

Class Discussion

Write a function that returns the second element in the argument list.

SecondElement([4,9,3])=9

SecondElement([7,1])=1

SecondElement([])=**ERROR**

SecondElement([6])=**ERROR**

```
listElementT SecondElement(listADT list)
{

    return(Head(Tail(list)));
}
```

```
listElementT SecondElement(listADT list)
{
    if (ListIsEmpty(list)) exit(EXIT_FAILURE);
    if (ListIsEmpty(Tail(list))) exit(EXIT_FAILURE);
    return(Head(Tail(list)));
}
```

Class Discussion

Write a function that changes the first element in the argument list to 0.

ZeroFirst([5,8,7,6])=[0,8,7,6]

ZeroFirst([4,9])=[0,9]

ZeroFirst([1])=[0]

ZeroFirst([])=**ERROR**

```
listADT ZeroFirst(listADT list)
{
    if (ListIsEmpty(list)) exit(EXIT_FAILURE);
    return (Cons(0,Tail(list)));
}
```

Does this work? And WHY?

```
listADT ZeroFirst(listADT list)
{
    listElementT head;
    if (ListIsEmpty(list)) exit(EXIT_FAILURE);

    head = Head(list);
    head = 0;
    return (list);
}
```

Class Discussion

What will be the values of list1 and list2?

```
main()
{
    list1 = Cons(1, Cons(8, EmptyList()));
    list2 = Cons(9, list1);
    /* list1=?, list2=? */
    list1 = Cons(6, Cons(7, EmptyList()));
    /* list1=?, list2=? */
    list2 = Cons(2, EmptyList());
    /* list1=?, list2=? */
}
```

Class Discussion

Write a function that exchanges the second and third elements in the argument list.

Exchange23([3,7,2,5,6])=[3,2,7,5,6]

Exchange23([3,4,5])=[3,5,4]

Exchange23([1,2])=**ERROR**

Exchange23([9])=**ERROR**

Exchange23([])=**ERROR**

Class Discussion

Write a function that returns the reversed argument list.

`Reverse([]) = []`

`Reverse([1,2]) = [2,1]`

`Reverse([6,8,3,2,4,5,7,1]) = [1,7,5,4,2,3,8,6]`

```
listADT Reverse(listADT list)
{
    if (ListIsEmpty(list)) return(EmptyList());
    return(Append(Reverse(Tail(list)), Head(list)));
}
```

```
listADT Reverse(listADT list)
{
    listADT ReversedTail;
    if (ListIsEmpty(list)) return(EmptyList());
    ReversedTail = Reverse(Tail(list))
    return(Append(ReversedTail, Head(list)));
}
```