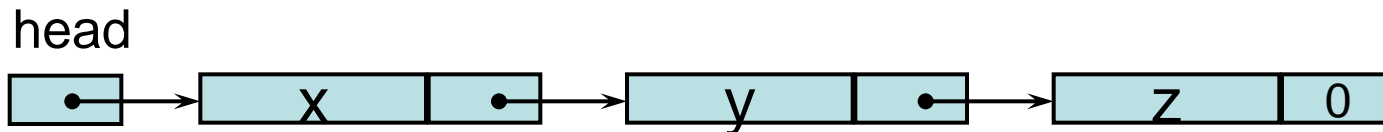


## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

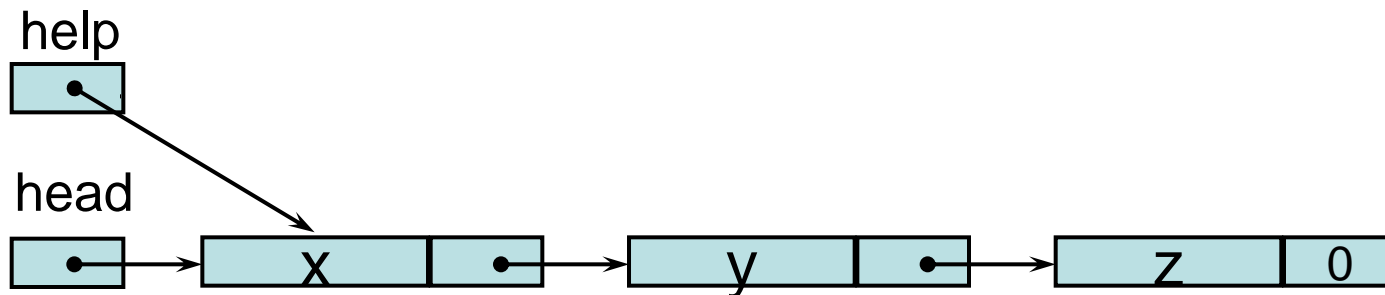
a: z

Vor dem Test



## Inklusionstest

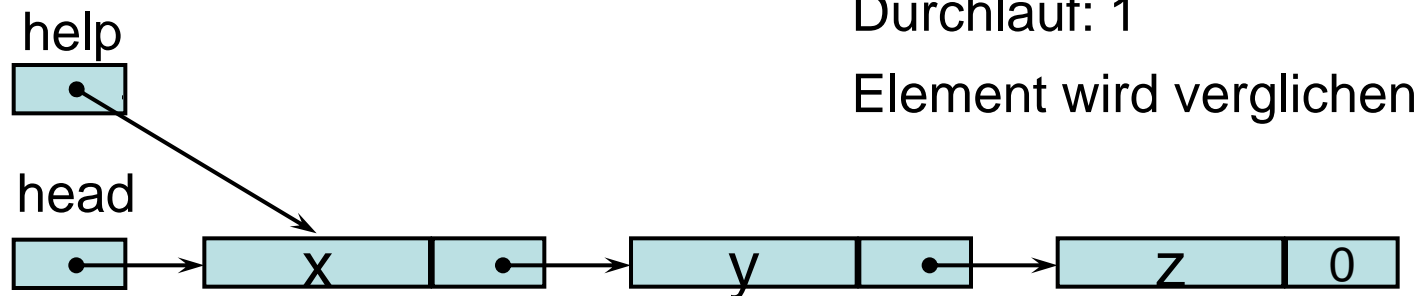
```
int List::Member(ItemType a) {  
    Element* help = head;                                a: Z  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```



## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

a: Z



## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

a: z

help



head



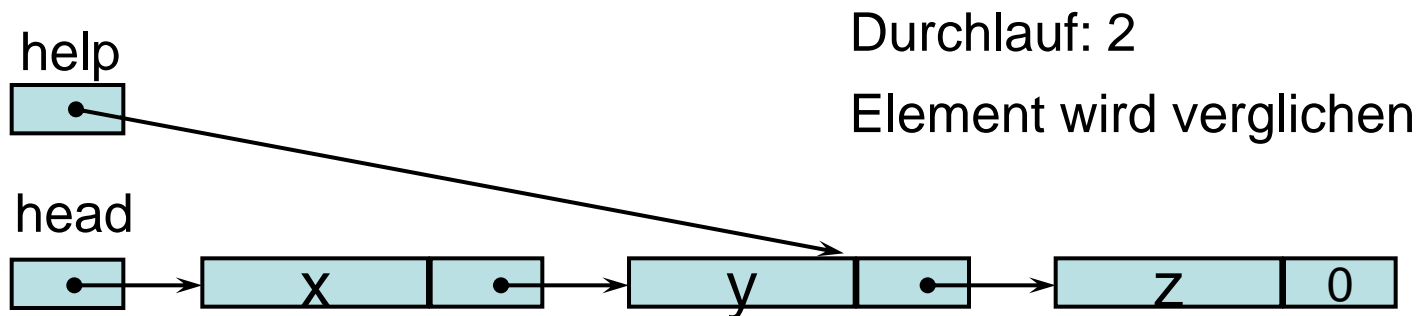
Durchlauf: 1

Zeiger geht weiter

## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

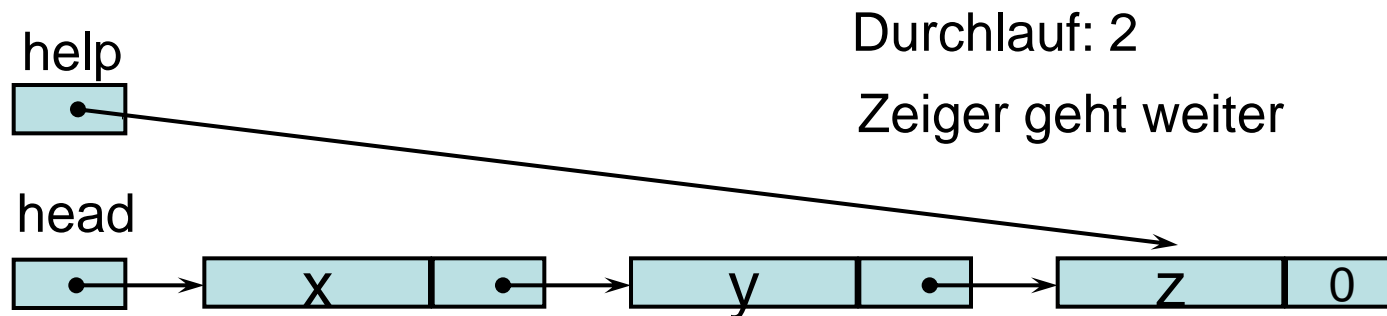
a: Z



## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

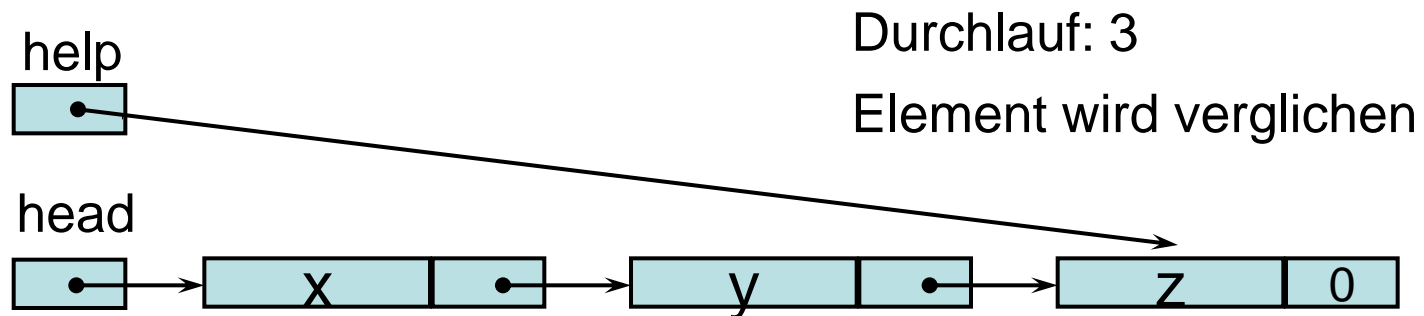
a: z



## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

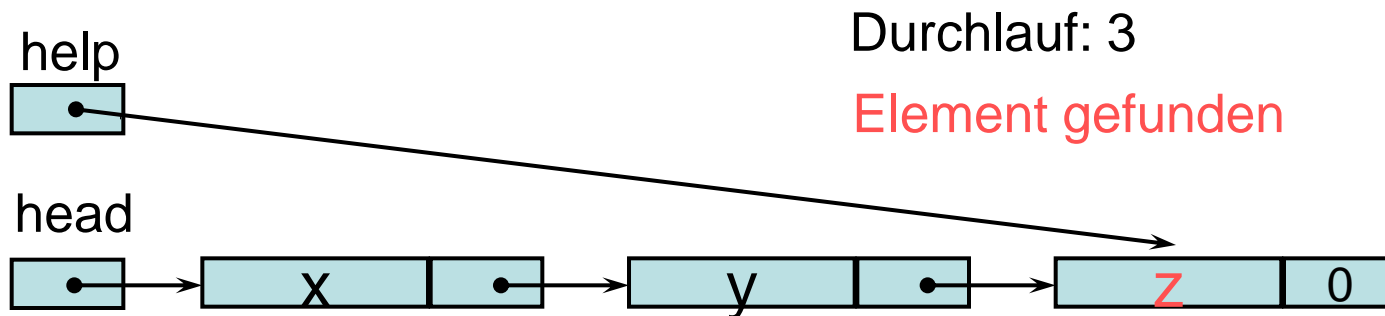
a: Z



## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

: z





## Inklusionstest

```
int List::Member(ItemType a) {  
    Element* help = head;  
    while(help != 0 && help->value != a)  
        help = help->next;  
    if(help != 0) return 1;  
    else return 0;  
}
```

a : z

Nach dem Test

