

```
typedef char          *Element;
typedef struct ListNode *List;
struct ListNode { Element elem; List next; };
```

▪ Zugriff auf ein Listenelement

```
Element getAt(List l, unsigned int idx) {
    assert( !isEmpty(l) || (idx == 0) );
    return idx == 0
        ? head(l)
        : getAt( tail(l), idx-1 );
}
```

▪ Einfügen in eine Liste

```
List insertAt(List l, unsigned int idx, Element e) {
    assert( !isEmpty(l) || (idx == 0) );
    return idx == 0
        ? cons(l, e)
        : ( l->next = insertAt( tail(l), idx-1, e ), l );
}
```