> is prefix of #self  and  self = #self[1, |#self|)
    Integer length (void);
        //! restores self
        //! ensures: length = |self|
}
```