

Jack Thake

CS163 Test Plan

Develop the test plan: For each member function that you plan to write, think about how to test it – what flow of control exists in the member function and how would you test out all conditions:

Test Case(s)	Expected Result	Verified? (yes/no)
CS-alert class		
display → populated object	successfully print alert	
display → unpopulated object	don't print, return FAILURE	
operator ==	should check equality	
operator = assign populated to empty	should successfully deep copy	
operator = assign empty to populated	should deallocate populated object making it empty	
Cs-alert-stack class		
push → valid object	should successfully add to stack	
push → invalid object	should not attempt to push	
pop → populated stack	successfully pop item	
pop → unpopulated stack	shouldn't attempt to pop return FAILURE	
display → populated stack	successfully print	
display → unpopulated stack	should fail, don't attempt to print	

Verify correctness: Using the above test plan, create a test program that tests the interactions of all functions together.