



CS 225

Data Structures

August 26 – Lists and List ADT

G Carl Evans



PotD – Problems of the Day

- Right now, about 1 per lecture all due by start of September 6th
- Later one each weekday starting at midnight due midnight the next day.
- 1 point of extra credit for each problem with 100% to a maximum of 40 points
- Approximately 60 problems released during the semeste



Exams

- 6 Exams 60 points each
- Exam dates <https://courses.engr.illinois.edu/cs225/fa2022/exams/>
- Exam 0 starts next week
- Exam 0 practice on PL now



List ADT



What types of “stuff” do we want in our list?

--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--



Templates



template1.cpp

```
1  
2  
3 T maximum(T a, T b) {  
4     T result;  
5     result = (a > b) ? a : b;  
6     return result;  
7 }
```

List.h

```
1 #pragma once
2
3
4 class List {
5     public:
6
7
8
9
10
11
12
13
14     private:
15
16
17
18 };
19
20
21
22
```

List.hpp

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
```




List Implementations

1.

2.

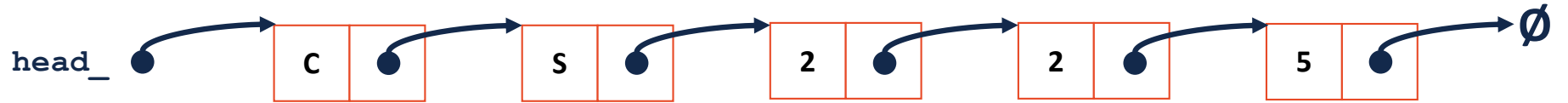
Linked Memory



List.h

```
28 class ListNode {
29     T & data;
30     ListNode * next;
31     ListNode(T & data) : data(data), next(NULL) { }
32 };
```

Linked Memory



List.h

```
1 #pragma once
2
3 template <class T>
4 class List {
5     public:
6     /* ... */
7
8     private:
9     class ListNode {
10         T & data;
11         ListNode * next;
12         ListNode(T & data) :
13             data(data), next(NULL) { }
14     };
15
16 };
17
18
19
20
21
22
```

List.hpp

```
1 #include "List.h"
2
3 template <class T>
4 void List<T>::insertAtFront(const T& t) {
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22 }
```



Running Time of Linked List `insertAtFront`

List.cpp

```
24 template <typename T>
25 T List<T>::operator[](unsigned index) {
26
27
28
29
30
31 }
```

List.cpp

```
33 ListNode *& List<T>::_index(int index) const {  
34  
35  
36  
37  
38  
39  
40 }
```