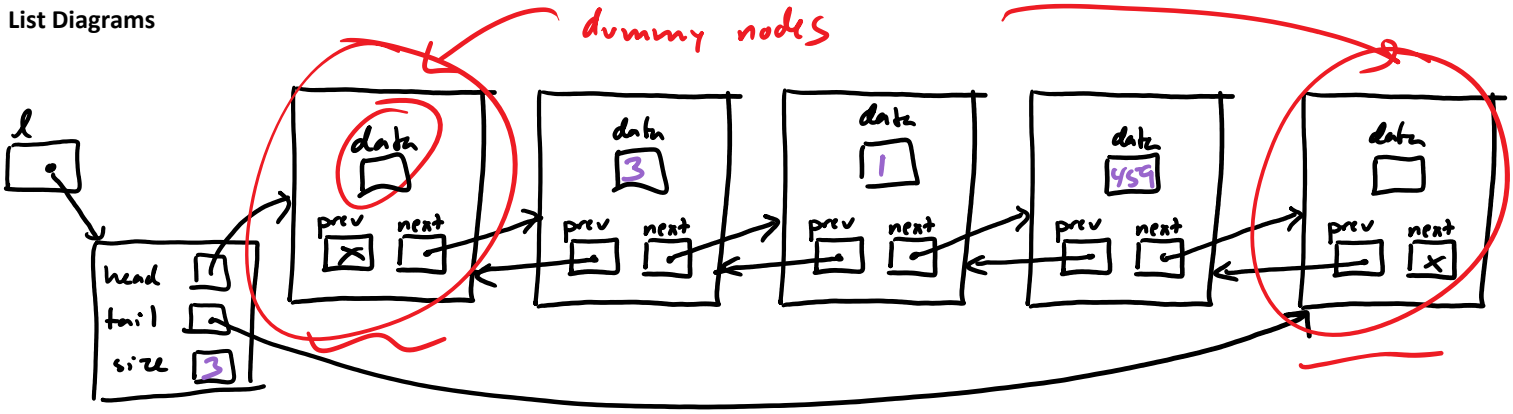
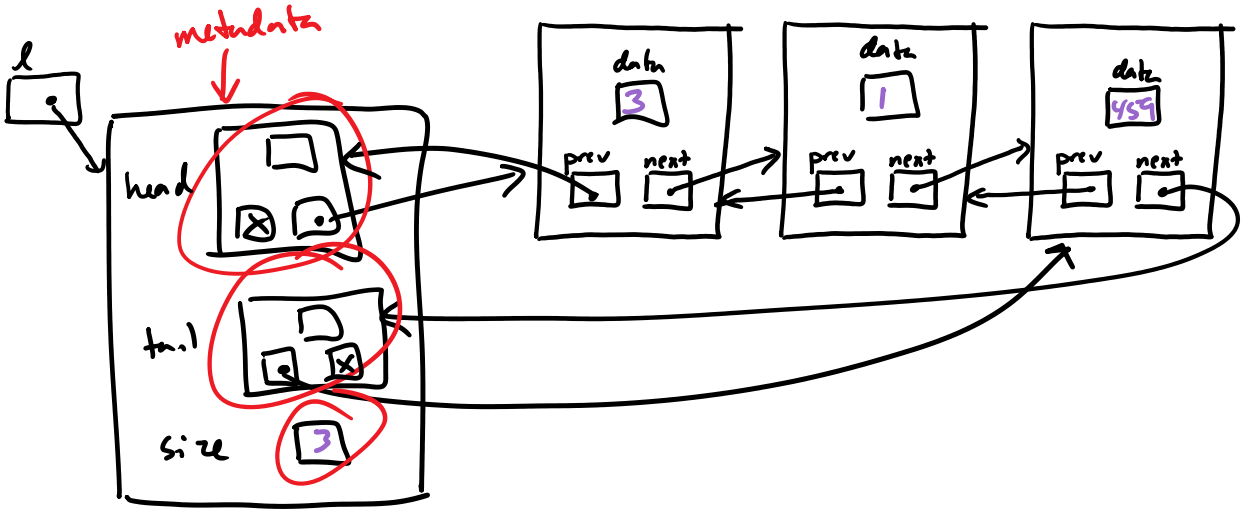
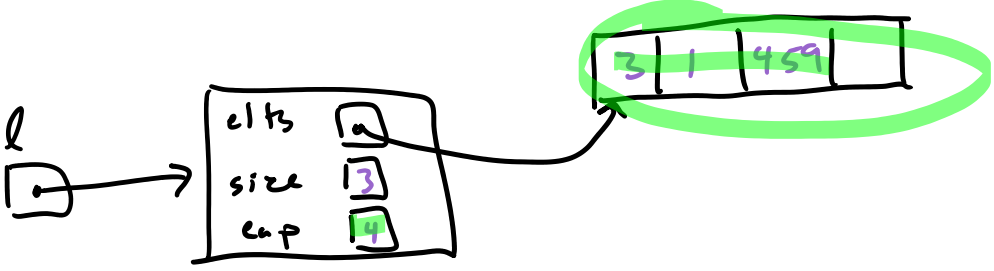
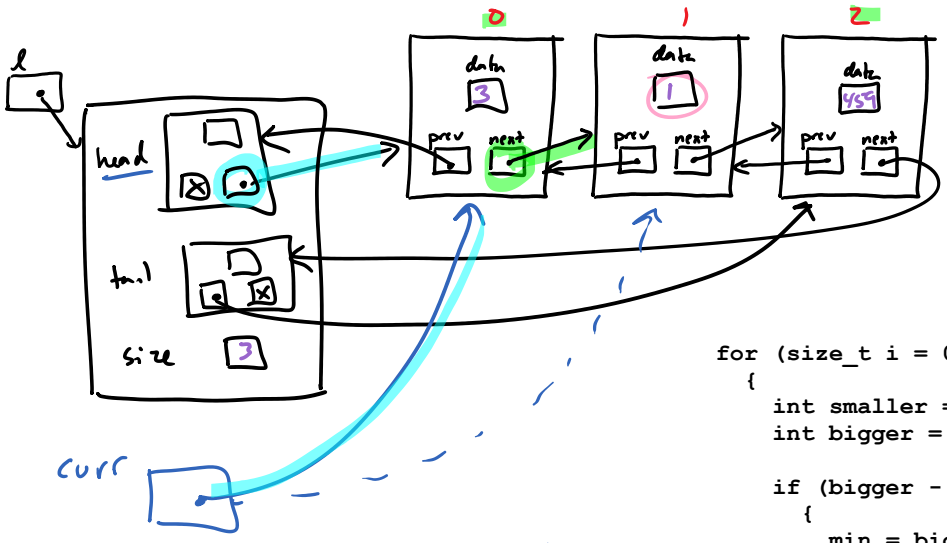
List Diagrams







Get




```
for (size_t i = 0; i < n - 1; i++)
  {
    int smaller = list_get(nums, i);
    int bigger = list_get(nums, i + 1);

    if (bigger - smaller < min)
      {
        min = bigger - smaller;
      }
  }
```

i steps

```
curr = l->head.next;
for (j = 0; j < i; j++)
    curr = curr->next;
return curr->data;
```

i: 0, 1, 2, ⋮, n-2

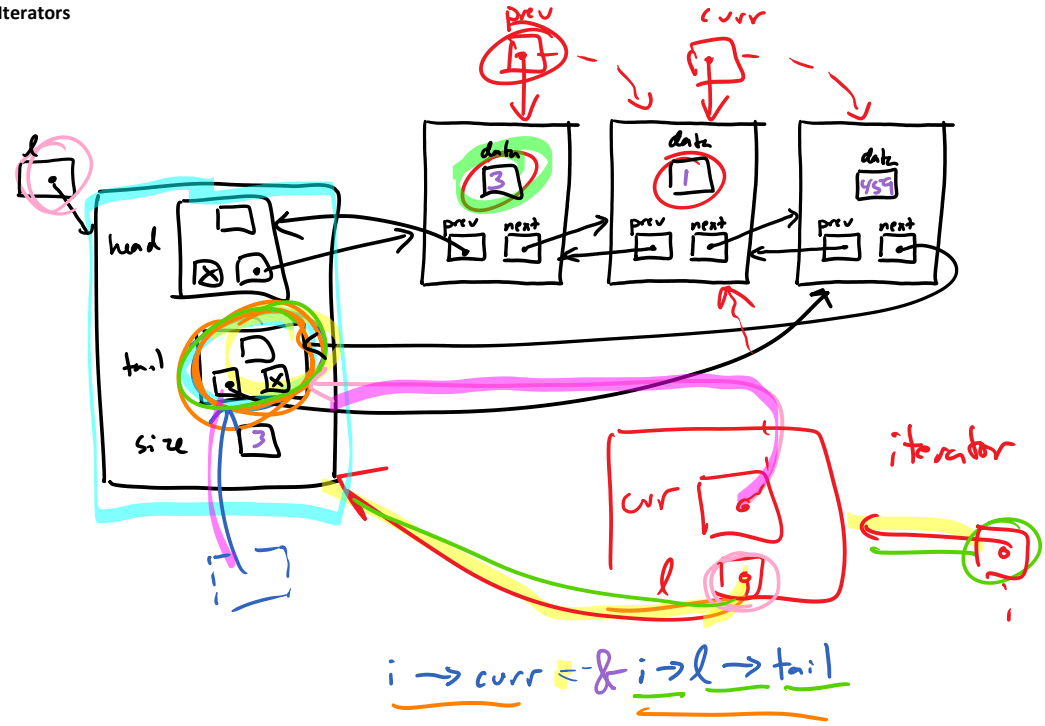
iterations in get: 0, 1, 2, ⋮, n-2

$\frac{(n-2)(n-1)}{2}$

$\frac{n^2}{2} - \cdots$

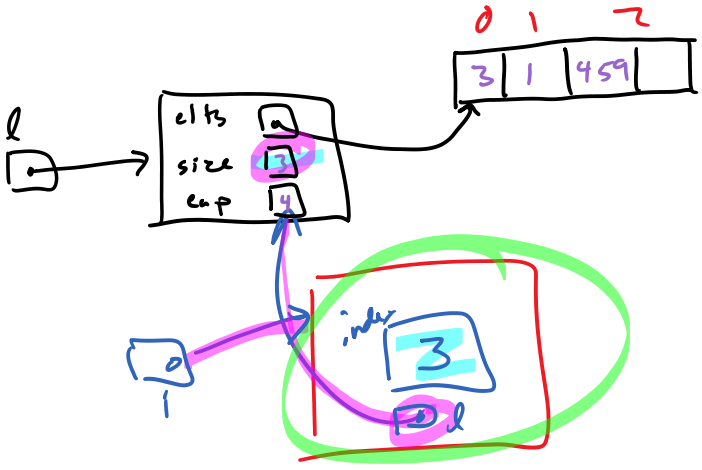
$\Theta(n^2)$

Iterators

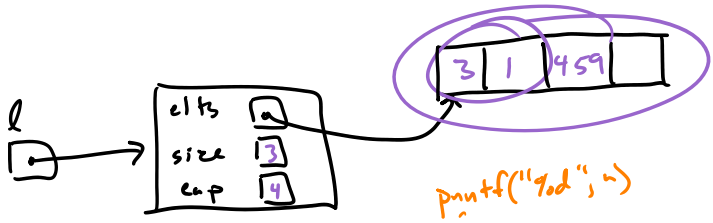


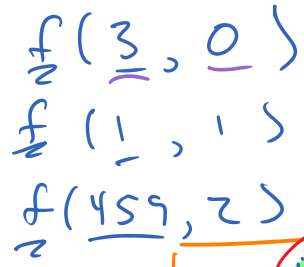
next

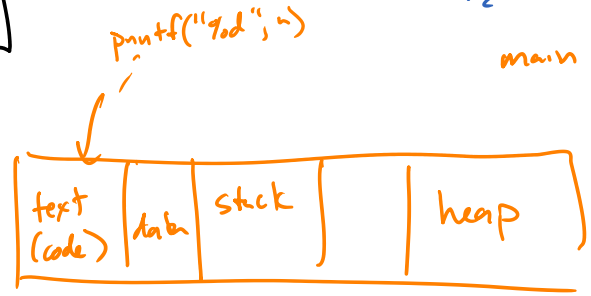
return 3

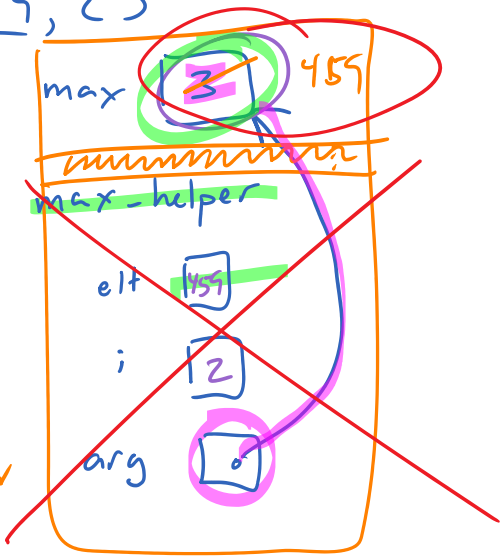


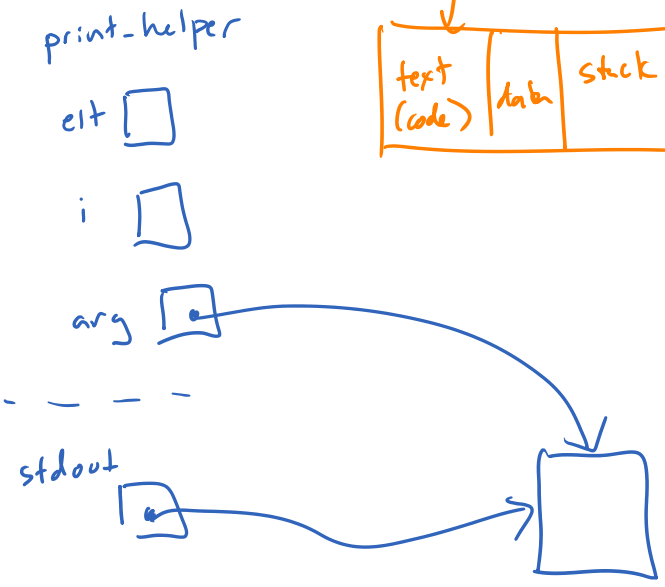
if i->index < i->l->size


l
elts
size 3
cap 4
3
1
459


f(3, 0)
f(1, 1)
f(459, 2)


printf("%d", n)
text (code)
data
stack
heap


main
max
3
459
max_helper
elt
459
i
2
arg
0
stack
max_helper


print_helper
elt
i
arg
stdout