

5-1) List Quiz

Due No due date **Points** 4 **Questions** 4

Available Sep 22 at 12am - Sep 22 at 11:59pm 1 day

Time Limit None

This quiz was locked Sep 22 at 11:59pm.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	4 minutes	1 out of 4

Score for this quiz: **1** out of 4

Submitted Sep 22 at 12:47pm

This attempt took 4 minutes.



Question 1

1 / 1 pts

Why have I used sentinel nodes in our list class?

Correct!

- ☒ To make it safer
- ☐ To make it smaller
- ☐ To make it faster
- ☐ To make it do more

They are there specifically to make sure Head and Tail are never null, so you never have to check and can't crash that way.

Question 2**0 / 1 pts**

Because we have next and prev pointers in each node, we can quickly access any node.

You Answered☒ True**Correct Answer**☐ False

Vector's array allows instant indexed access. List has to start at Head and loop through next pointers until it gets to the right spot.

We only have hooks for the ends of the chain.

**Question 3****0 / 1 pts**

Because our node is an "inner class", the user must qualify the name when they use it. (`List<>::ListNode`)

You Answered☒ True**Correct Answer**☐ False

They would if they were allowed to, but the node is private so the user can't even see it. We don't want them to know how we work.

Question 4

0 / 1 pts

I have two pointers, A and B. They are each pointing at valid nodes. The next and prev pointers of those nodes are correctly set up to make A and B adjacent.

I want to put a new node pointed at by C in between those two. What is wrong with the code below?

```
// In an Insert method
ListNode *C = new ListNode;
A->mNext = C;
B->mPrev = C;
```

you Answered

☒ mNext and C aren't the same type

Correct Answer

- ☐ C's node doesn't point at anything
- ☐ A or B could be null and could crash
- ☐ Next and Prev don't have new's

A) Yes they are. ListNode pointers

C) Because of sentinel nodes, we know for sure they are not null

D) One new per object. Draw this process, and you'll see you only draw one new box. Next and Prev are pointers pointing at existing nodes, which is fine.

Quiz Score: **1** out of 4

