Easy to Walk Over the List of Players

Let's see how it's used.

How do we do something for all players?

```
player_t* p;
for (p = player_list.next;
     &player_list != p;
     p = p->next) {
   // do something for all players
}
```
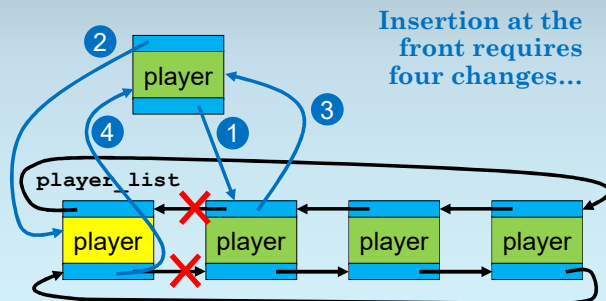And how do we advance to the next player?

The Opposite Direction is Equally Easy

But…what if we want the other direction?

Just change next to prev in both cases!

```
player_t* p;
for (p = player_list.prev;
     &player_list != p;
     p = p->prev) {
   // do something for all players
}
```

Insertion Requires Four Changes In Correct Order

Insertion at the front requires four changes…

Insertion at Either End of the List is Easy

Given a new player_t* p, we have…

```
p->next = player_list.next;
p->prev = &player_list;
player_list.next->prev = p;
player_list.next = p;
```
Or, at the end,
```
p->prev = player_list.prev;
p->next = &player_list;
player_list.prev->next = p;
player_list.prev = p;
```

2