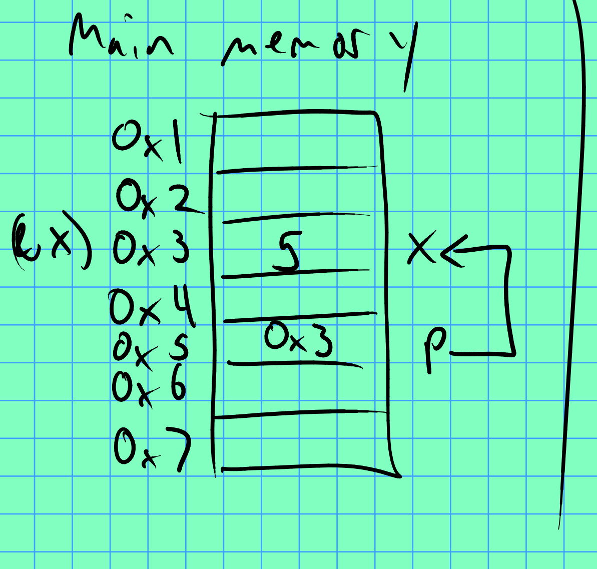
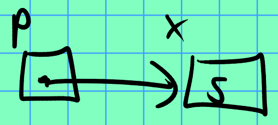
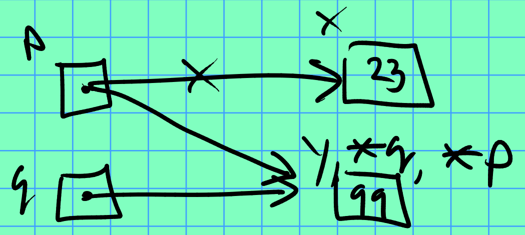
```
int x;
x = 5;
int * p;
p = &x;
```



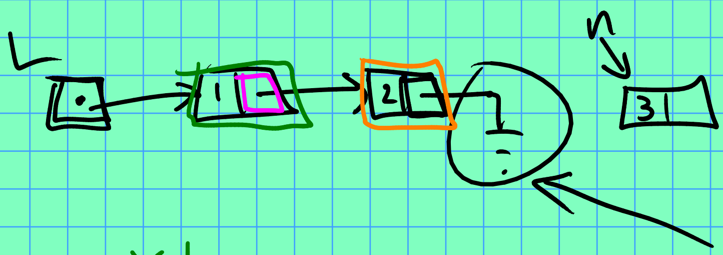


Exercise whatsthe output/7.cpp



```
int x, y;
int* p = &x, *q = &y
p = q;
```

Exercise: read stdin into list (in order).



*L

(*L).next, L->next

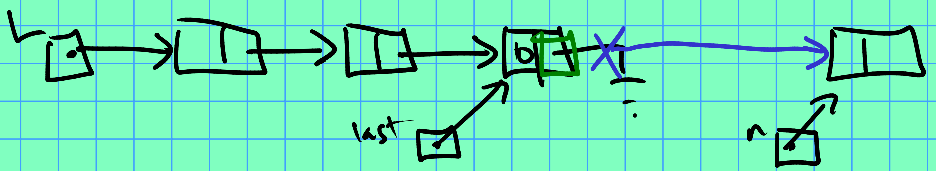
*(*L).next

(*(*L).next).next

≡

L->next->next

Gross! Let's keep track of the last node explicitly.



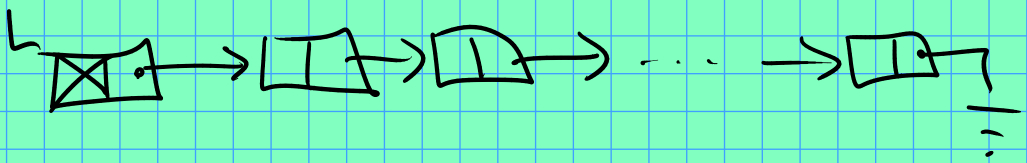
```
node* last;  // last->next is where
                      new nodes will go.

node * n = new node;
last->next = n;
```

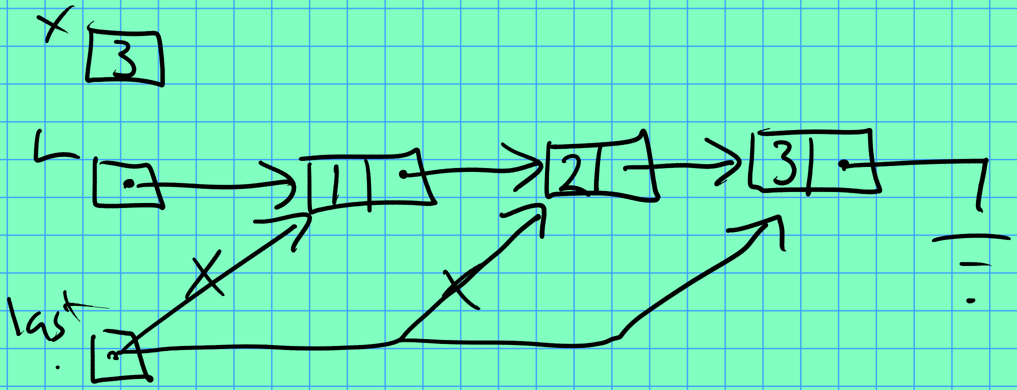
Technical nuisance: need special case for first node
(L is not the .next of any node!)

Alterative: use a "dummy" node ?

```
Node L;
```



```
echo 1 2 3 | ./lists
```



```
delete last;
```

For the next project ...