



E 463 Operating Systems

Electrical and Computer Engineering Department
Engineering College
King Abdulaziz University

La#4
Winter-2023

Lab 4

#	Names:	ID	Section
1	Ahmed Alkhayal	1945541	C3

Instructor:

Dr. Abdulghani M. Al-Qasimi

Date: 6/6/2023

Exercise 1:

Changed code:

Before	After
head_ = new Node(marker->value(), marker->next());	head_ = marker->next();
temp->next (marker->next()); delete temp; temp = 0; return 0;	temp->next (marker->next()); delete marker; marker = 0; temp = 0; return 0;

Output:

```
Creating Node, 1 are in existence right now
Creating Node, 2 are in existence right now
Creating Node, 3 are in existence right now
Creating Node, 4 are in existence right now
The fully created list is:
4
3
2
1

Now removing elements:
Destroying Node, 3 are in existence right now
3
2
1

Destroying Node, 2 are in existence right now
3
2

Destroying Node, 1 are in existence right now
3

Destroying Node, 0 are in existence right now
```

Reasoning:

When the first element was removed, a memory leak occurred because a new node was created and assigned as the head. To fix this issue, we can update the head_ pointer to point to the next marker, effectively removing the first element from the list. Additionally, changing delete temp; to delete marker; ensures proper deletion of the current marker node. Setting marker to 0 afterwards prevents any potential misuse of the deleted memory

Exercise 2:

By resolving the bug in exercise 1, this issue was also fixed. The next output demonstrates that the code is now bug-free. I followed the mentioned sequence of inserting 1, 2, 3, 4 and then removing 2

Output:

```
Creating Node, 1 are in existence right now
Creating Node, 2 are in existence right now
Creating Node, 3 are in existence right now
Creating Node, 4 are in existence right now
The fully created list is:
4
3
2
1

Now removing elements:
Destroying Node, 3 are in existence right now
4
3
1
```