

## 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)
add_animal - empty table	added to place in table, return 1 if valid, 0 if not.	X
add_animal - overflowing table	added to place in chain, return 1 if valid, 0 if not.	X
load_from - empty file	return 0	
load_from - one entry	load data, return 1, place in table	
load_from - many entries	load data, return 1, place in table	X
remove animal - empty list	return 0	
remove animal - 1 item	clear list, return 1.	
remove animal - many items	remove item, return 1.	
remove animal - no match	return 0.	
remove animal - match found	remove node, and return 1	
retrieve animal - same as above	Same as above, but pass by reference instead of remove	
display animal - same as above	Same as above, but display instead of remove.	
remove_all - empty and full	return 0 if empty, delete and return 1 if full.	
display favorites, 0, 1,full	return 1, display all. 0 if empty.	
clear favorites 0, 1,full	return 1, delete list, 0 if empty.	
add_favorite, 0, 1,full	return 1 and enqueue. 0 if not valid.	
fetch_favorite 0,1, full, head, rear	report 1 and return object by reference. return 0 if no match, or empty. Dequeue object	
is_empty - empty, full	return 1 if empty, 0 if full	
peek favorite - empty, full	return 1 if empty, 0 if full and pass object by reference.	

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