

COP 3502 - 1/9/2024

## Dynamic Memory Allocation

Static memory is known at compile time

```
int num;  
char word[100]; //etc.
```

What about?

```
int n; scanf("%d", &n);
```

```
int data[n]; → BAD DON'T DO!
```

Correct ways:

```
int* data = malloc(sizeof(int) * n);
```

```
OR int* data = calloc(n, sizeof(int));
```

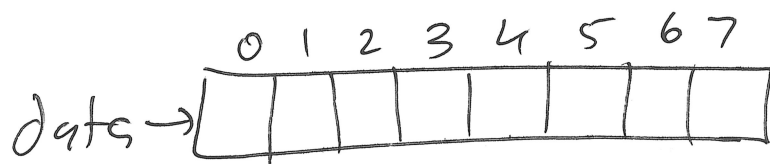
### Why use DMA?

① Don't know how much mem I need at compile time!

② I want memory allocated in a func to exist after the function is done running

③ I'm using lots of memory!

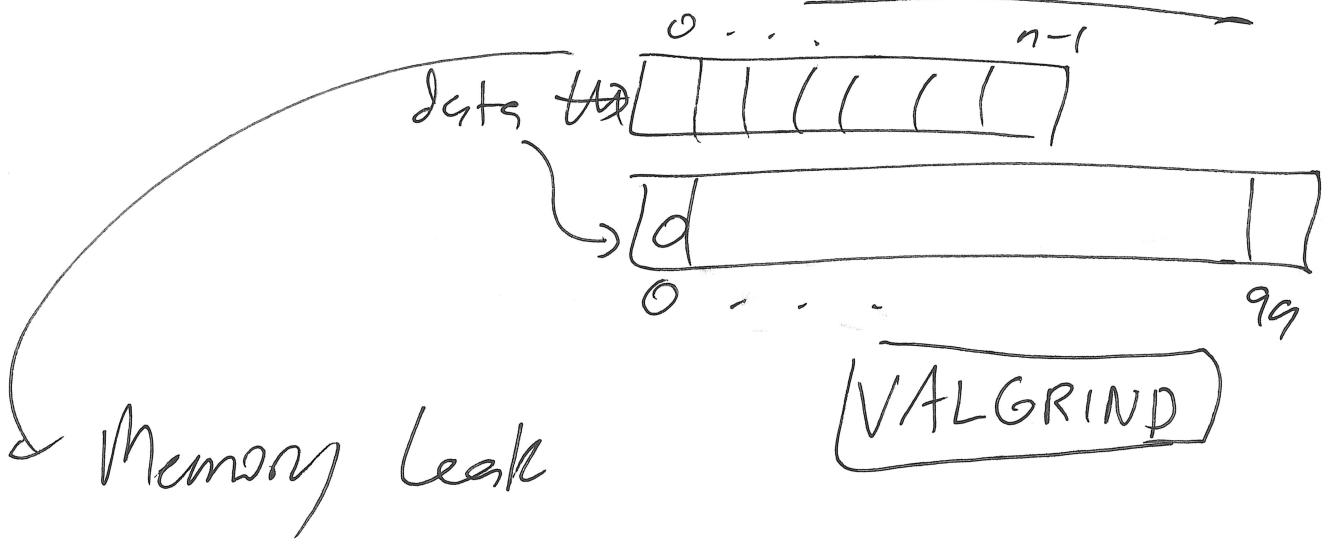
n [8]



free(data)

```
int* data = malloc(n * sizeof(int));
```

```
data = malloc(100 * sizeof(int));
```



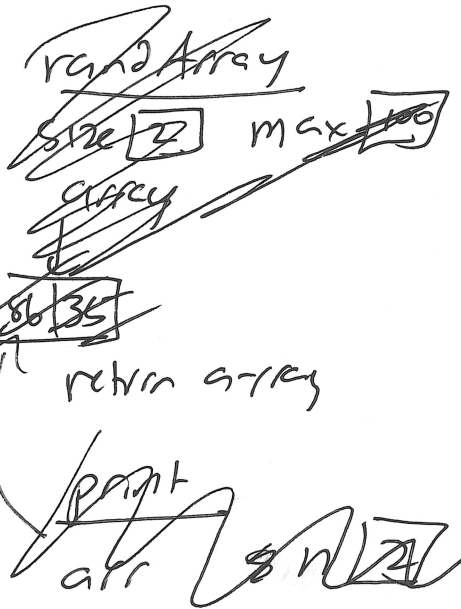
# Pic of 2mapractice

main

tmp = randArray(...)

print(tmp, 3+i, 2)

free(tmp)



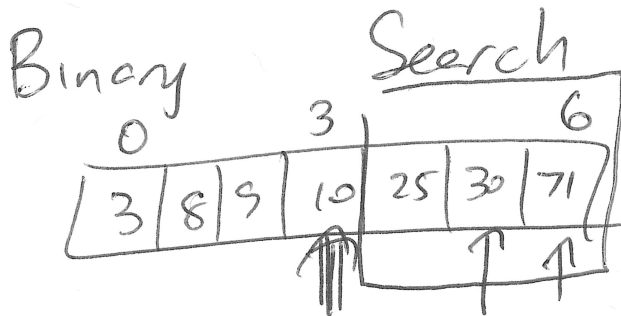
Searching for an item in a sorted array.

3	8	9	10	25	30	71
✓	✓	✓	✓	✓	✓	✓
✗	✗	✗	✗	✗	✗	✗

find 41

no

Run-time :  $O(n)$ .



41

8

- $L=0, H=6 \rightarrow m=3$
- $L=4, H=6 \rightarrow m=5$
- $L=6, H=6 \rightarrow m=6$
- $L=6, H=5$  no

$L=0, H=6, m=3$   
 $L=0, H=2, m=1$   
 found

# Bin Search Analysis

$$n \rightarrow n/2 \rightarrow \frac{n}{4} \rightarrow \frac{n}{8} \dots$$

After  $k$  steps search space

$$\frac{n}{2^k} = 1$$

$$n = 2^k$$

$$k = \log_2 n$$

getline function