

COP 3502 9/12/2025

- 1) Local Volunteers 1130 - 530 Sat 9/13/25
- (3) LL INTRO
- 2) GROUP INFO

Linked Lists

- 1) Draw Pictures
 - 2) Writing code
 - 3) Showing corresponding pics
 - 4) Volunteers come up we'll form LL
- malloc for dyn sized array

→

3	6	7	4	8	2	6
---	---	---	---	---	---	---

 (size 7)

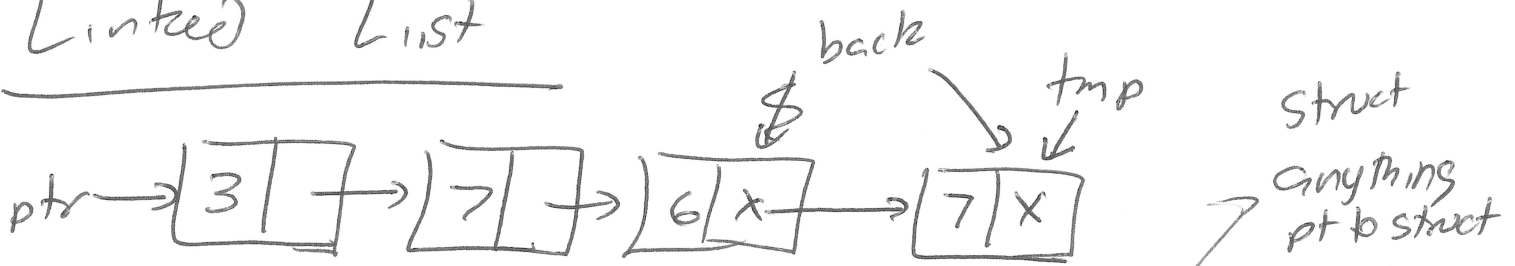
realloc

→

3	6	7	4	8	2	6						
---	---	---	---	---	---	---	--	--	--	--	--	--

arrays we use realloc to resize when we need to. (Double in size when full
1/2 in size when < 25% full)

Linked List



typedef struct node {

int data;

struct node* next;

} node;

node* createNode(int num) {

node* tmp = malloc(sizeof(node));

tmp->data = num;

tmp->next = NULL;

return tmp;

node* insertFront(node* list, int num) {

node* tmp = createNode(num);

tmp->next = list

return tmp;

main

ptr → NULL

ptr = createNode(5);

ptr = insertFront(ptr, 7);

insert Front

list num [7]

tmp

~~create Node~~

~~num [5]~~

~~tmp →~~



Why doesn't print function lose the ptr to front of the list?

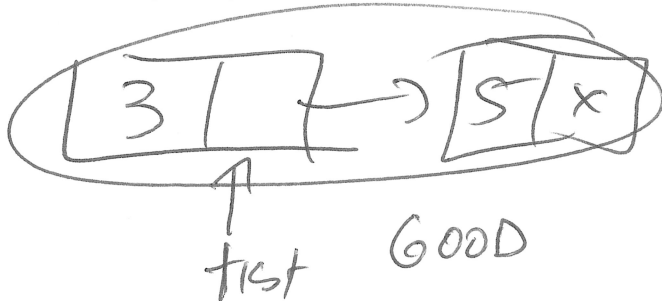


Sumlist function
iteratively
recursive

is Sorted

sumlist (→ [2] → [3] → [1] → X)
= 2 + sumlist (→ [3] → [1] → X)

[3] → X ✓ yes



Check

list → next
ISN'T NULL

list → data to (list → next) → data

NULL → SOMETHING

NULL PTR ERROR

if this NULL
then whole
thing is a
RUN TIME
ERROR