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_project_manager		
add_priority(1)	should add a list of priority 1	
remove_priority(1)	should remove list of priority 1	
remove_priority(non-existing)	should return FAILURE	
add_project(1, project)	should add project to the first priority list	
add_project(non-existing, project)	should return FAILURE	
remove_project(valid name)	should remove the corresponding project	
display_priority(valid priority)	should print out that priority	
display_priority(non-existing)	should return FAILURE	
display_all() -> initialized list	should print everything	
display_all() -> uninitialized list	should return FAILURE	
CS_project_list		
add_item(project)	should add the item to the list	
remove_item(valid name)	should remove the item	
remove_item(invalid name)	should return FAILURE	
display() -> initialized list	should print everything	
display() -> uninitialized list	should return FAILURE	

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