Lab 1 Assignment

Simple Linked List

You are given

- A solution file containing
  - The Linked List class mentioned in lecture
    - LinkedList.h
    - LinkedList.cpp
  - A main file to use the Linked List:
    - main.cpp
- In each assignment, you should only modify ONLY ONE file
  - In this one, it is “LinkedList.cpp”

main()

- But let's look at what are we supposed to do first

```
int main()
{
    List l;
    l.insertHead(123);
    l.insertHead(11);
    l.insertHead(9);
    l.insertHead(1);
    l.insertHead(20);

    for (int i = 0; i < 5; i++) {
        cout << "The current list is: ";
        l.print();
        cout << "Does 9 exist in the list?" << (l.exist(9) ? "Yes" : "No") << endl;
        l.removeHead();
    }
    return 0;
}
```

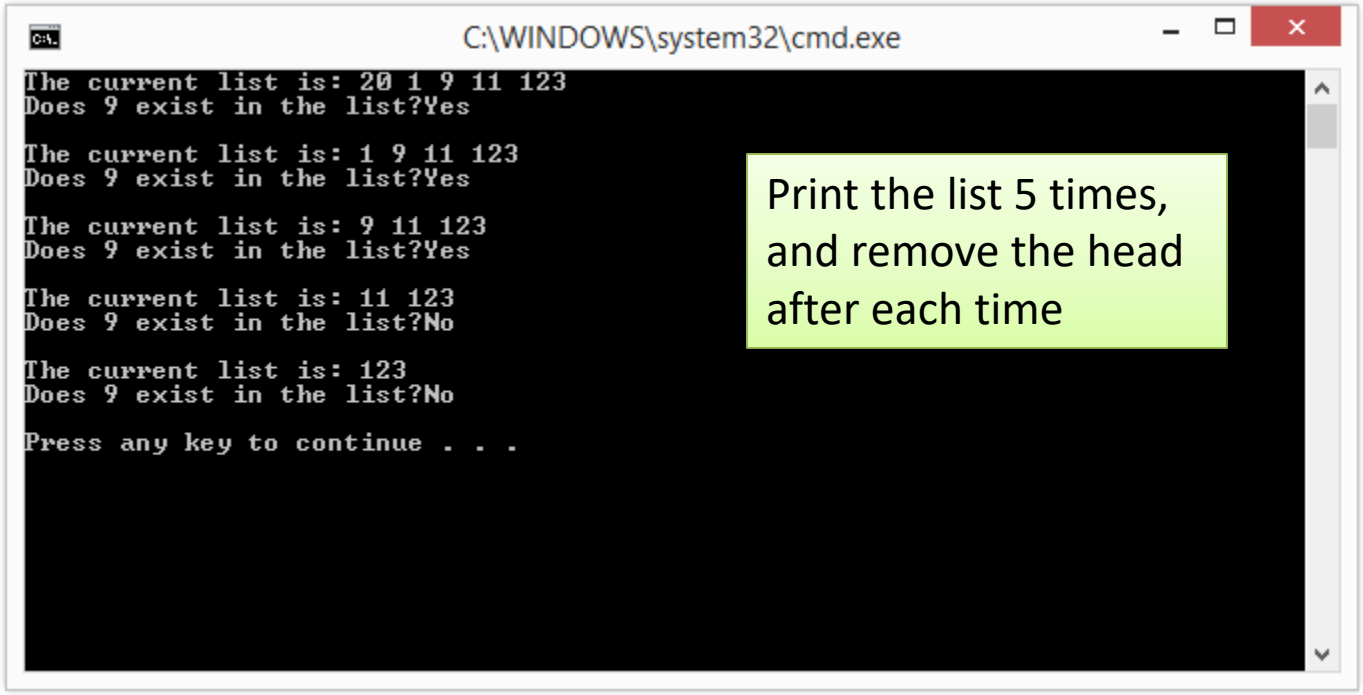
Already implemented

Your **first** task: implement a function to print the list

Second task: implement the function "exist" to ask if a number is in the list

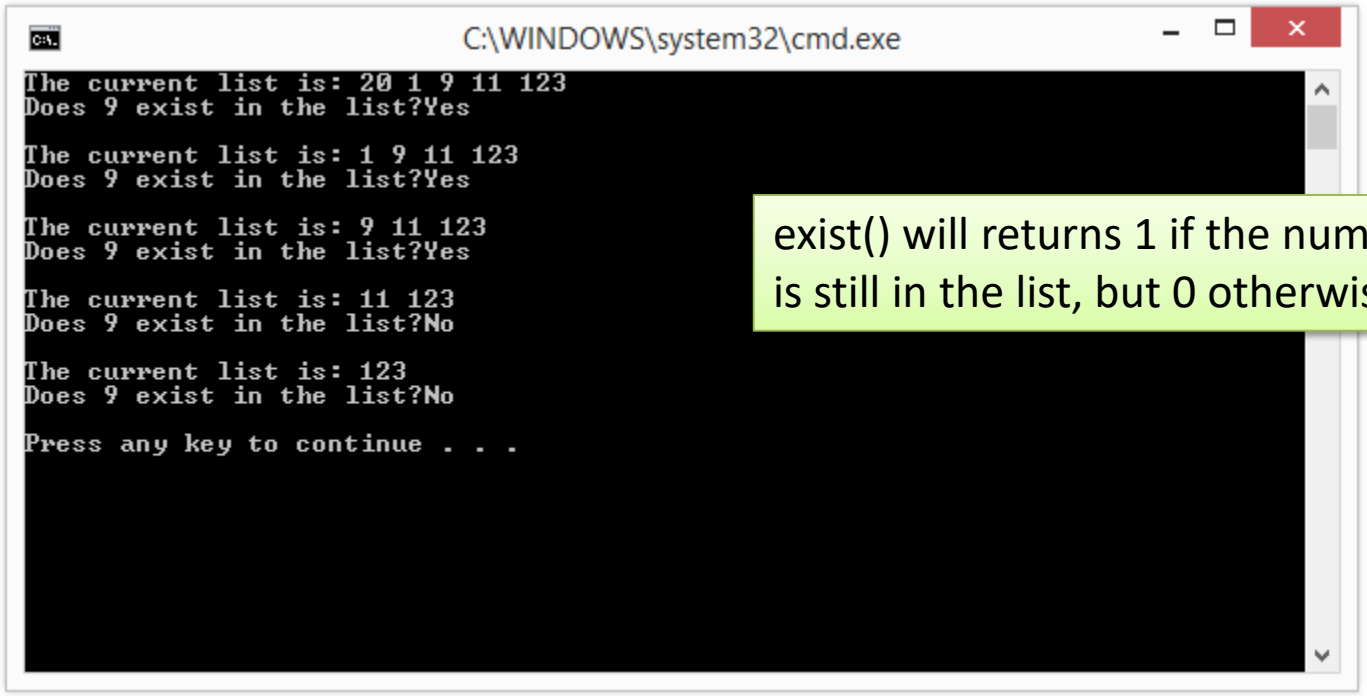
The Method print()

```
for (int i = 0; i < 5; i++) {
    cout << "The current list is: ";
    l.print();
    cout << "Does 9 exist in the list?" << (l.exist(9) ? "Yes" : "No") << endl;
    l.removeHead();
}
return 0;
```



The Method exist()

```
for (int i = 0; i < 5; i++) {
    cout << "The current list is: ";
    l.print();
    cout << "Does 9 exist in the list?" << (l.exist(9) ? "Yes" : "No") << endl;
    l.removeHead();
}
return 0;
```



exist() will returns 1 if the number is still in the list, but 0 otherwise

(? :) Syntax

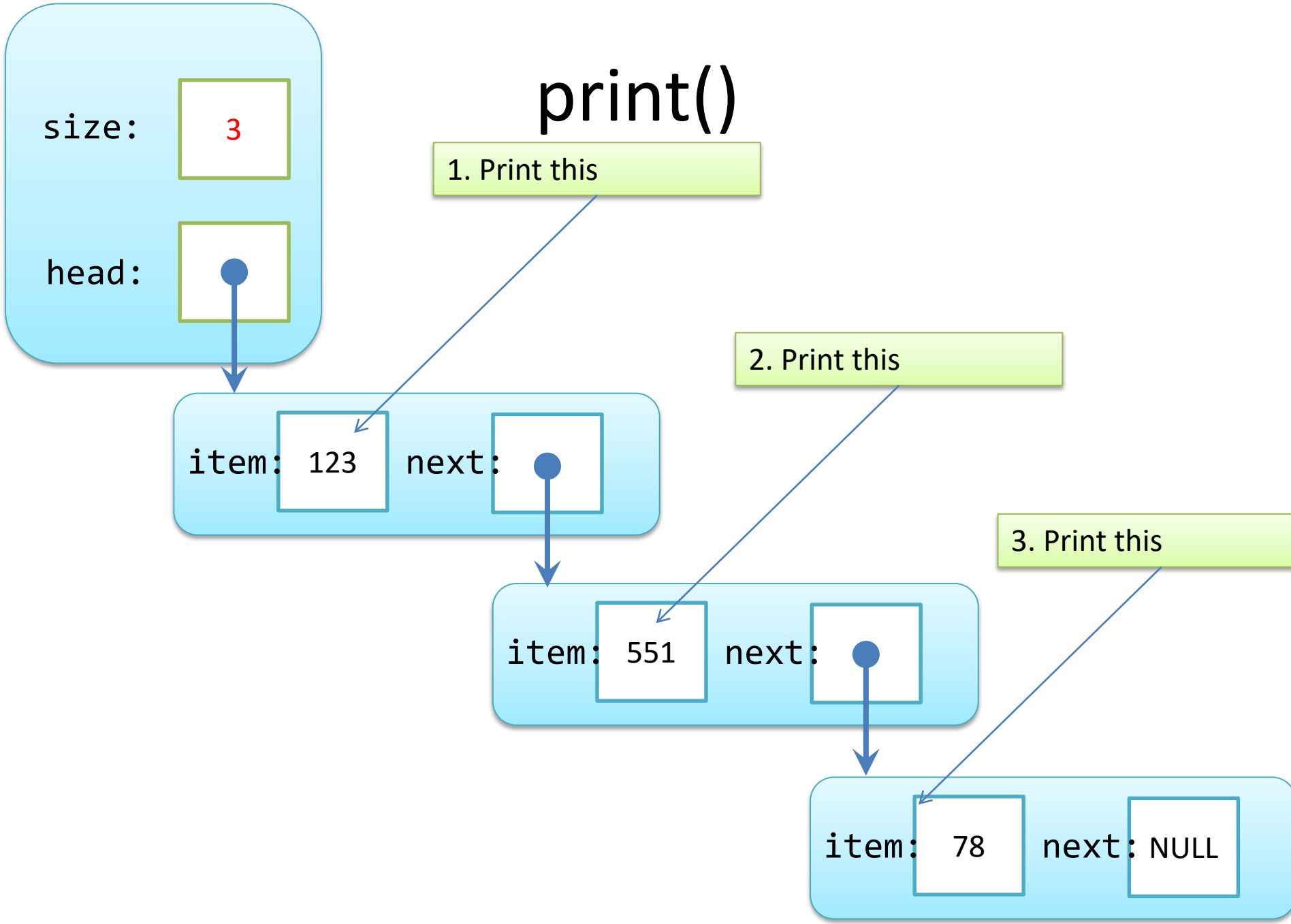
- It is a syntax in C/C++
- we can write an expression like this:

```
( v ? a : b)
```

- If v is not 0, the expression is equal to a
  - otherwise b

How to Implement print()

- Idea?
  - Print the items in the linked list one by one
  - starting from the head to the tail


print()
size:
3
head:
1. Print this
item:
123
next:
2. Print this
item:
551
next:
3. Print this
item:
78
next:
NULL

Starting From the Head…

- But what if the head == NULL?
  - no printing
- If the head is not NULL
  - print it and move to the next node
  - then print it and move to the next node
  - then print it and move to the next node
  - .
  - .
  - until NULL

How many times?

The Method print()

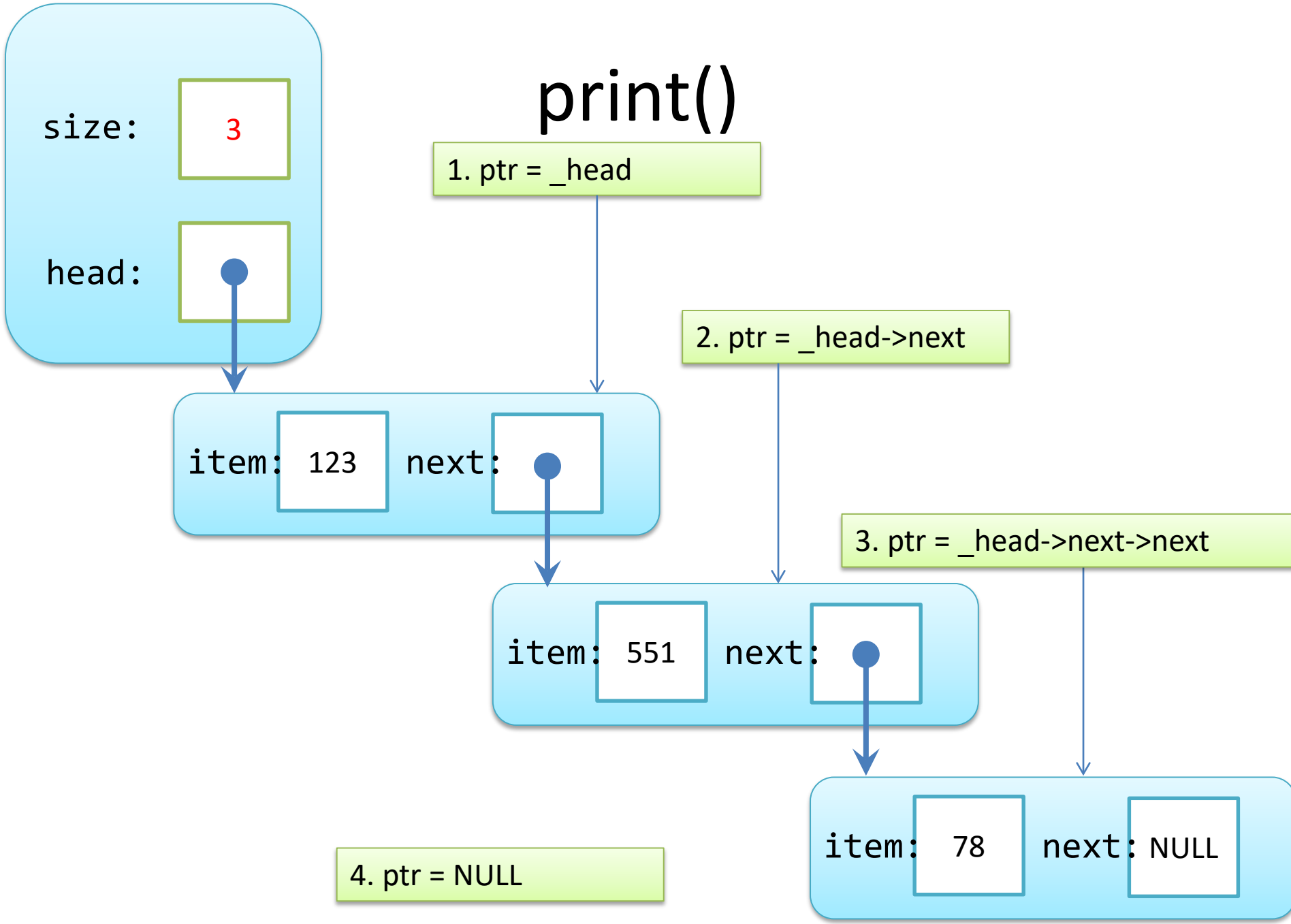
```
void List::print() {
  ListNode* ptr = _head;

  for (int i=0;i<_size;i++) {
    cout << ptr._item;
    ptr = ptr->_next;
  }

  cout << endl;
}
```

- What if we do not want to create an integer i?

The Method print()

- Looping a linked list with a pointer

```
void List::print() {
  ListNode* ptr = _head;
  while (ptr) {
    cout << ptr->_item << " ";
    ptr = ptr->_next;
  }
  cout << endl;
}
```


print()
size:
3
head:
1. ptr = _head
2. ptr = _head->next
3. ptr = _head->next->next
item:
123
next:
item:
551
next:
item:
78
next:
NULL
4. ptr = NULL

The Method print()

- Shorter

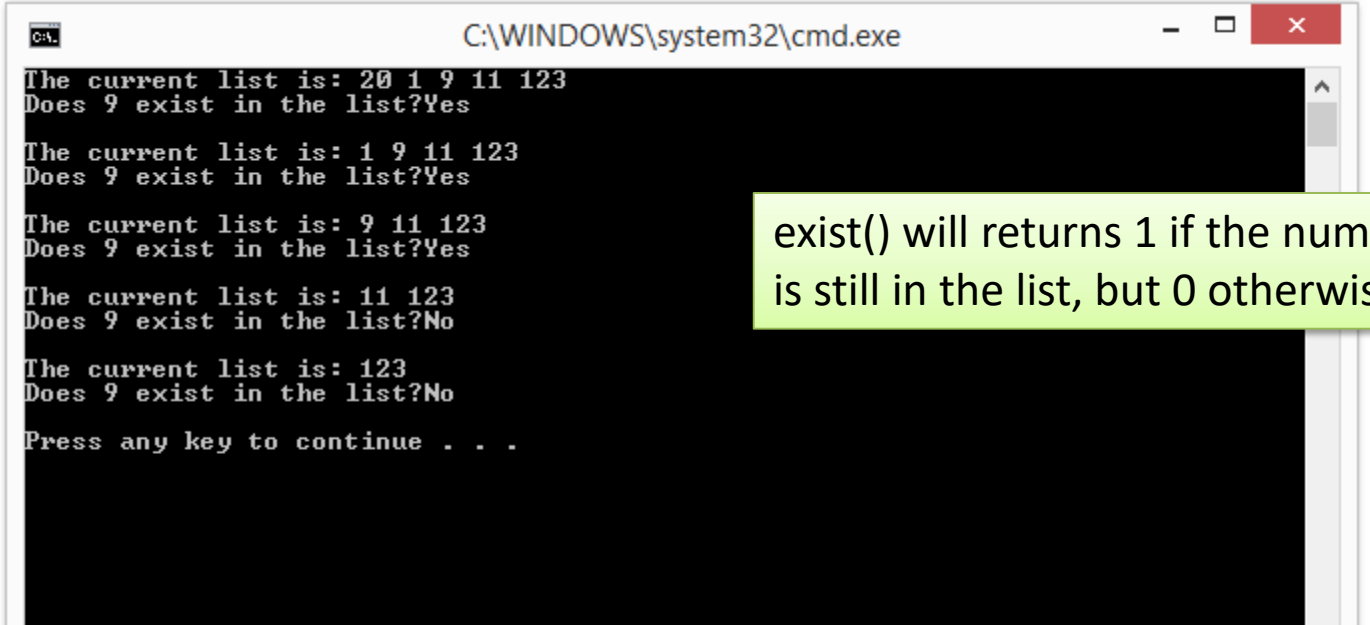
```
void List::print() {
  ListNode* ptr = _head;
  for (; ptr; ptr = ptr->_next)
    cout << ptr->_item << " ";
  cout << endl;
}
```

- Even shorter

```
void List::print() {
  for (ListNode* ptr=_head;ptr;ptr=ptr->_next)
    cout << ptr->_item << " ";
  cout << endl;
}
```

Your Assignment

- Implement the function “exist”
  - e.g. exist(9)
    - will return 1 if 9 is in the list
    - return 0 if 9 is NOT in the list



Additional Tasks (Not Graded)

- **Other Member Functions of class List**
- You can add and implement the following functions:
  - `headItem()` that will return the first item in the list if the list is not empty.
  - `empty()` that will return 1 if the list has nothing, and 0 otherwise.
  - `tailItem()` that will return the last item in the list if the list is not empty.
  - `removeTail()` that will removes the last item of the list if the list is not empty.

Additional Tasks (Not Graded)

- **Create A Project of Your Own**
  - You can try to create a MSVS solution or XCode project of your own. You can use the .h and .cpp files in this assignment, or some finished assignments from your previous courses. Additionally, you could try to divide your previous code into .h and .cpp files.