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)
list_manager::list_manager(0)	don't create any lists	
list_manager::list_manager(5)	create 5 lists	
list_manager::add_item(NULL)	return false, don't attempt to add	
list_manager::add_item(Project Struct)	return true, add item to list	
list_manager::remove_item(name)	return true, remove item	
list_manager::remove_item(NULL)	return false don't attempt to remove	
list_manager::remove_item(non-existant)	return false don't attempt to remove	
list_manager::display_priority(1)	return true, print priority 1	
list_manager::display_priority(0)	return false, don't print	
list_manager::display_priority(non-existant)	return false, don't print	
list_manager::display_all() → no lists	return false, don't print	
list_manager::display_all() → more than 1 list	return true, print all items	
project_list::add_item(NULL)	return false, don't add	
project_list::add_item(project_struct)	return true, add item	
project_list::display_all() → initialized list	return true, print the list	
project_list::display_all() → empty list	return false, don't print	
project_list::remove_item(NULL)	return false, don't remove	
project_list::remove_item(non-existant)	return false, don't remove	
project_list::remove_item(name)	return true, remove item	

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