Name: Dennis Wu and Barry Ibarra

Project: Programming Assignment #1

Course: CPSC131

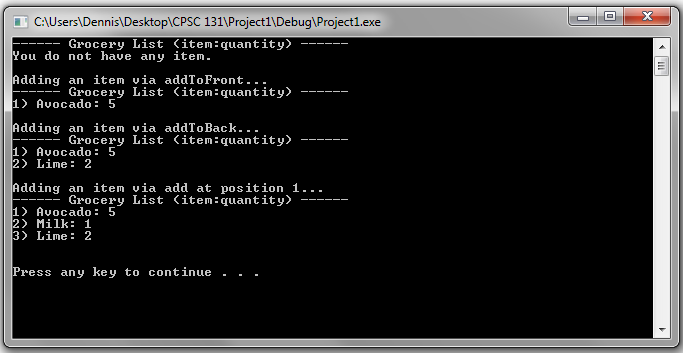
Professor: Dr Abhishek Verma

Pseudo code for Add function

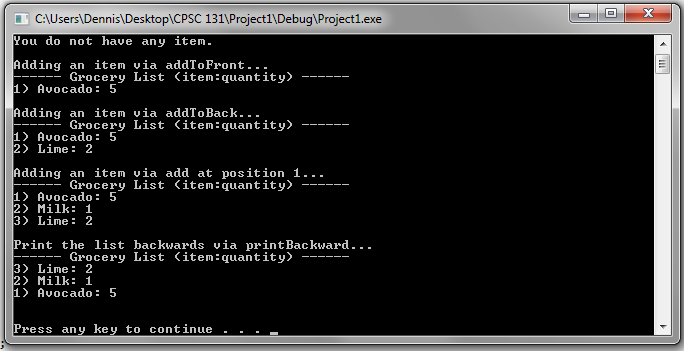
Function: add(string itemName, real quantity, integer position)  
Input: itemName, quantity, position  
Return: nothing  
 INIT new item node with itemName, quantity  
 IF list is empty OR position is 0   
 THEN   
 CALL addToFront with itemName, quantity  
 ELSE IF position >= total count of items   
 THEN  
 CALL addToBack with itemName, quantity  
 ELSE  
 INIT current = head  
 FOR i =1 step 1 to position  
 CALL goNext with current  
 END FOR  
 SET new previous item = previous item  
 SET new next item = current item  
 SET previous next item = new item  
 SET previous current item = new item  
 END IF  
 INCREMENT itemCount

Screenshot of add and print backward function

void add(string itemName, double quantity, int position)



void printBackward() const;

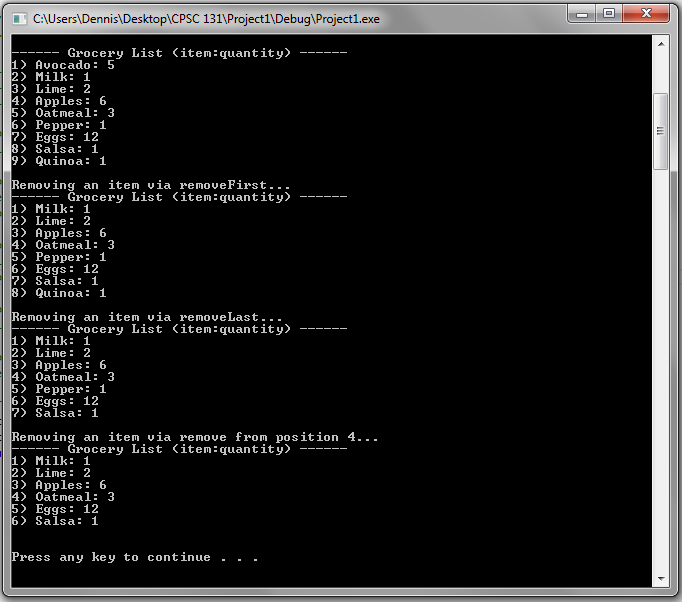


Pseudo code for remove function

Function: remove(integer position)  
Input: position  
Return: nothing  
 INIT lastItem to itemCount - 1  
 IF list <> empty AND position < itemCount  
 THEN   
 INIT current = head  
 IF position is 0 AND itemCount is 1  
 THEN  
 SET head, tail to NULL  
 ELSE   
 IF position is 0  
 THEN  
 CALL removeFirst  
 ELSE IF position is lastItem  
 CALL removeLast  
 ELSE   
 FOR i = 1 step 1 to position  
 CALL goNext with current  
 END FOR  
 SET previous next item = next current item  
 END IF  
 END IF  
 END IF  
 DECREMENT itemCount

Screenshot of remove function

void remove(int position)



Pseudo code for peek and look up function

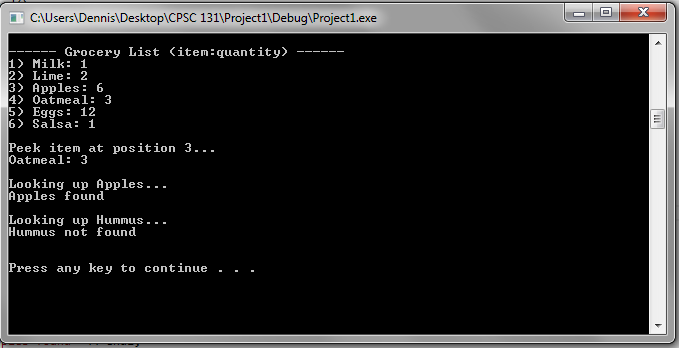
Function: peek(integer position)  
Input: position  
Return: nothing  
 IF position < itemCount AND position >= 0  
 THEN  
 INIT current = head   
 FOR i = 1 step 1 to position  
 CALL goNext with current  
 END FOR  
 PRINT itemName, qauntity  
 END IF

Function: lookup(string itemName)  
Input: position  
Return: true or false  
 INIT current = head  
 WHILE current NOT EQUAL empty  
 DO   
 IF current item name = itemName  
 THEN  
 lookup <- TRUE  
 END IF  
 CALL goNext with current  
 END WHILE  
 lookup <- FALSE

Screenshot of peek and look up function

void peek(int pos) const;

bool lookup(string itemName) const;



Pseudo code for deal function

Function: deal(GroceryList secondList)  
Input: second grocery list  
Return: nothing  
 INIT current = head  
 INIT numSplit = itemCount/2  
 IF itemCount NOT EQUAL 1  
 THEN  
 IF itemCount is 2  
 THEN  
 CALL goNext with current  
 CALL secondList.addToFront with itemName, quantity  
 CALL removeLast  
 ELSE  
 FOR i = 1 step 1 to numSplit  
 CALL goNext with current  
 CALL secondList.addToBack with itemName, quantity  
 CALL goNext with current  
 CALL remove with i+1  
 END FOR  
 END IF  
 END IF

Screenshot of deal function

void deal(GroceryList & secondList)

