Laboratory 1: 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 1: 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 Text ADT using the program in the file *test1.cpp*. This program supports the following tests.

Lab 1 Online Test Plans	
Test	**Action**
1-1	Tests the constructors.
1-2	Tests the length operation.
1-3	Tests the subscript operation.
1-4	Tests the assignment and clear operations.

Test Plan 1-1 (constructors)			
Test case	**String**	**Expected result**	**Checked**
Simple string	`alpha`	`alpha`	
Longer string	`epsilon`		
Single-character string	`a`		
Empty string	*empty*		

Test Plan 1-2 (length operation)			
Test case	String	Expected length	Checked
Simple string	`alpha`	`5`	
Longer string	`epsilon`		
Single-character string	`a`		
Empty string	*empty*		

Test Plan 1-3 (subscript operation)			
Test case	*n*	Expected character	Checked
Middle character	2	p	
First character	0		
Last character	4		
Out of range	10		

Test Plan 1-4 (assignment and clear operations)			
Test case	Assignment statement	Expected result	Checked
Simple assignment	`assignStr = alpha;`	`alpha`	
Single-character string	`assignStr = a;`		
Empty string	`assignStr = empty;`		
Source string longer than destination buffer	`assignStr = epsilon;`		
Assign to self	`assignStr = assignStr;`		

Check assignment by clearing destination	`assignStr = alpha;`
	`assignStr.clear();`		

Laboratory 1: Programming Exercise 1

Name ______________________________ Date ____________________

Section ______________________________

Test Plan 1-5 (lexical analysis program)		
Test case	Expected result	Checked
Program in the file *progsamp.dat*		

Laboratory 1: Programming Exercise 2

Name ______________________________ Date ____________________

Section ______________________________

Test Plan 1-6 (toUpper and toLower operations)			
Input	**toUpper output**	**toLower output**	**Checked**
InigoMontoya (*)	INIGOMONTOYA	inigomontoya	
a			
My name is Inigo Montoya.			

* Inigo Montoya is a character in the movie *The Princess Bride*, 1987.

Laboratory 1: Programming Exercise 3

Name ______________________________ Date ____________________

Section ______________________________

Test Plan 1-7 (relational operations)

Test case	Pair of strings	Expected result	Checked
Second string greater	`alpha  epsilon`		
First string greater	`epsilon  alpha`		
Identical strings	`alpha  alpha`		
First string embedded in second	`alp  alpha`		
Second string embedded in first	`alpha  alp`		
First string is a single character	`a  alpha`		
Second string is a single character	`alpha  a`		
First string is empty	*empty* `alpha`		
Second string is empty	`alpha` *empty*		
Both strings are empty	*empty* *empty*		

Laboratory 1: Analysis Exercise 1

Name ______________________________ Date ____________________

Section ______________________________

A full-page version of this exercise with space for writing in answers is available in the online supplements for Lab 1.

Part A

What are the implications of having no destructor in a class like Text that does dynamic memory allocation? What are the practical consequences of not having a destructor for these classes in a long-running program?

Part B

What other operators might it make sense to overload in the Text class? Name four and briefly describe how they would work.

Part C

Are there any operators that it does not make sense to overload in the Text class? Why not?

Laboratory 1: Analysis Exercise 2

Name ____ Date ______________________

Section ______________________________________

Part A

Design another method for the Text ADT and give its specification below. You need not implement the method, simply describe it.

Function prototype:

Requirements:

Results:

Part B

Describe an application in which you might use your new method.