2460681S – Wesley Scott

insertTail

A picture containing indoor, wall, whiteboard, white

Description automatically generated

algorithm: no return, takes E elem

create a new node (called newNode)

if the first node in list is null:

create a new node (called first), assigned to first

node is given a generic element and its next is the list first/head

return yielding nothing

newNode’s next element is set to null

create a new node (called last)

set last equal to the first element in the list

while the next node of last is not null (traverse list)

set last to its next value

set last next to newNode

return yielding nothing

end

deleteAlternate

A picture containing indoor

Description automatically generated

algorithm: no return, no args

if the first element in list is null return yielding nothing

create new node called pred

assign pred equal to the first element in the list

create new node called curr

assign curr equal to the next element in the list (after first)

while pred and curr are not null

change the next link of previous node (set pred’s next to curr’s next)

clear the current node (set curr to null)

update previous and current nodes

(set pred equal to the next of pred)

If pred is not null set curr to pred’s next

end

merge

A close up of text on a white background

Description automatically generated

algorithm: returns linked list, takes 2 lists

create a new linked list called mergedList

create a new node (a), set equal to the first element the list method was called on (this.)

create a new node (b), set equal to the first element of argument list1

create a new node (c), set equal to the first element of argument list2

while a, b or c is not null

if b and c are larger than a or null

insert a into mergedList

set a to next unique element in its list

if a and c are larger than b or null

insert b into mergedList

set b to next unique element in its list

if a and b are larger than c or null

insert c into mergedList

set c to next unique element in its list

if a and b are equal and a is less than c

insert a or b into mergedList

set a to next unique element in its list

set b to next unique element in its list

if a and c are equal and a is less than b

insert a or c into mergedList

set a to next unique element in its list

set c to next unique element in its list

if b and c are equal and b is less than a

insert b or c into mergedList

set b to next unique element in its list

set c to next unique element in its list

if a, b and c are equal

insert either a,b or c into mergedList

set a to next unique element in its list

set b to next unique element in its list

set c to next unique element in its list

set this list’s first element (a) to the first element of mergedList

(return mergedList)