
Laboratory 5 : Cover Sheet

Name _____ Date _____

Section _____

Place a check mark in the *Assigned* column next to the exercises your instructor has assigned to you. Attach this cover sheet to the front of the packet of materials you submit following the laboratory.

Activities	Assigned: Check or list exercise numbers	Completed
Implementation Testing	✓	
Programming Exercise 1		
Programming Exercise 2		
Programming Exercise 3		
Analysis Exercise 1		
Analysis Exercise 2		
	Total	

Laboratory 5: Implementation Testing

Name _____ Date _____

Section _____

Check with your instructor whether you are to complete this exercise prior to your lab period or during lab.

Test your implementation of the List ADT using the test program in the file *test5.cpp*. This program allows you to interactively test your implementation of the List ADT using the following commands.

Test Plan 5-1 (operations in the List ADT)			
Test case	Commands	Expected result	Checked
Insert at end	+a +b +c +d	a b c d	
Travel from beginning	< N N	a b c d	
Travel from end	> P P	a b c d	
Delete middle data item	-	a c d	
Insert in middle	+e +f +f	a c e f f d	
Remove last data item	> -	a c e f f	
Remove first data item	-	c e f f	
Display data item	@	Returns c	
Replace data item	=g	g e f f	
Clear the list	C	Empty list	

Note: The data item marked by the cursor is shown in **bold**.

Test Plan 5-2 (List ADT operations using integers)			
Test case	Commands	Expected result	Checked
Insert at end	+1 +2 +3 +4	1 2 3 4	
Travel from beginning	< N N	1 2 3 4	
Travel from end	> P P	1 2 3 4	
Delete middle data item	-	1 3 4	

Note: The data item marked by the cursor is shown in **bold**.

Laboratory 5: Programming Exercise 1

Name _____ Date _____

Section _____

Test Plan 5-3 (slide show)	
Test case	Checked
Slide show is in the file <i>slides.dat</i> . The full 14 slides should display properly. Examine the data to determine how long each slide should display. Verify that those that should display for longer do so.	

Laboratory 5: Programming Exercise 2

Name _____ Date _____

Section _____

Test Plan 5-4 (moveToBeginning operation)			
Test case	Commands	Expected result	Checked
Set up list	+a +b +c +d	a b c d	
Move last data item	M	d a b c	
Move second data item	N M	a d b c	
Move third data item	N N M	b a d c	

Note: The data item marked by the cursor is shown in **bold**.

Laboratory 5: Programming Exercise 3

Name _____ Date _____

Section _____

Test Plan 5-5 (insertBefore operation)			
Test case	Commands	Expected result	Checked
Set up list	+a +b +c	a b c	
Insert in middle	#d	a b d c	
Cascade inserts	#e	a b e d c	
Insert after head	P #f	a f b e d c	
Insert as head	P #g	g a f b e c	

Note: The data item marked by the cursor is shown in **bold**.

Laboratory 5: Analysis Exercise 1

Name _____ Date _____

Section _____

Given a list containing N data items, develop worst-case, order-of-magnitude estimates of the execution time of the following List ADT operations, assuming they are implemented using a linked list. Briefly explain your reasoning behind each estimate.

insert $O(\quad)$

Explanation:

remove $O(\quad)$

Explanation:

gotoNext O()

Explanation:

gotoPrior O()

Explanation:

Laboratory 5: Analysis Exercise 2

Name _____ Date _____

Section _____

Part A

Programming Exercise 3 introduces a pair of approaches for implementing an insertBefore operation. One approach is straightforward, whereas the other is somewhat less obvious but more efficient. Describe how you might apply the latter approach to the remove operation. Use a diagram to illustrate your answer.

Part B

The resulting implementation of the remove operation has a worst-case, order of magnitude performance estimate of $O(N)$. Does this estimate accurately reflect the performance of this implementation? Explain why or why not.