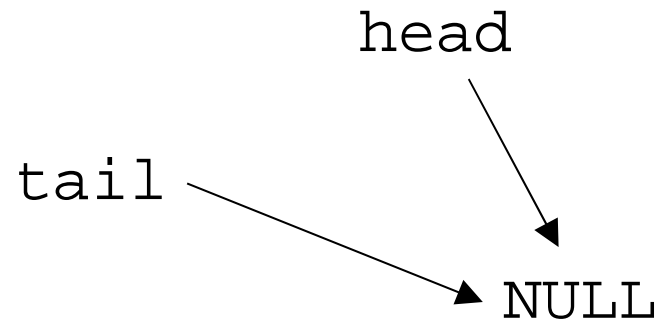


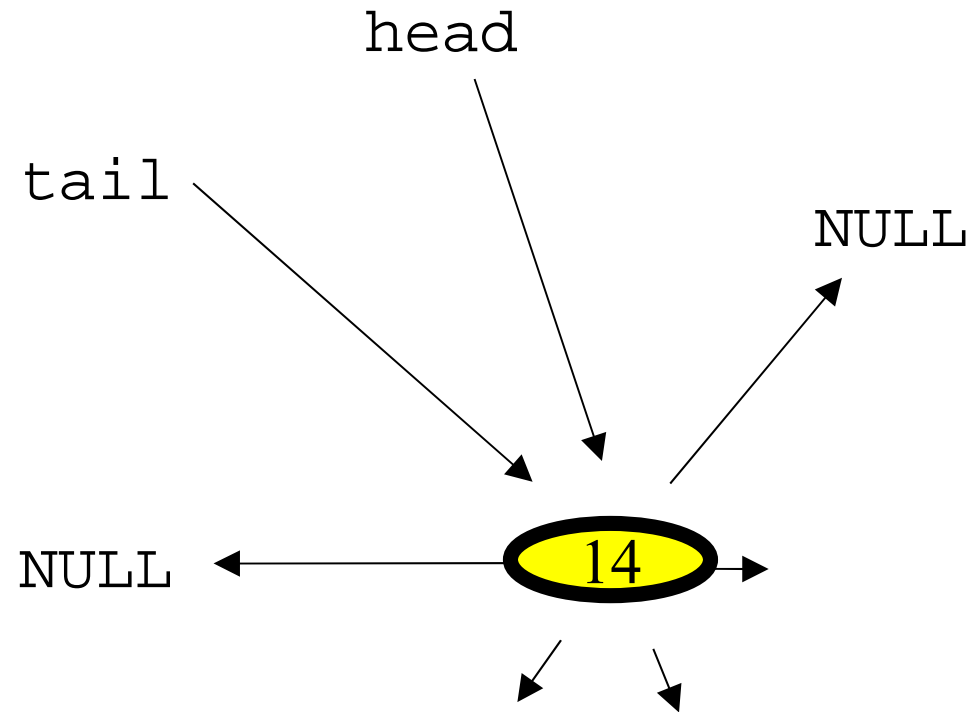
```

Class Cell {
    Cell *lefttree, *righttree, *leftnext, *rightnext, *parent;
    TreeObject *object;

    Cell (TreeObject *obj, Cell *lnx, Cell *prt) {
        parent = prt;
        leftnext = lnx;
        object = obj;
        rightnext = righttree = lefttree = NULL;
    }
};

```

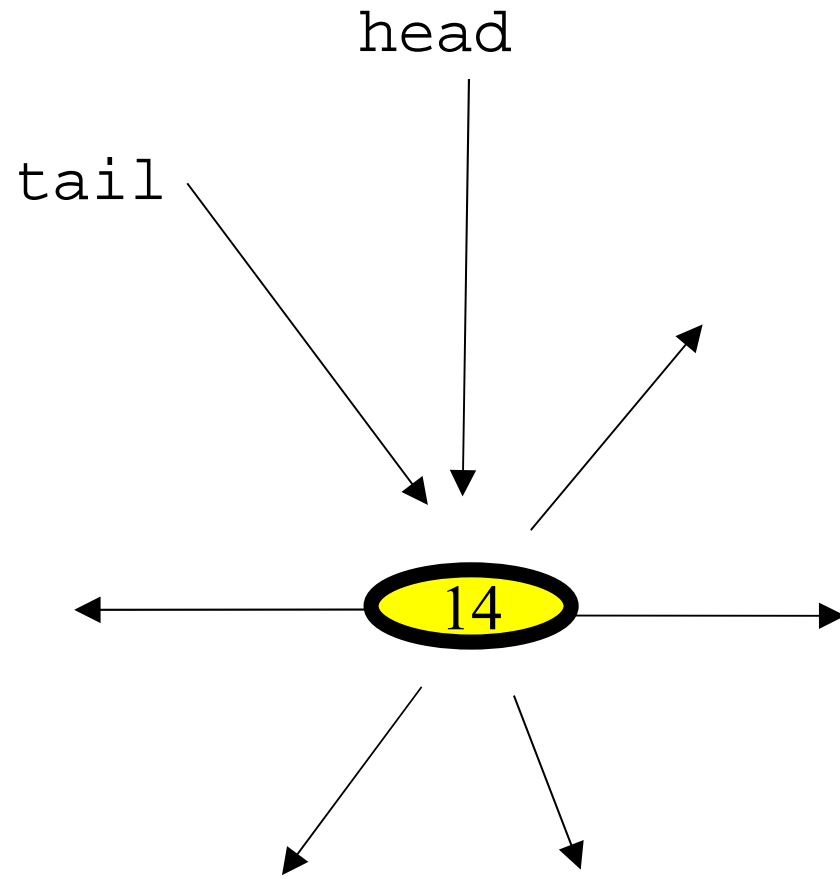


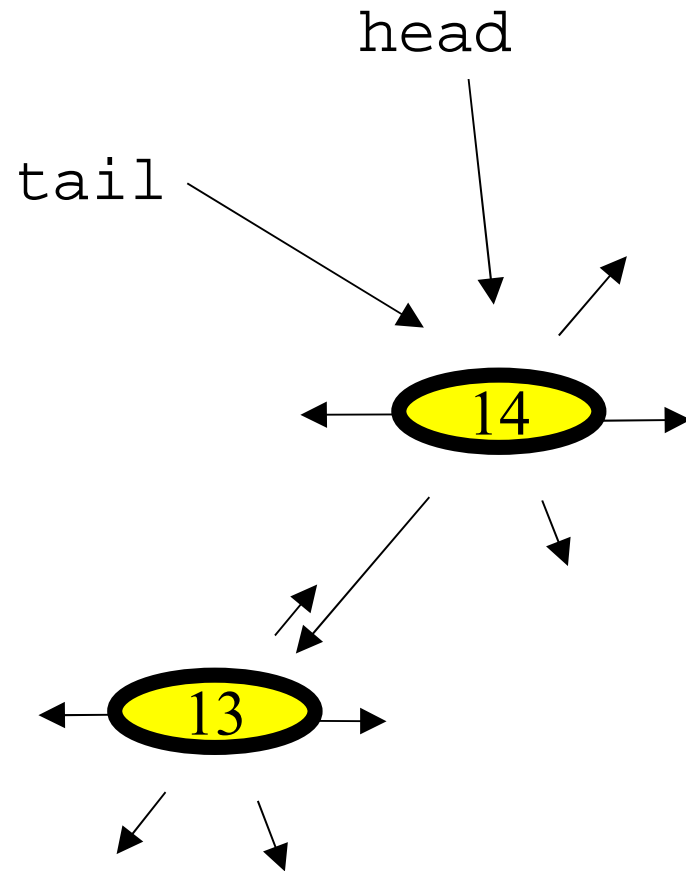


```
TreeObject *t = new IntegerObject(14);
```

```
...
```

```
if (head == NULL) {  
    head = tail = new Cell(t, NULL, NULL);  
}
```

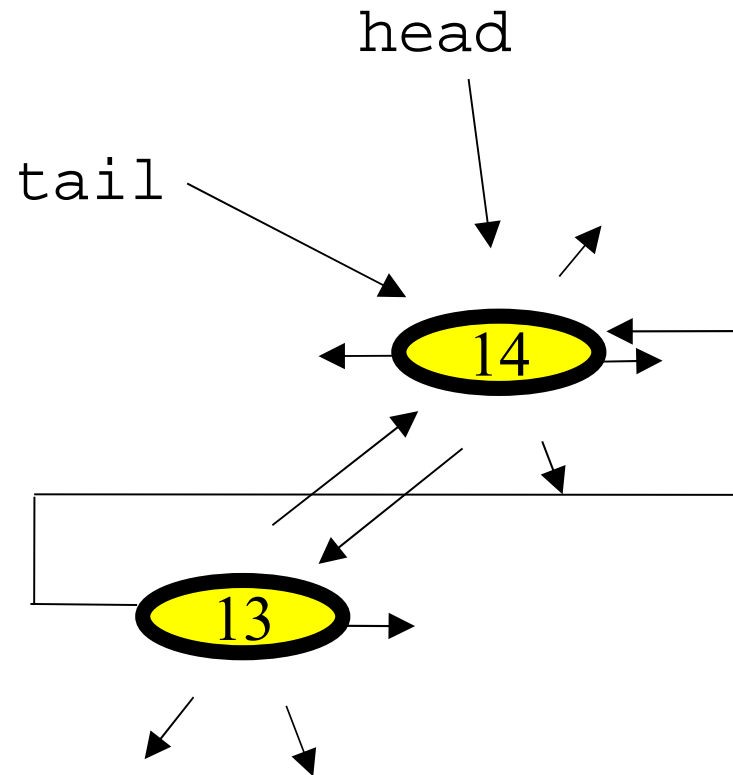




```
TreeObject *t = new IntegerObject(13);
```

...

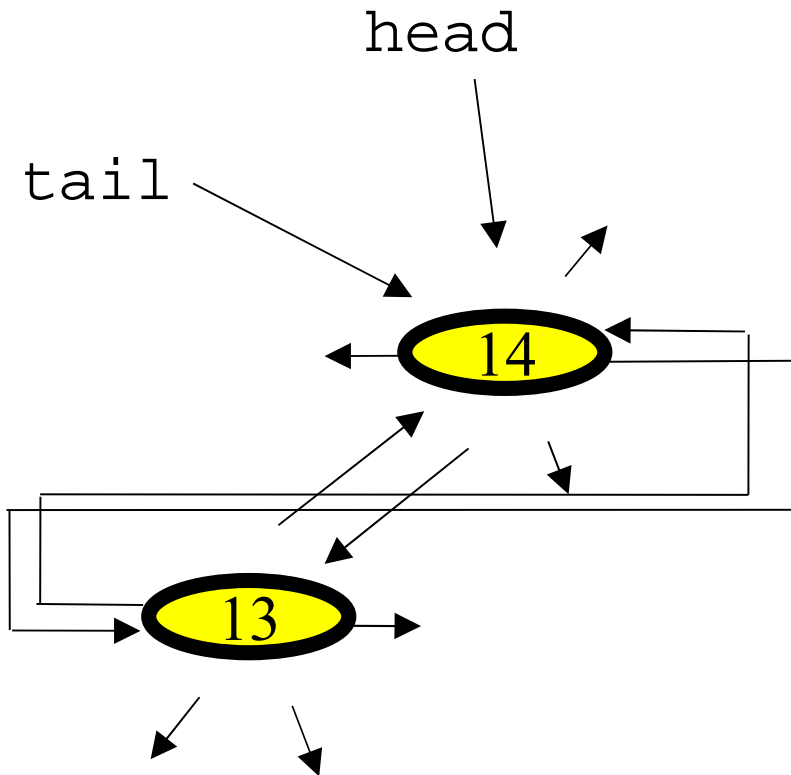
```
if (head == NULL) {
    head = tail = new Cell(t, NULL, NULL);
} else if (tail->lefttree == NULL) {
    if (tail->leftnext == NULL) {
        tail->lefttree = new Cell(...
```



```
TreeObject *t = new IntegerObject(13);
```

```
...
```

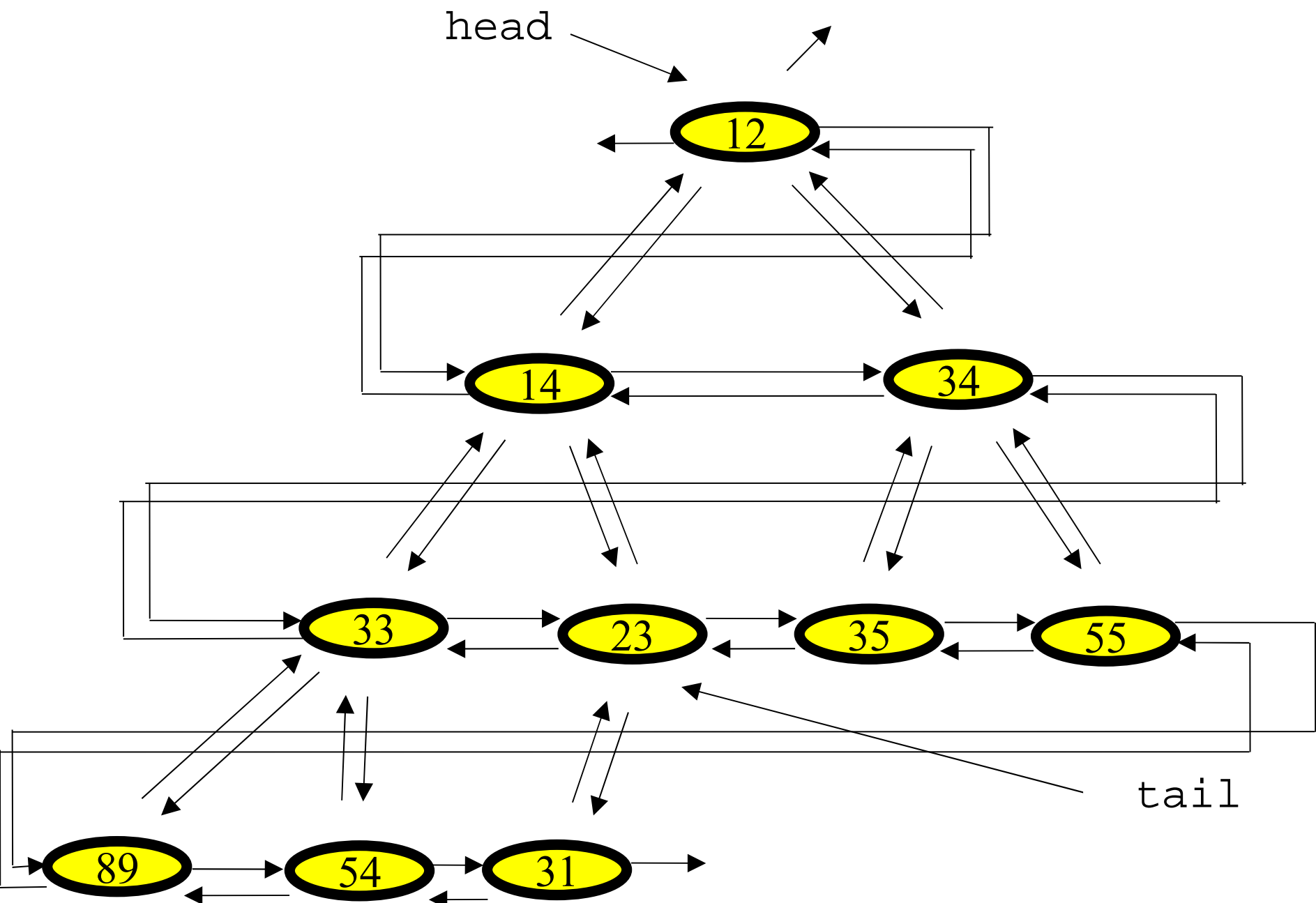
```
if (head == NULL) {
    head = tail = new Cell(t, NULL, NULL);
} else if (tail->lefttree == NULL) {
    if (tail->leftnext == NULL) {
        tail->lefttree = new Cell(t, tail, tail);
        ...
    }
}
```



```
TreeObject *t = new IntegerObject(13);
```

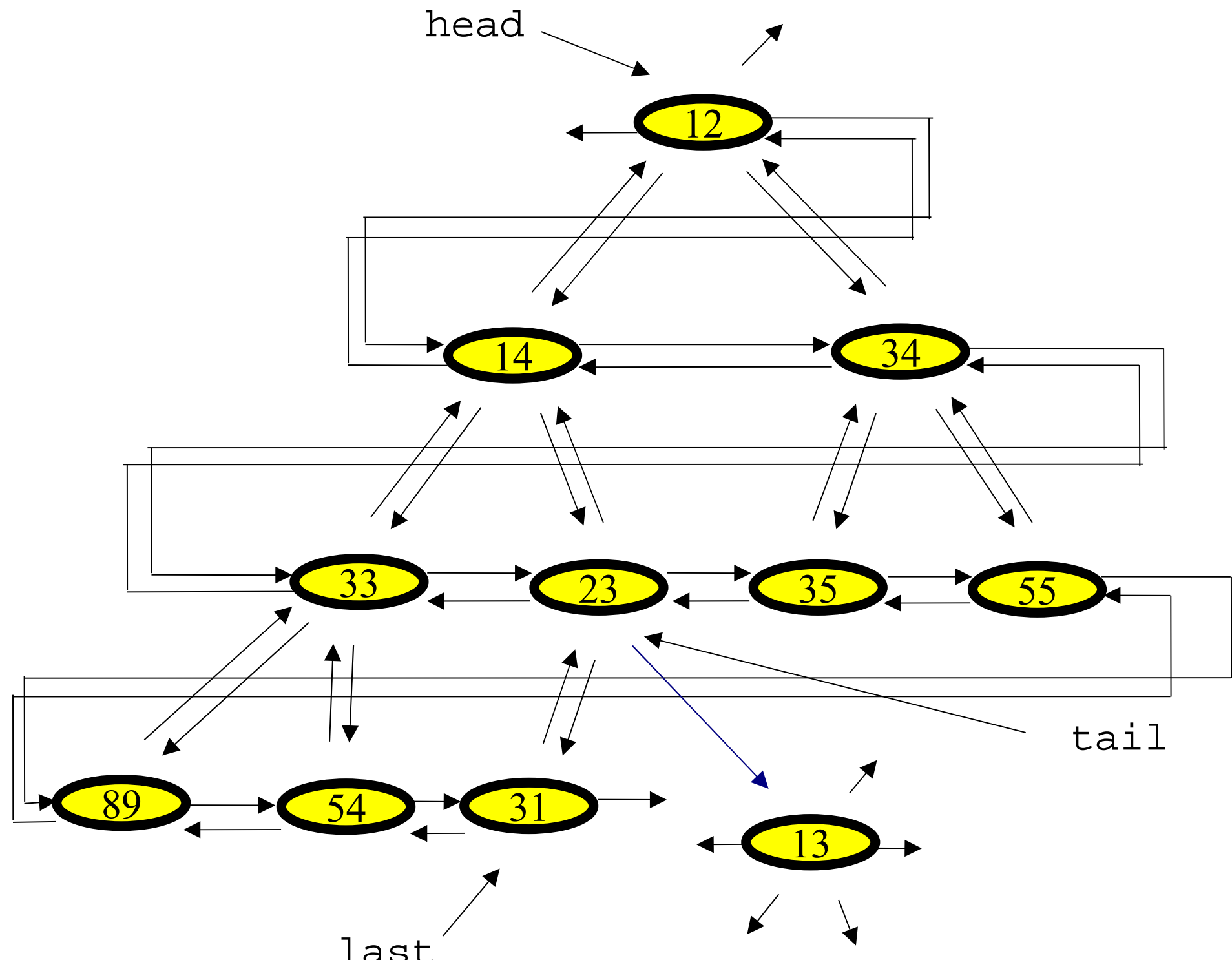
```
...
```

```
if (head == NULL) {
    head = tail = new Cell(t, NULL, NULL);
} else if (tail->lefttree == NULL) {
    if (tail->leftnext == NULL) {
        tail->lefttree = new Cell(t, tail, tail);
        tail->rightnext = tail->lefttree;
    }
}
```

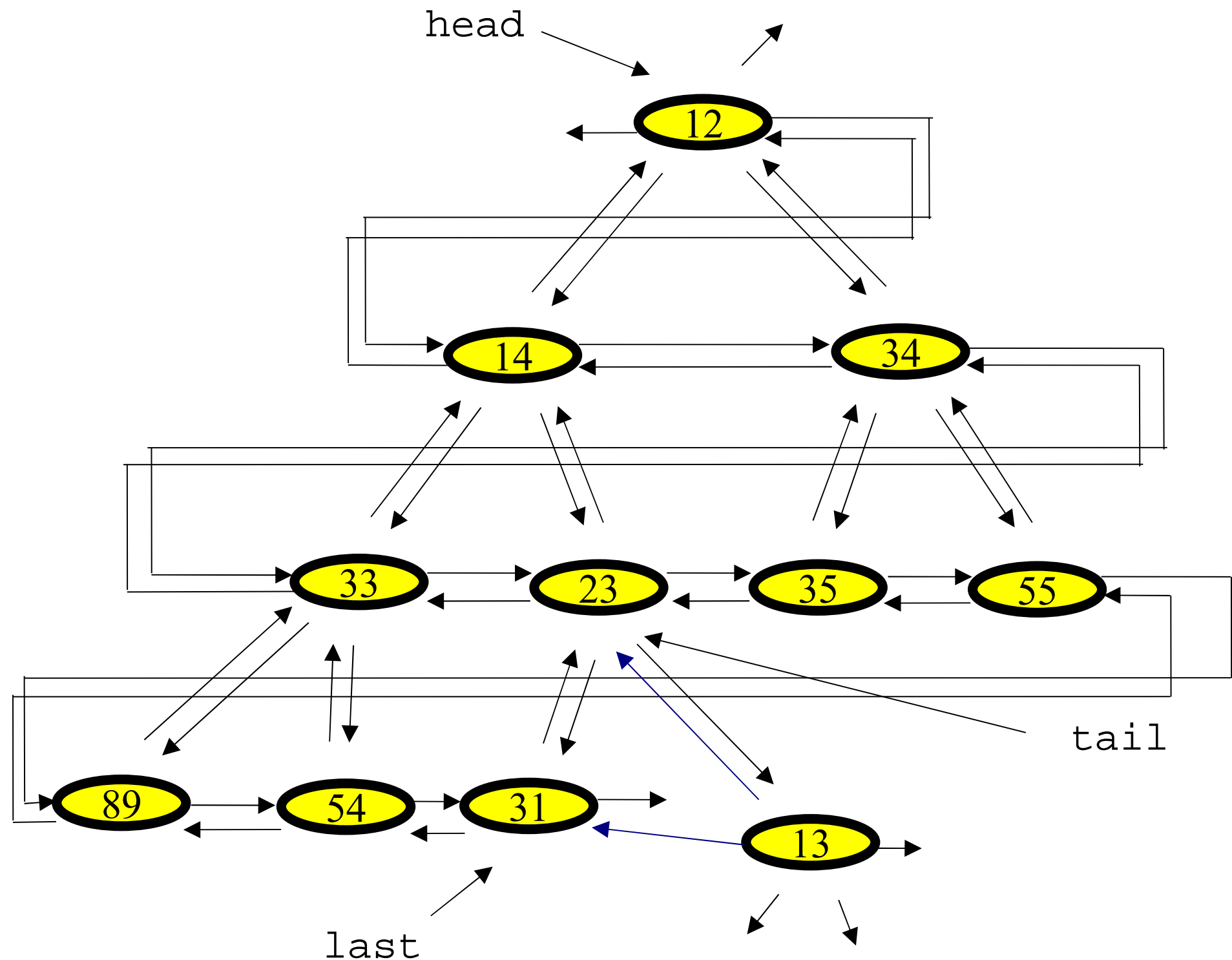


```
.last
Cell *last = tail->lefttree;
```

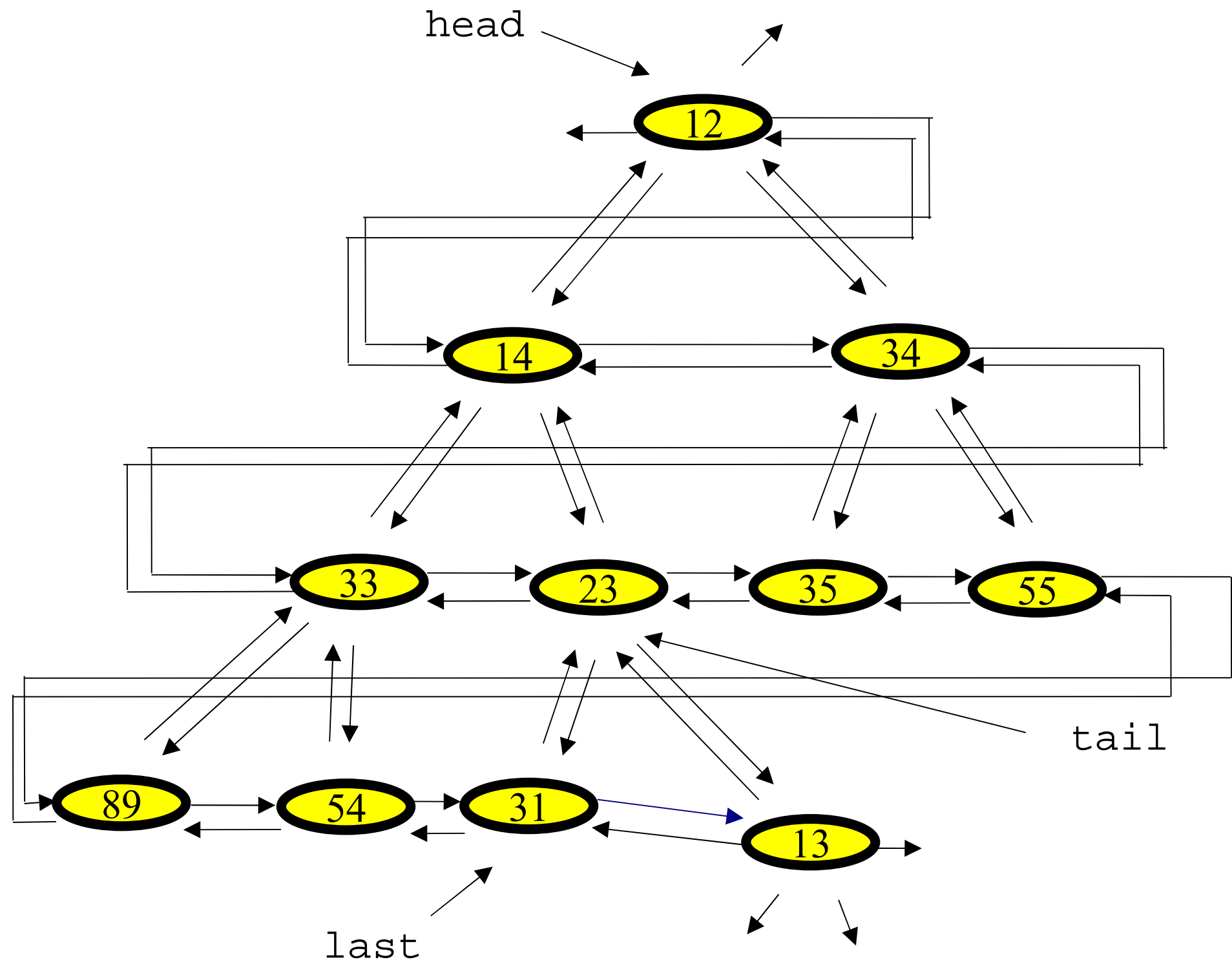




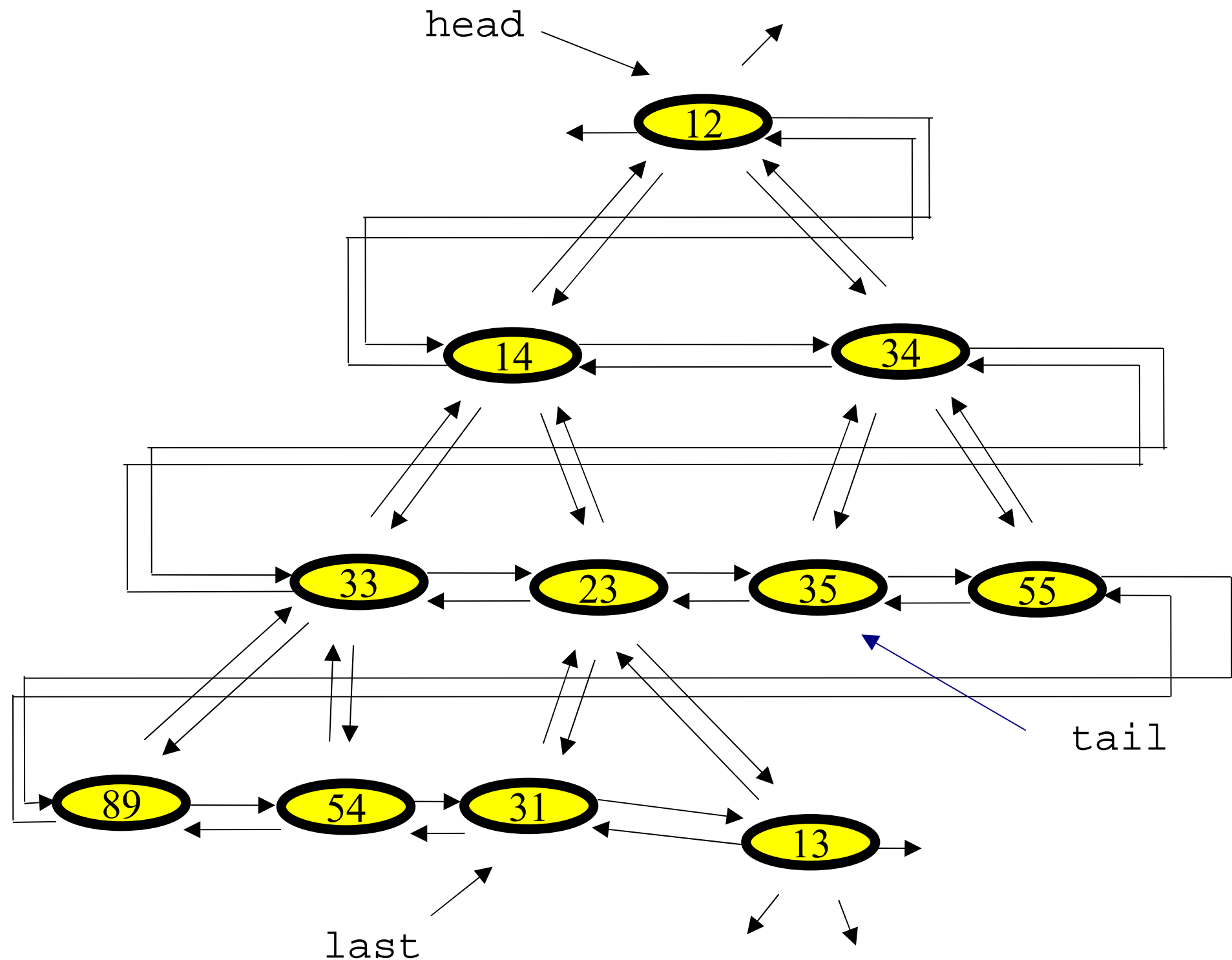
tail->righttree = new Cell(t, ...



```
tail->righttree = new Cell(t, last, tail);
```

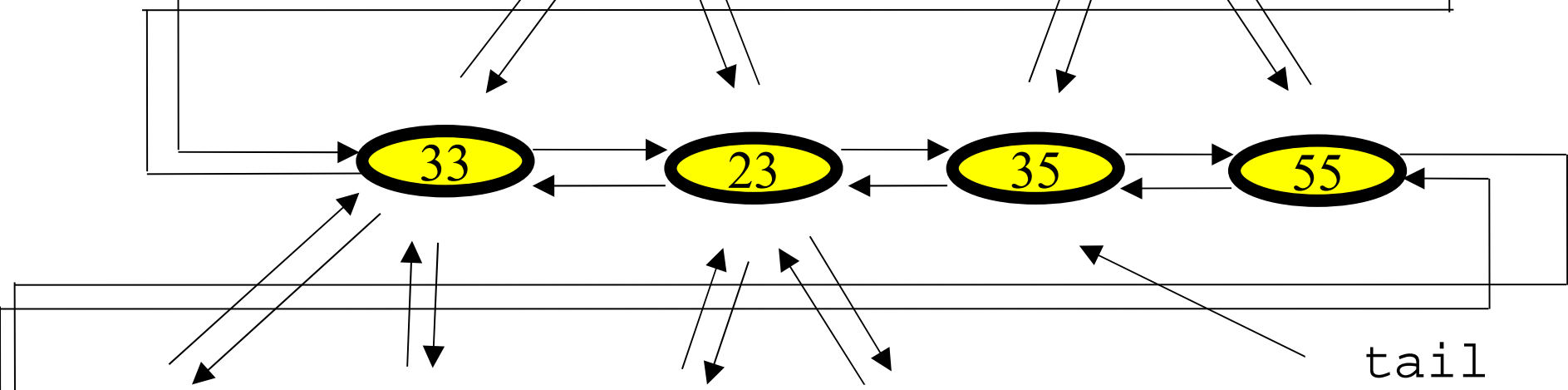
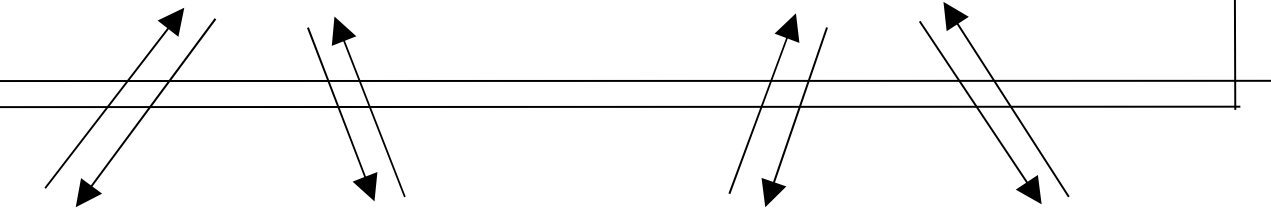
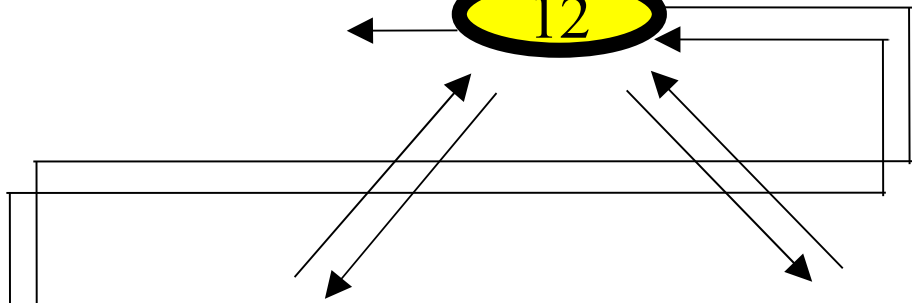
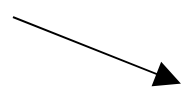


last->rightnext = tail->righttree;

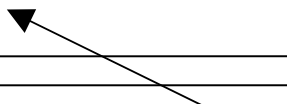


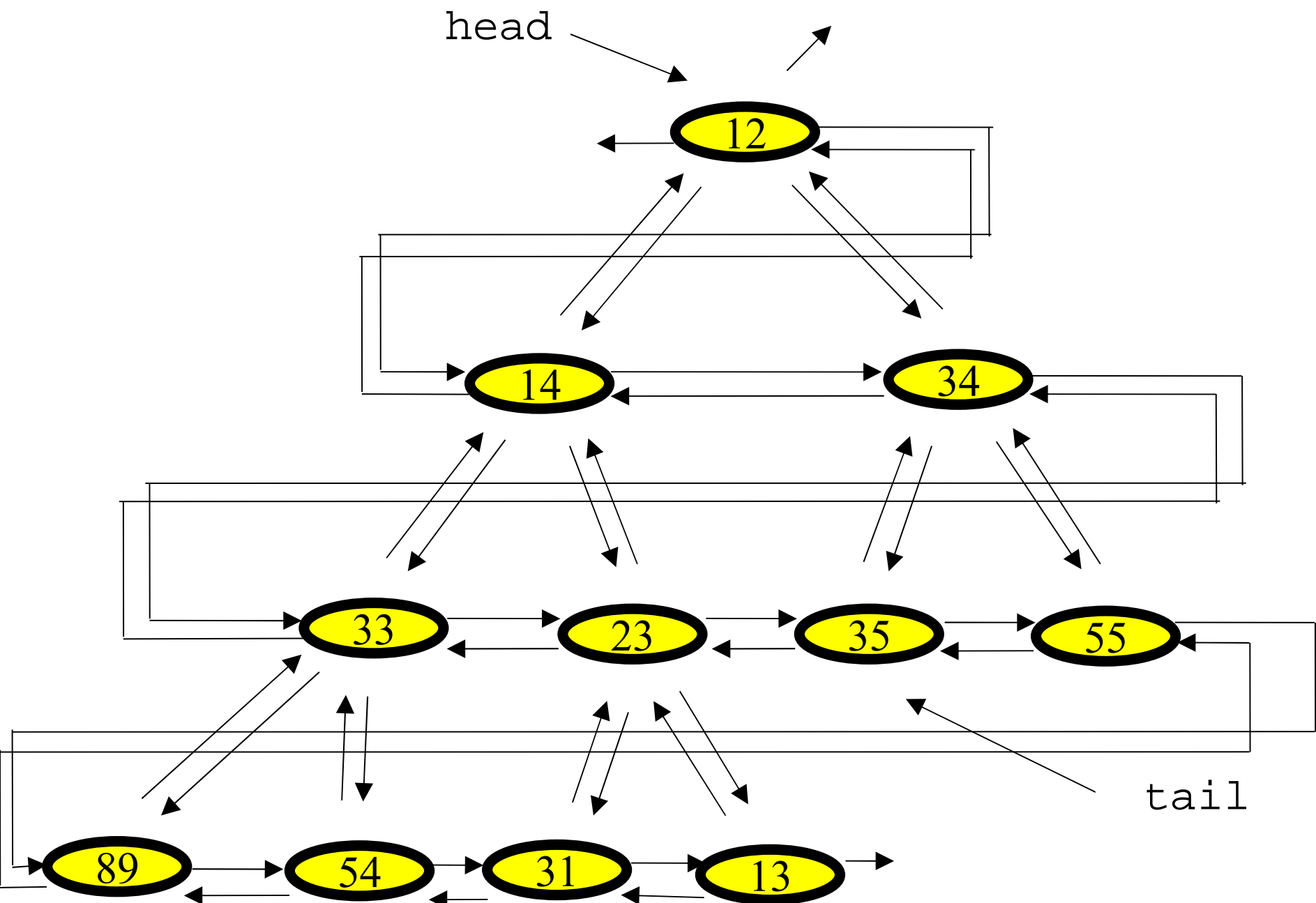
```
tail = tail->rightnext;
```

head

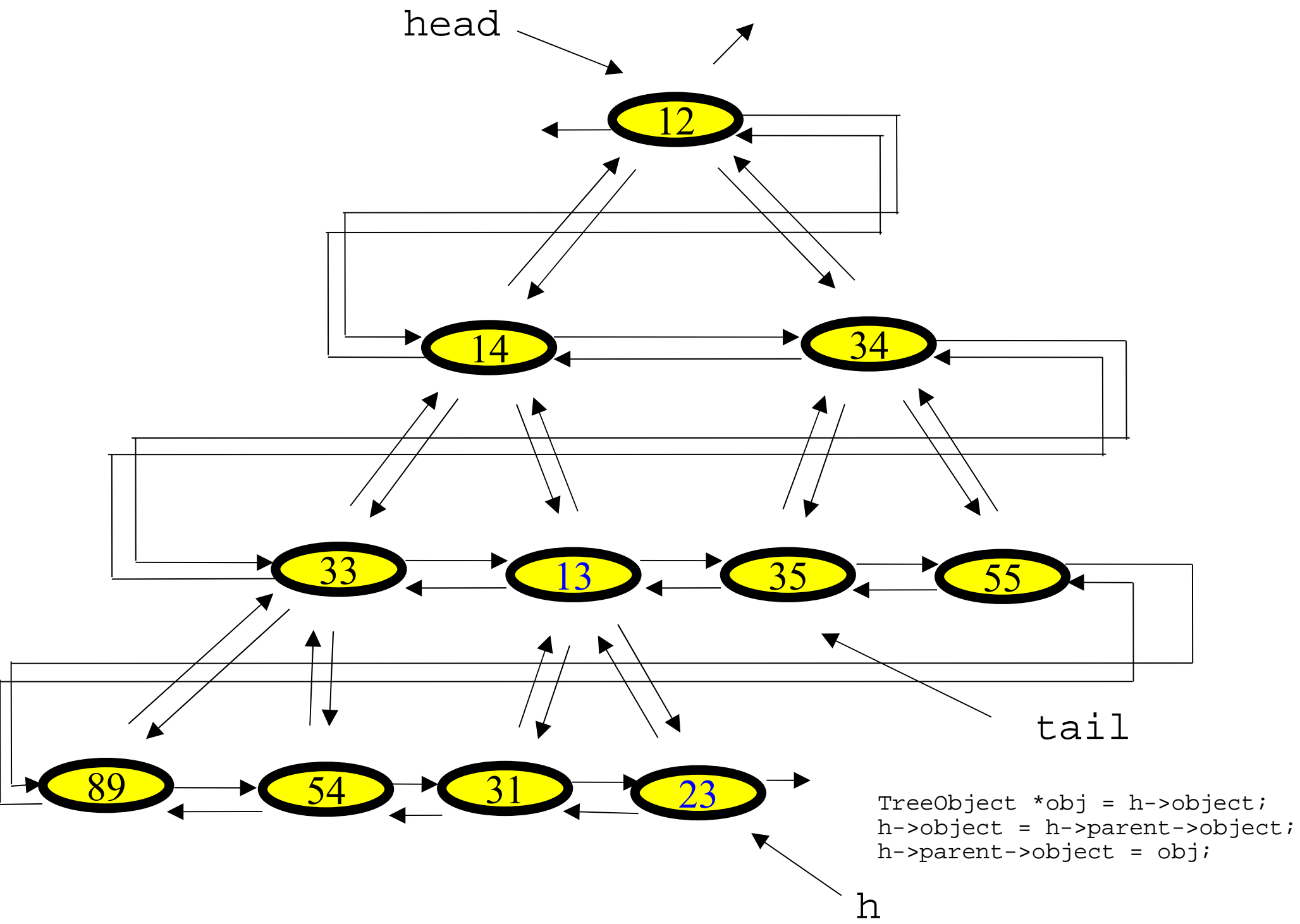


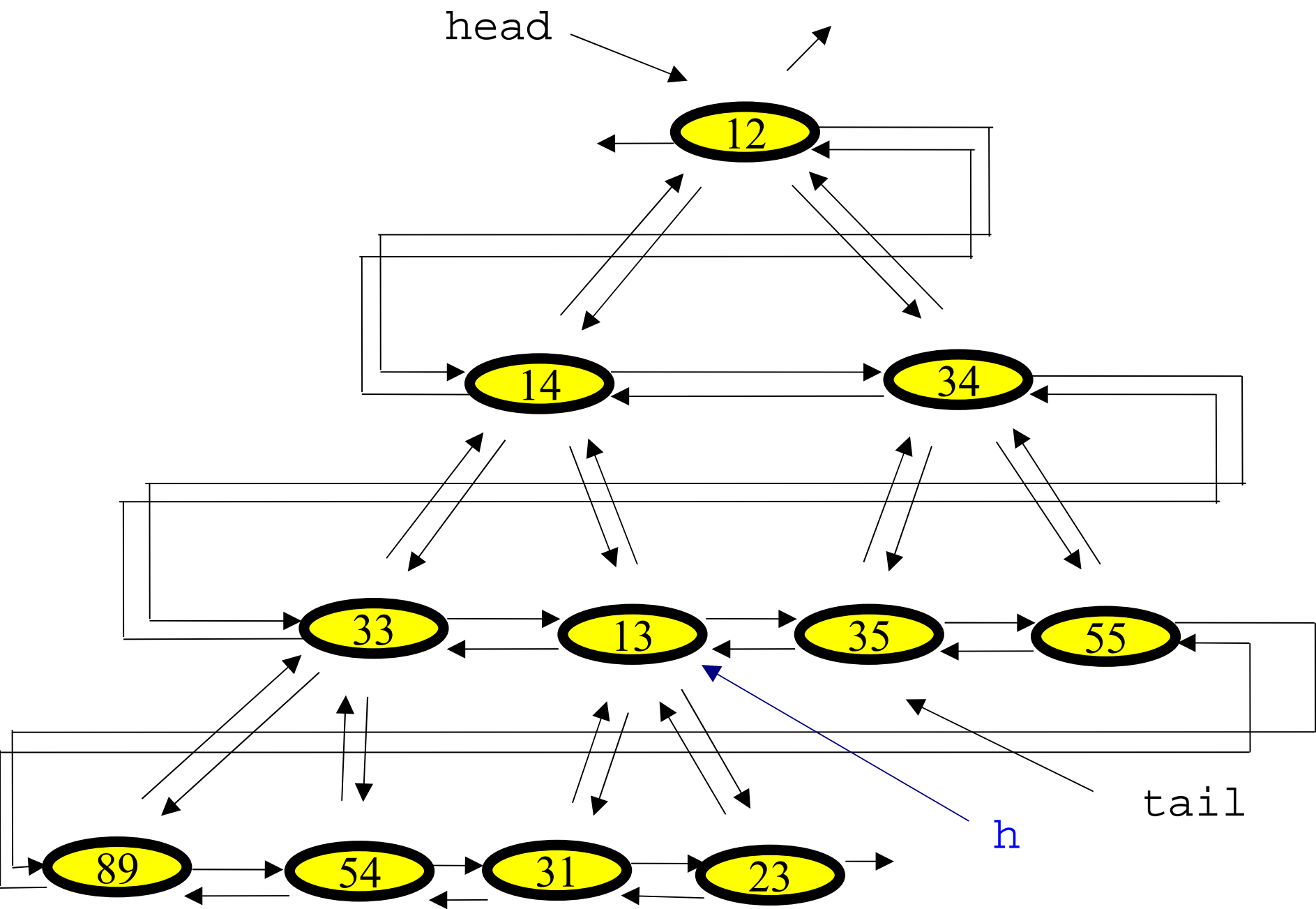
tail





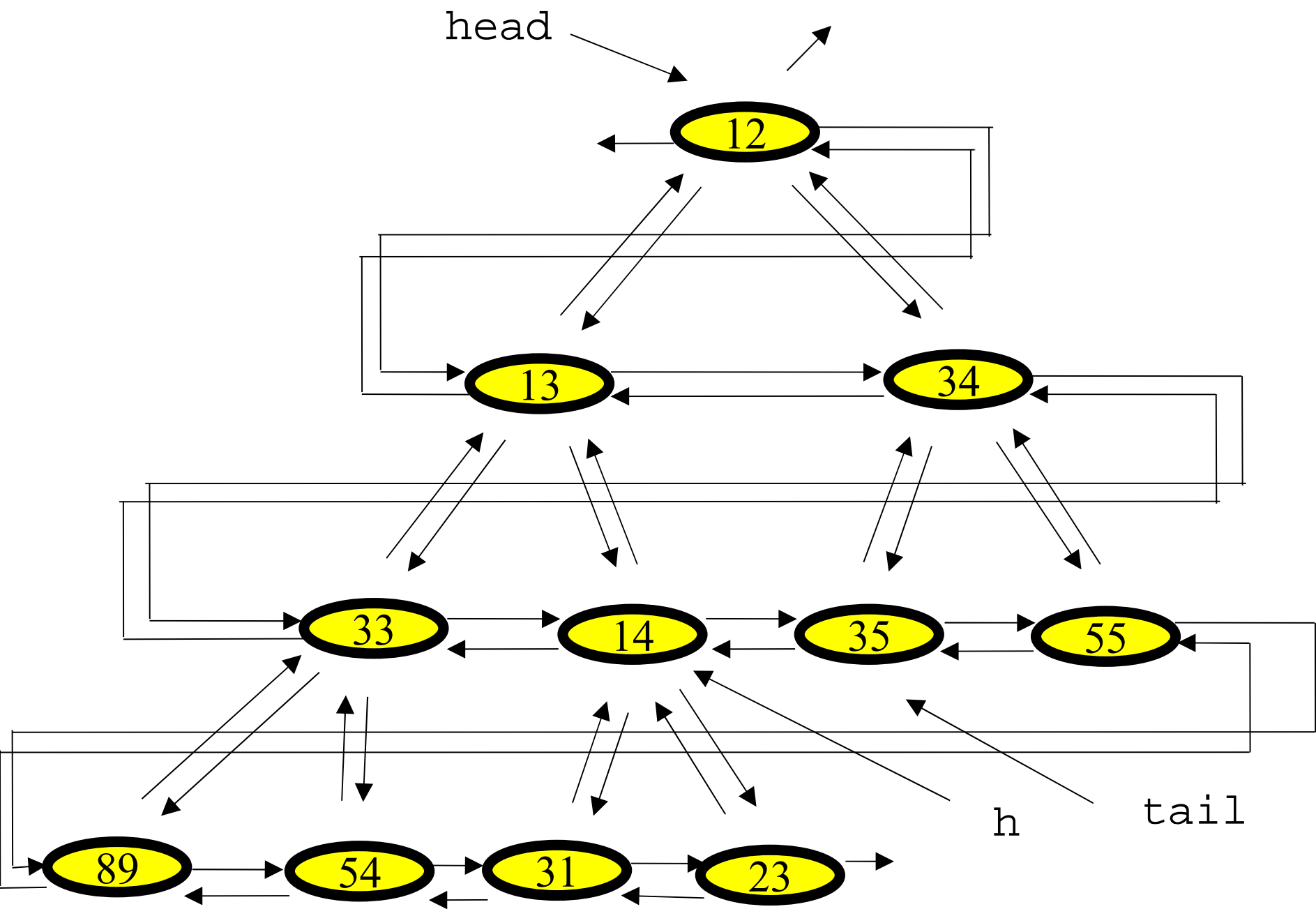
```
h = tail->leftnext->righttree;
```

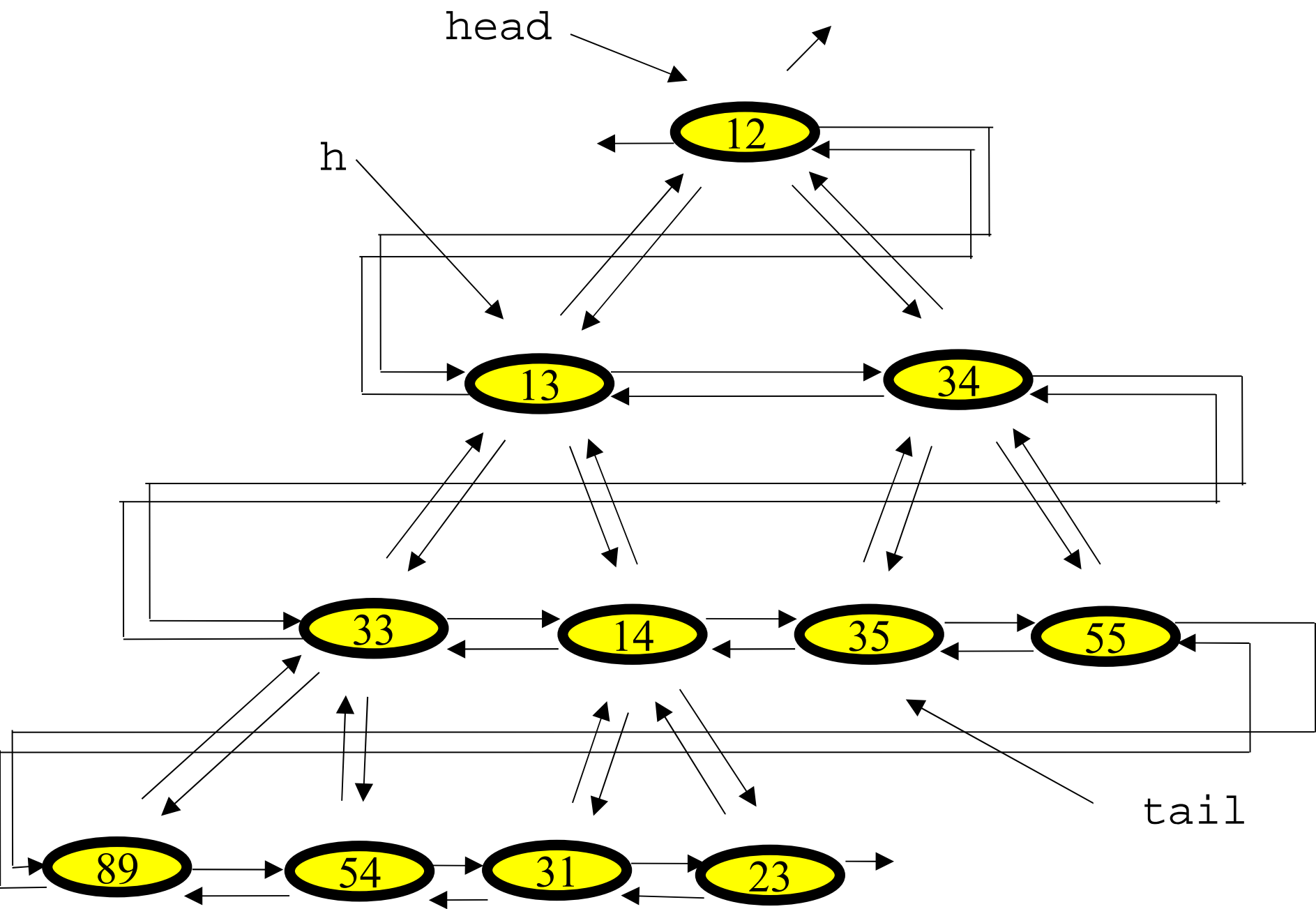


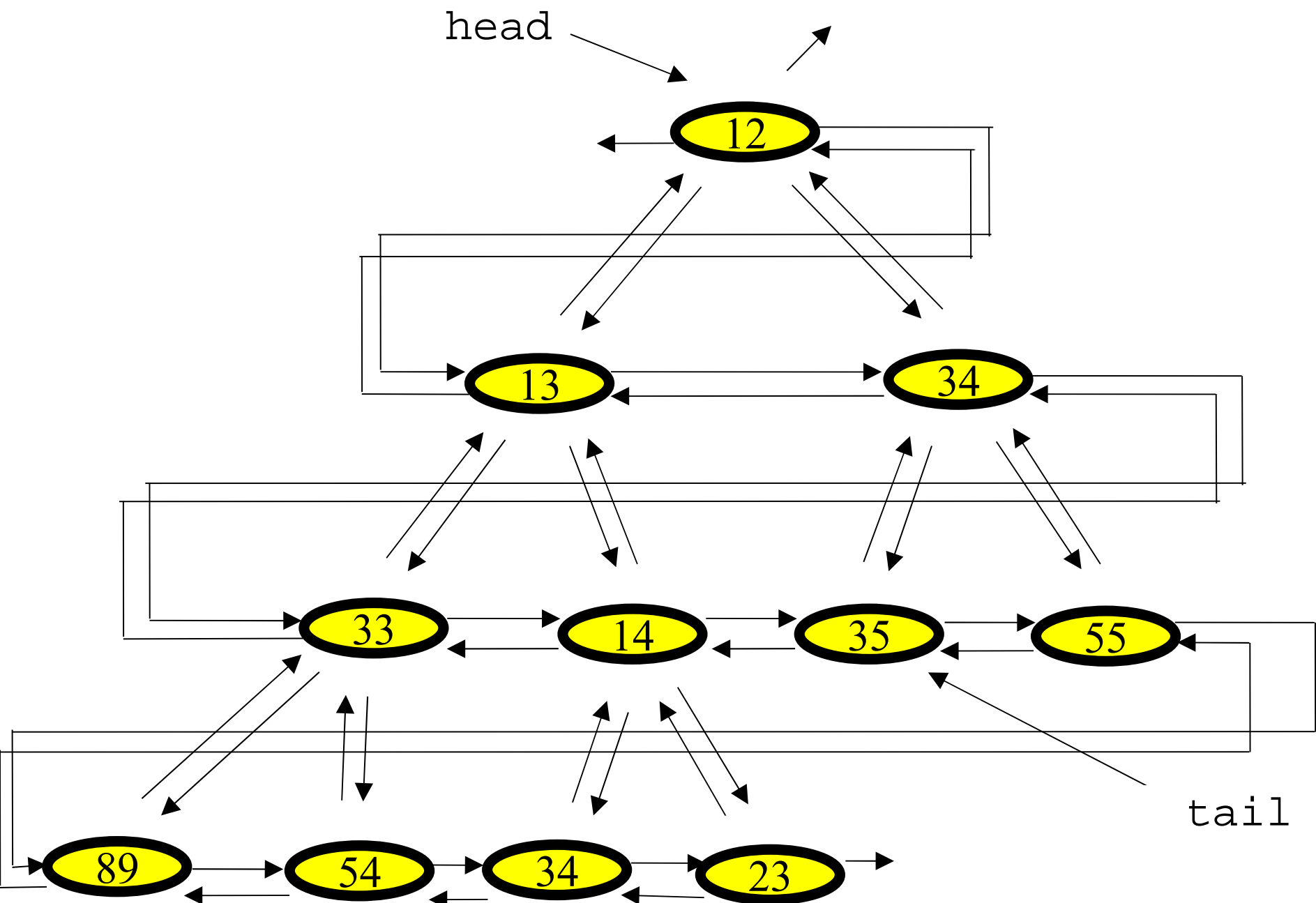


```
H = h->parent;
```



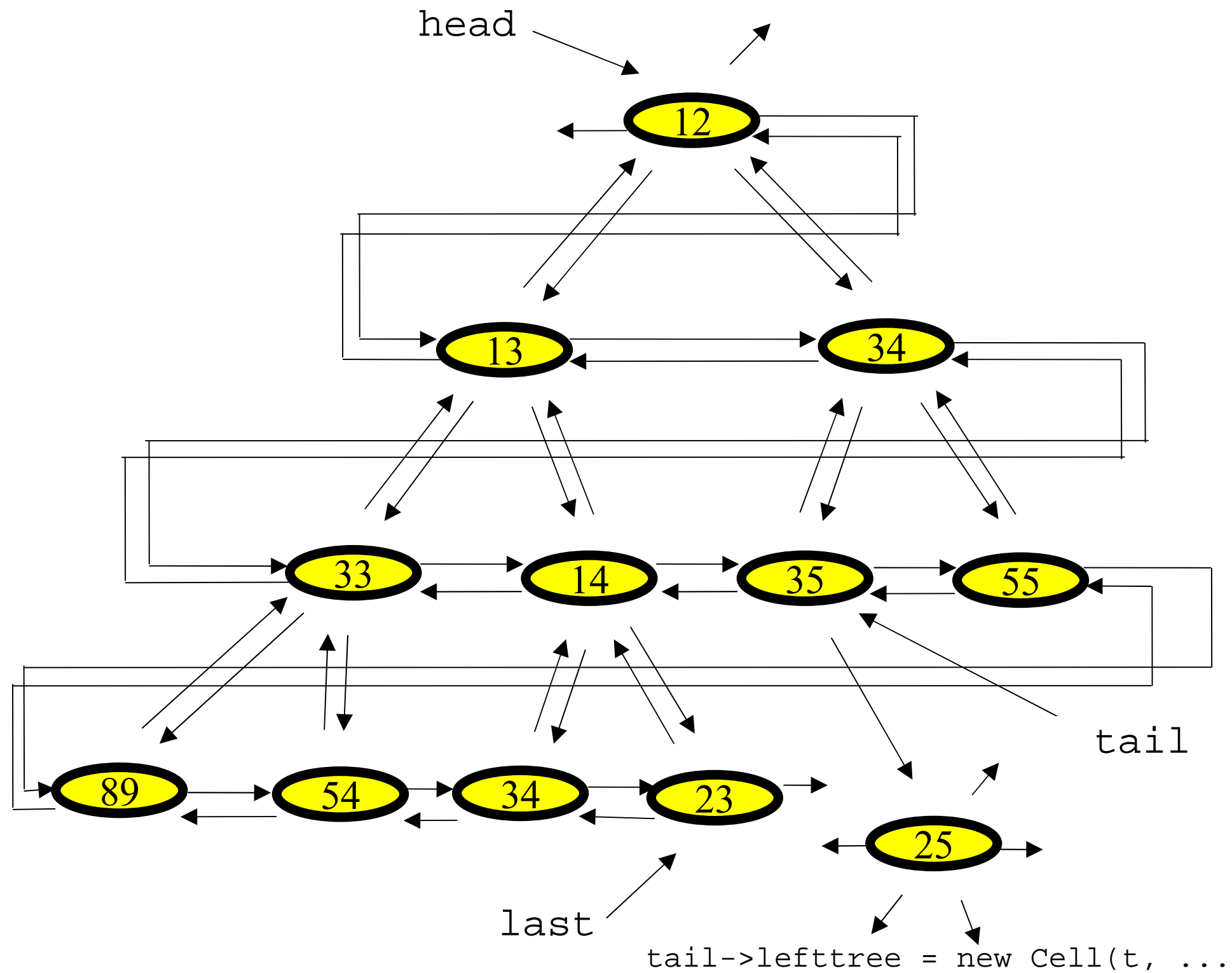


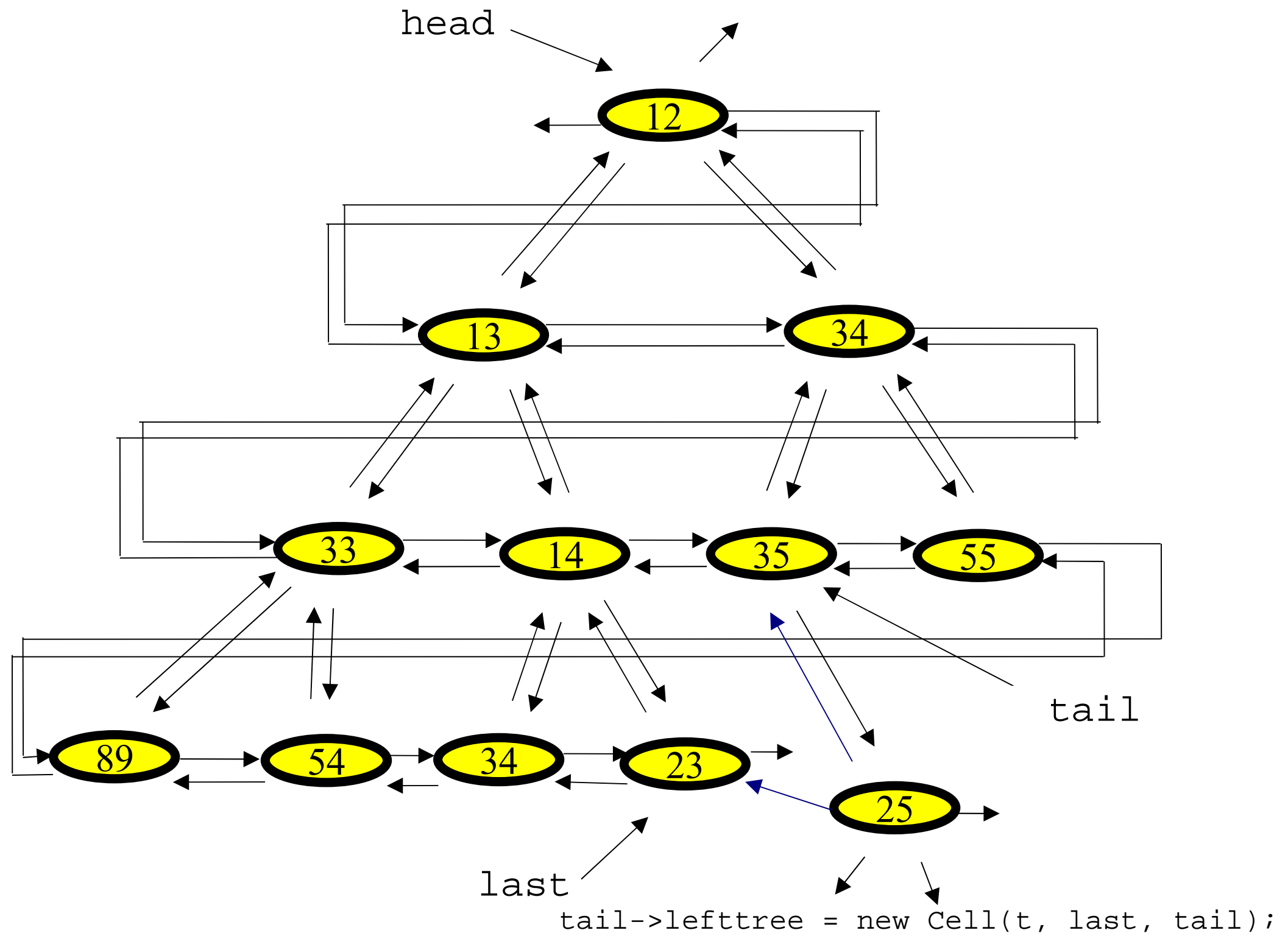


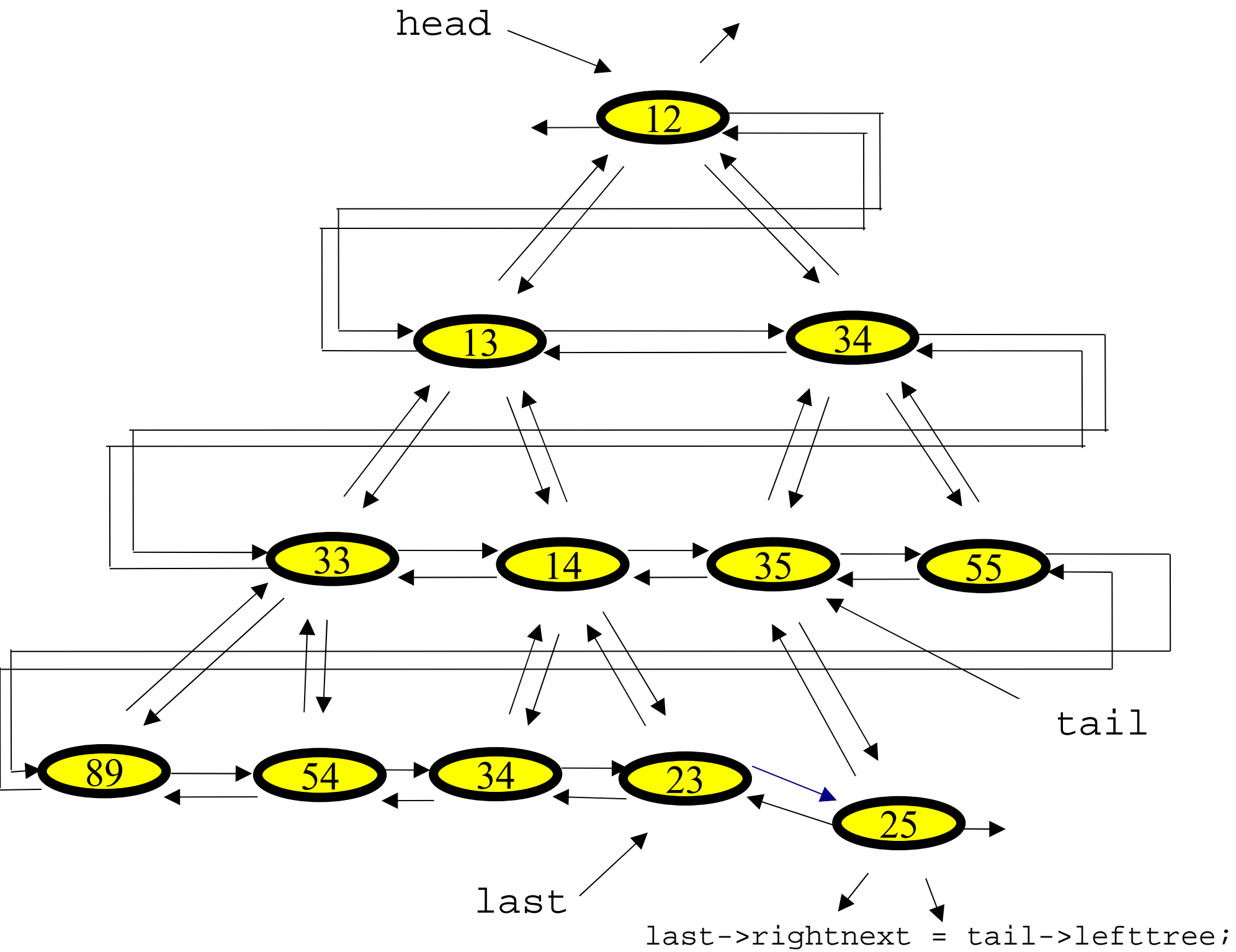


last

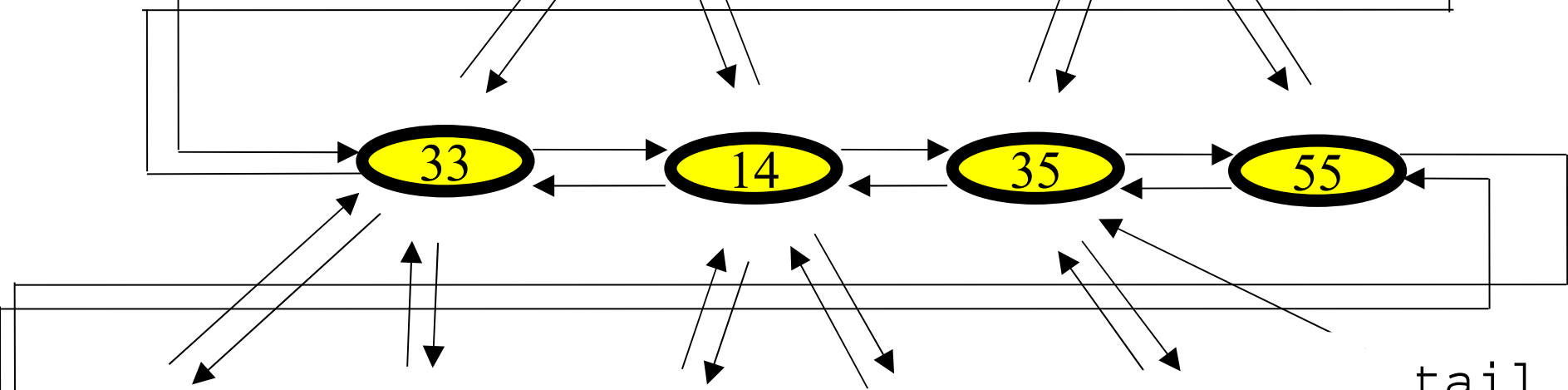
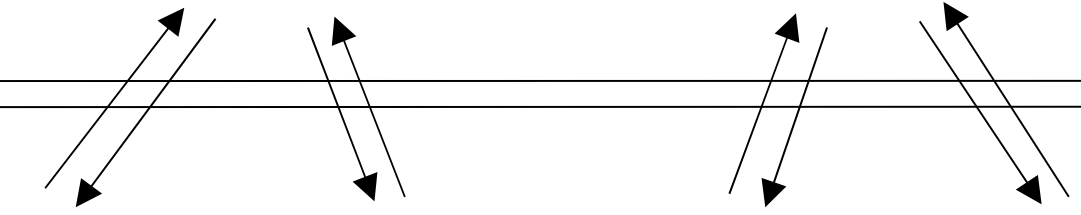
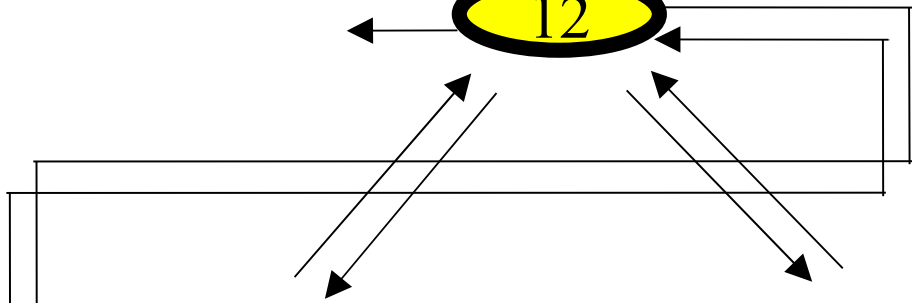
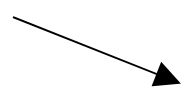
```
last = tail->leftnext->righttree;
```



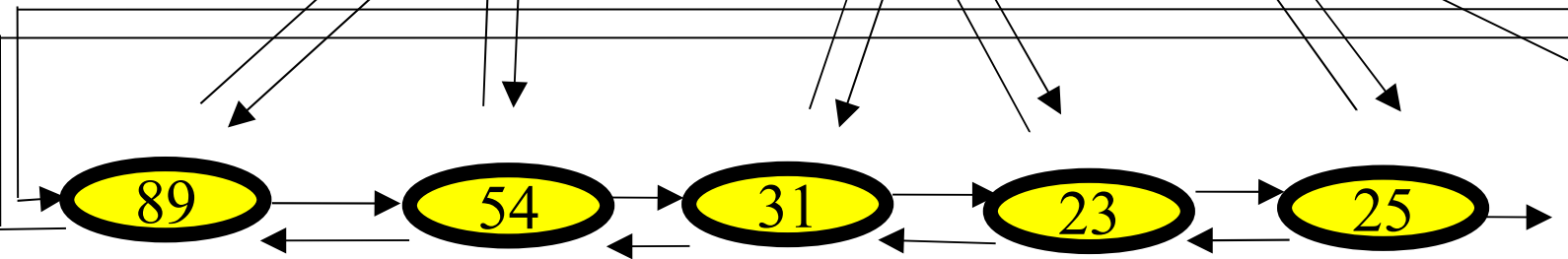


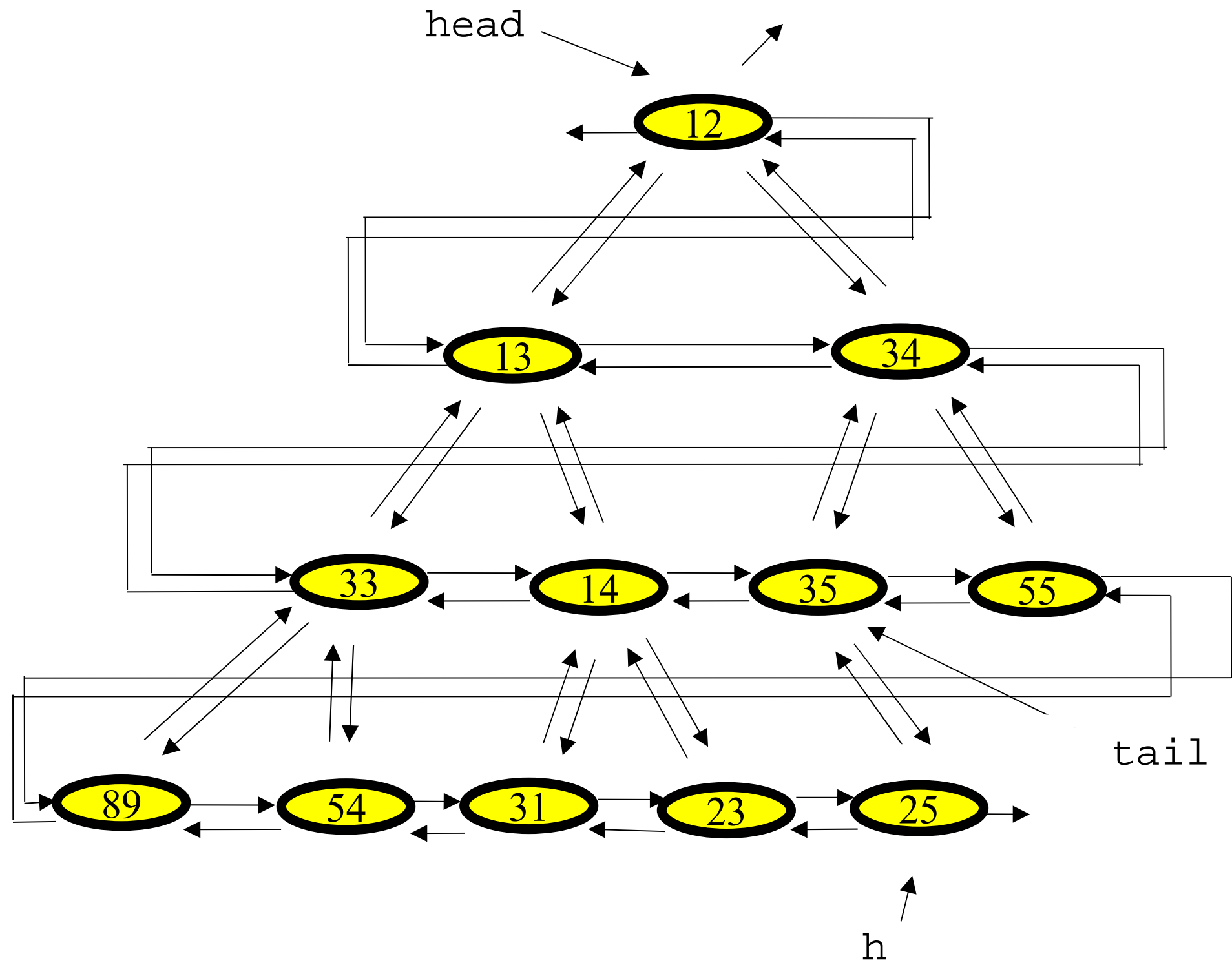


head

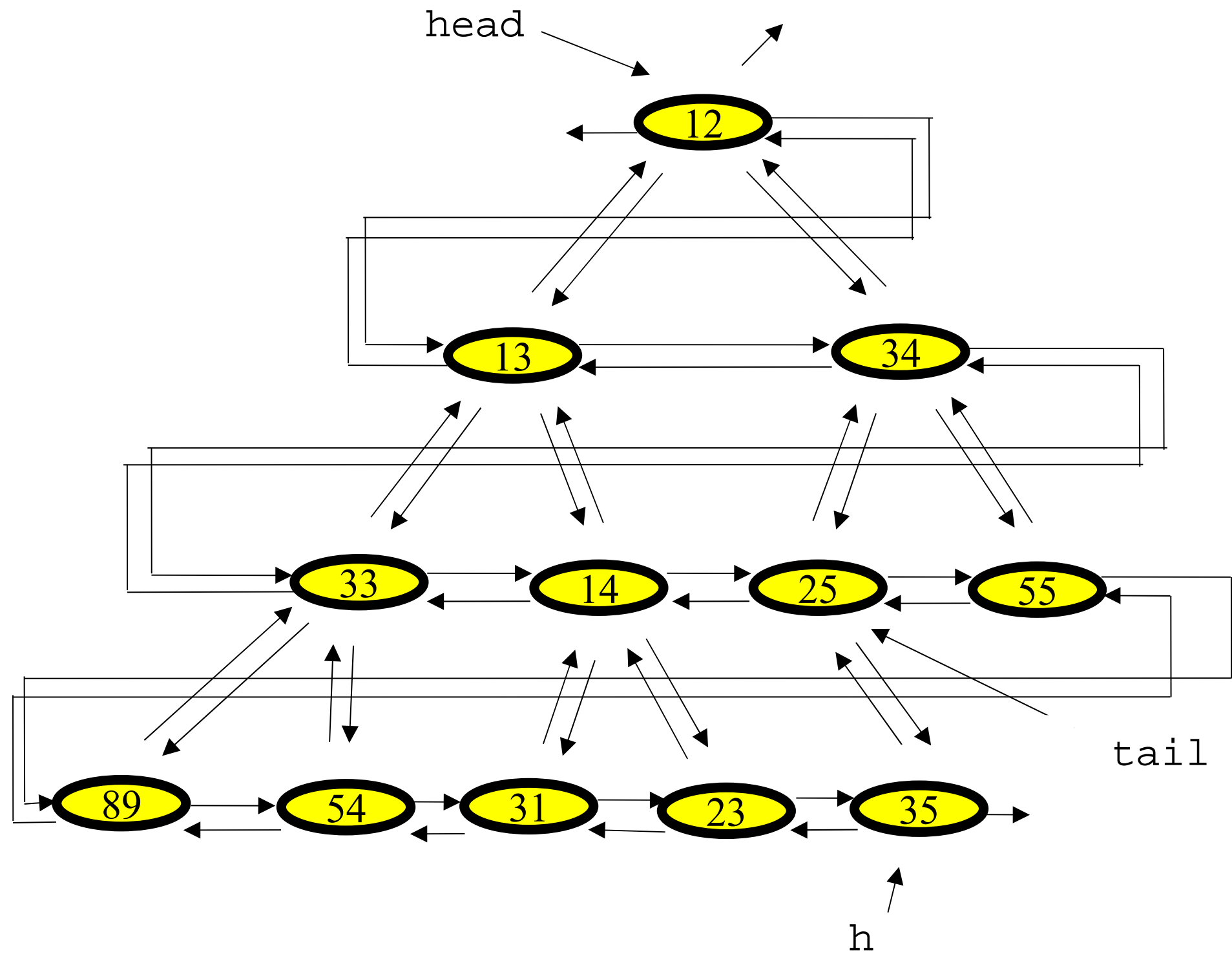


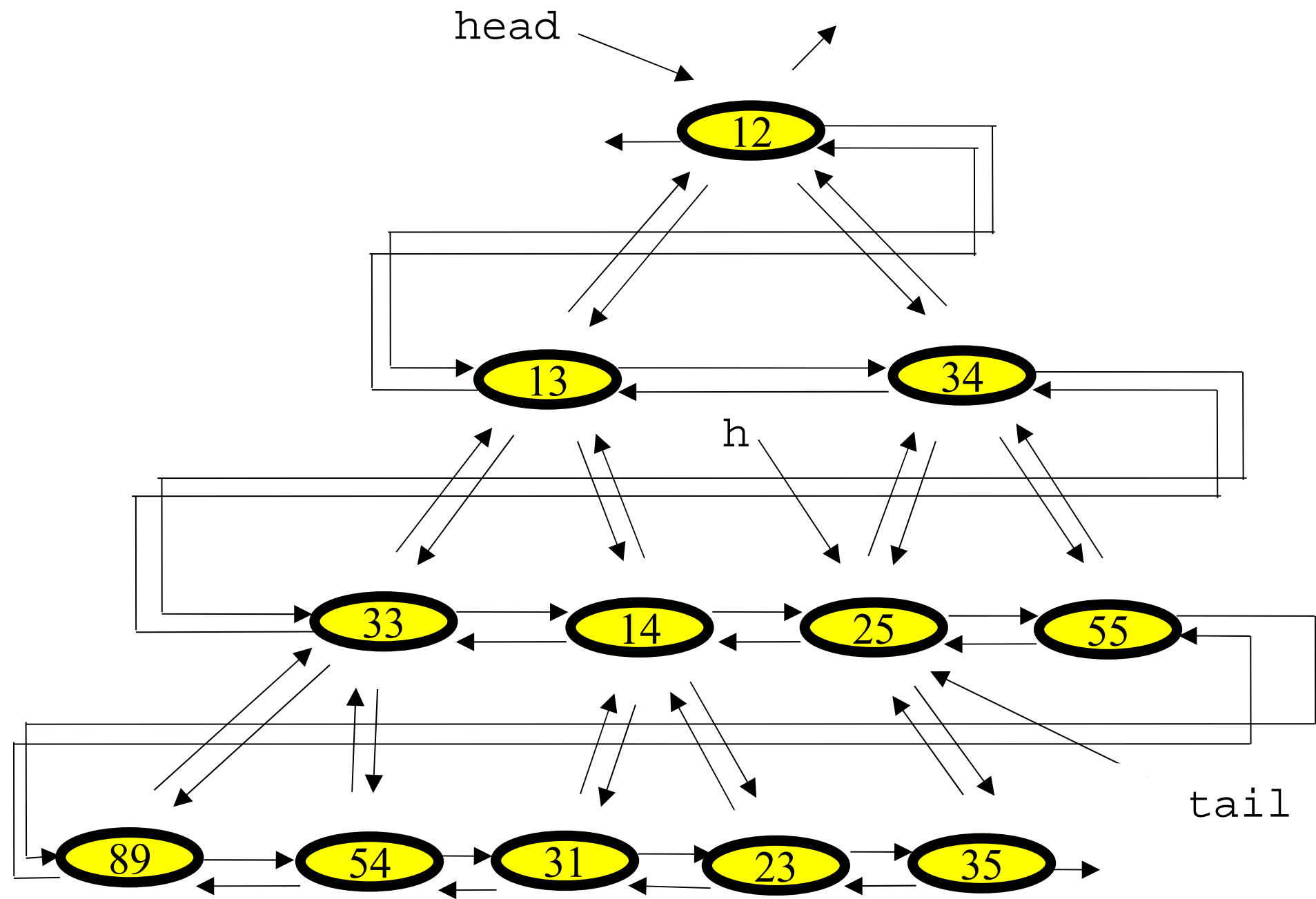
tail

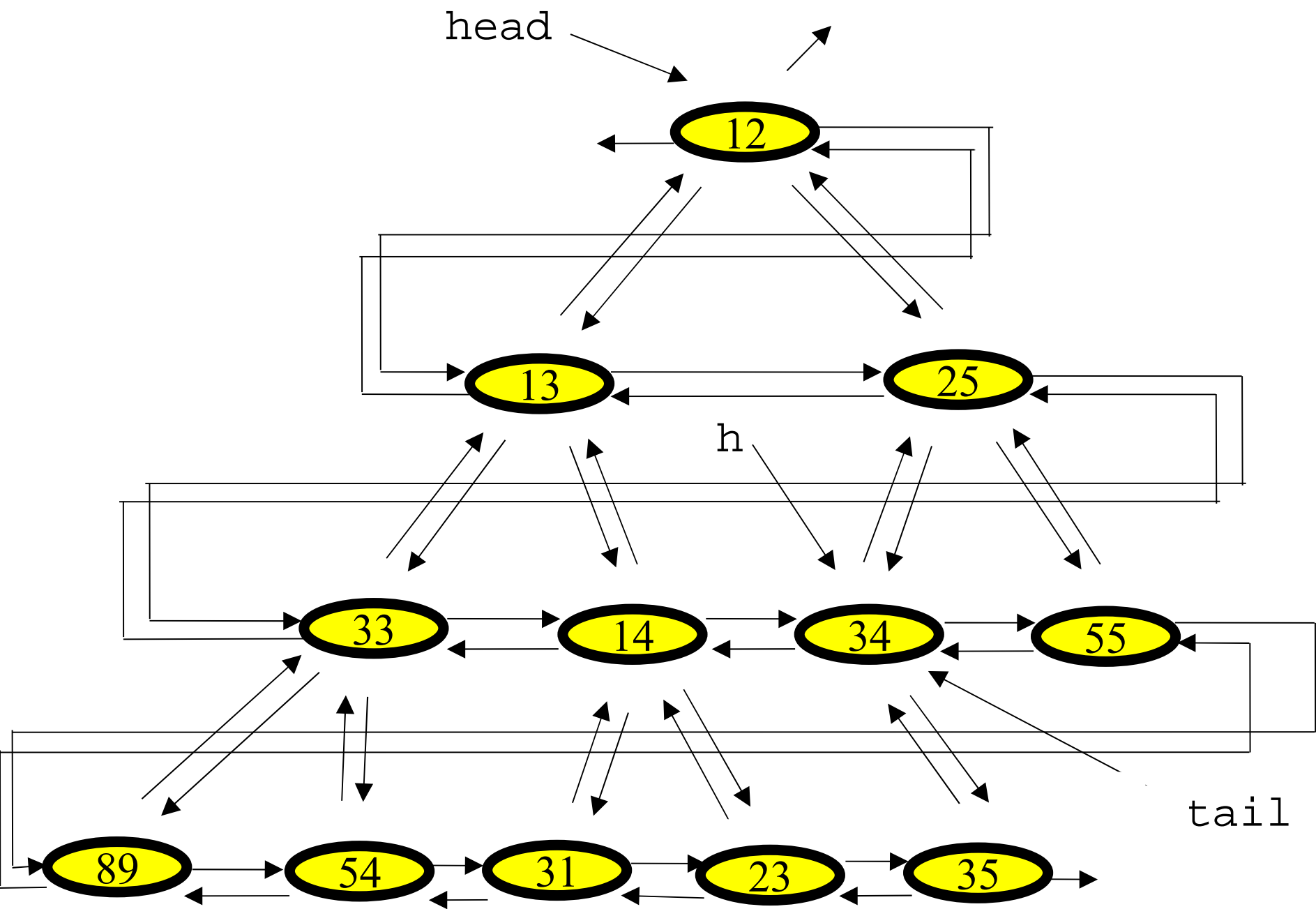


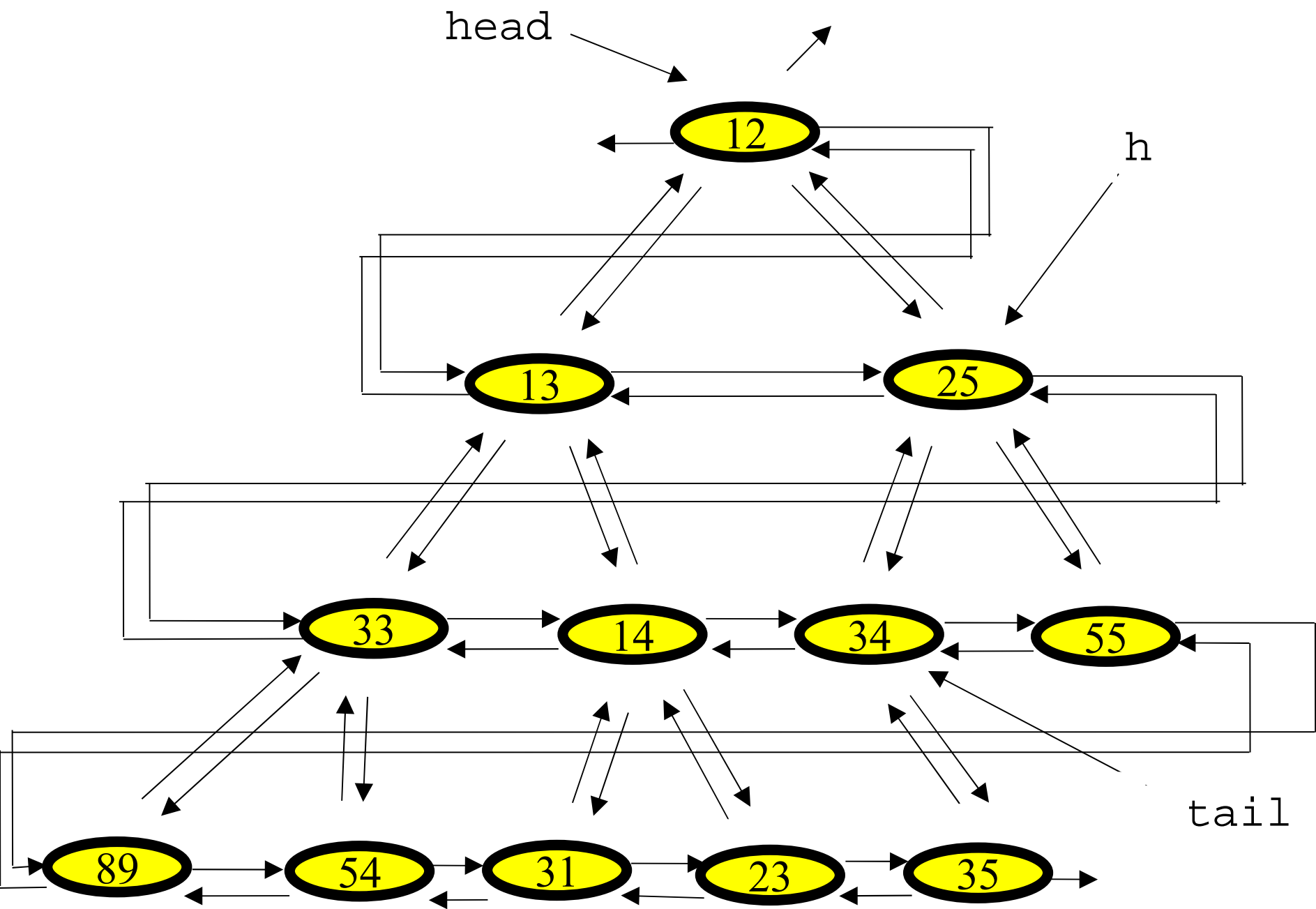




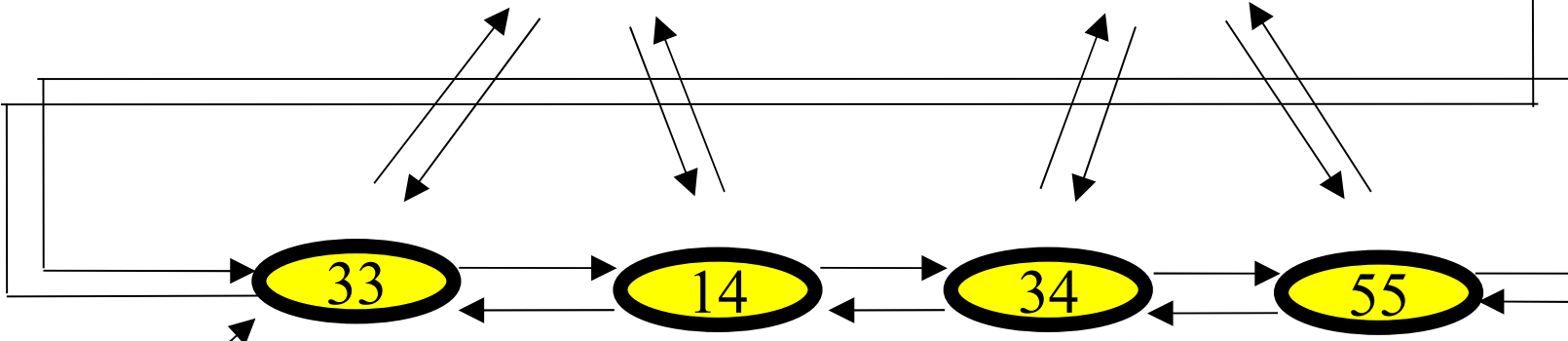
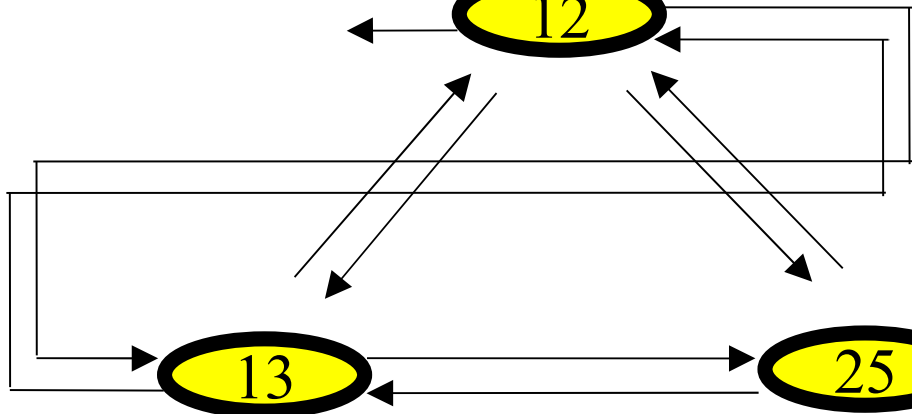
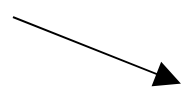








head



tail

