CS333 Project5 **#README**

Tracy Quan

Platform: macOS version 10.12.6

Directory Layout:

Project4/

|

| |\_\_/CSemantics.c

| |\_\_/CSemantics

| |\_\_/README.docx

|

|\_\_/JavaScript/

| |

| |\_\_/Task1.html

| |\_\_/Task2.html

| |\_\_/Task3.html

| |\_\_/Extension2.html

| |\_\_/Extension3.html

|

|

|\_\_/Bash/

| |

| |\_\_/Task1.sh

| |\_\_/Task2.sh

| |\_\_/Task3.sh

|

|\_\_/C++/

| |

| |\_\_/Task1.cpp (Extension1.1)

| |\_\_/Task1

| |\_\_/Task2.cpp (Extension1.2)

| |\_\_/Task2

**Part I: Polymorphism in C**

Create a generic linked list class in C.

**Command line:**

**(Basic version)**

gcc -o clltest clltest.c MyLinkedList.c

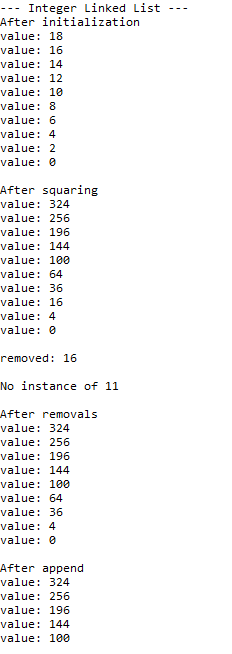
./clltest

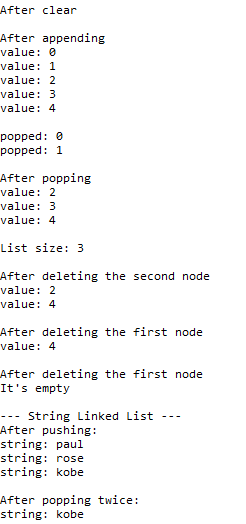
**(Two data types and extensions)**

gcc -o clltest\_ext clltest\_ext.c MyLinkedList\_ext.c

./clltest\_ext

**Output:**





**Part II: Selected Language**

**1. JavaScript**

**Task1:**

Implement a LinkedList class in your other two languages, making use of some form of polymorphism to implement the data structure. Create the same set of functions as above, but adapt the data types, as necessary. If your language enables templates, use them. Create a test function that demonstrates you can create linked lists to store two different types of data.

**Instruction:**

Open the Task1.html

**Output:**

**2. C++**

**Task1:**

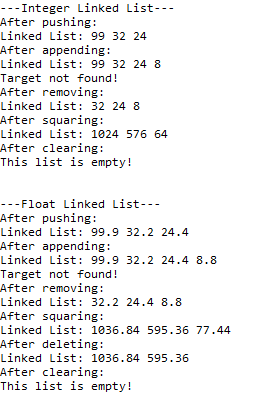
Implement a LinkedList class in your other two languages, making use of some form of polymorphism to implement the data structure. Create the same set of functions as above, but adapt the data types, as necessary. If your language enables templates, use them. Create a test function that demonstrates you can create linked lists to store two different types of data.

**Instruction:**

g++ -o Task1 Task1.cpp

./Task1

**Output:**



***Part III: Extensions***

**Extension1.1**