

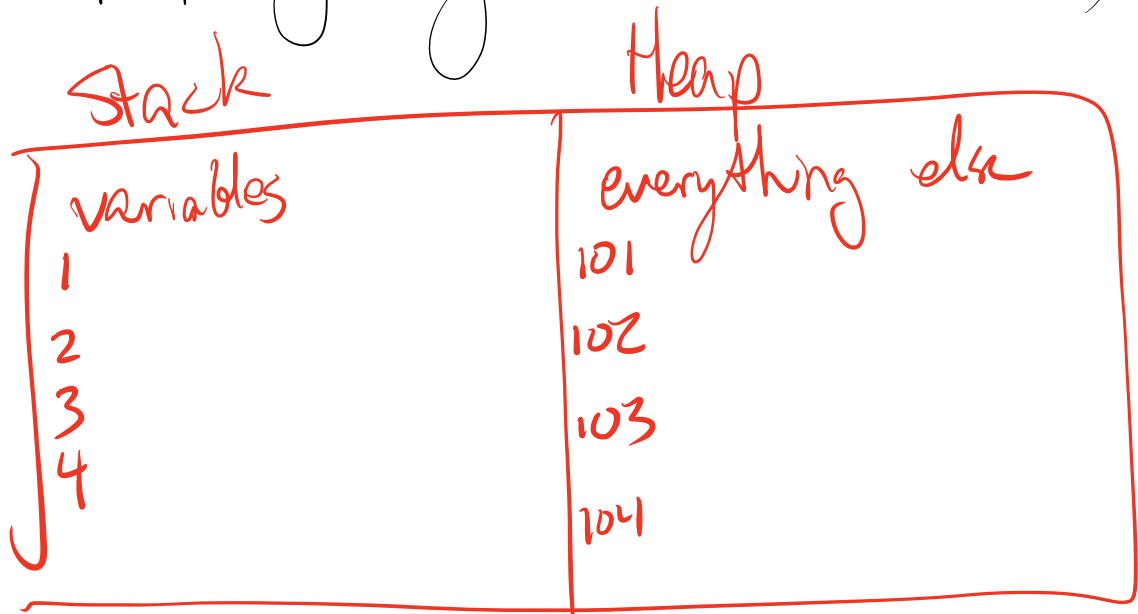
What is memory?

bits

bytes

blocks/words/"slots"

How is memory organized? (in C etc)



What does C do?

```
int x = 75;
```

```
int arr[5];
```

```
arr[0] = x;
```

```
arr[1] = sumof(x, arr[0]);
```

```
scanf ("%d", &x);
```

address of

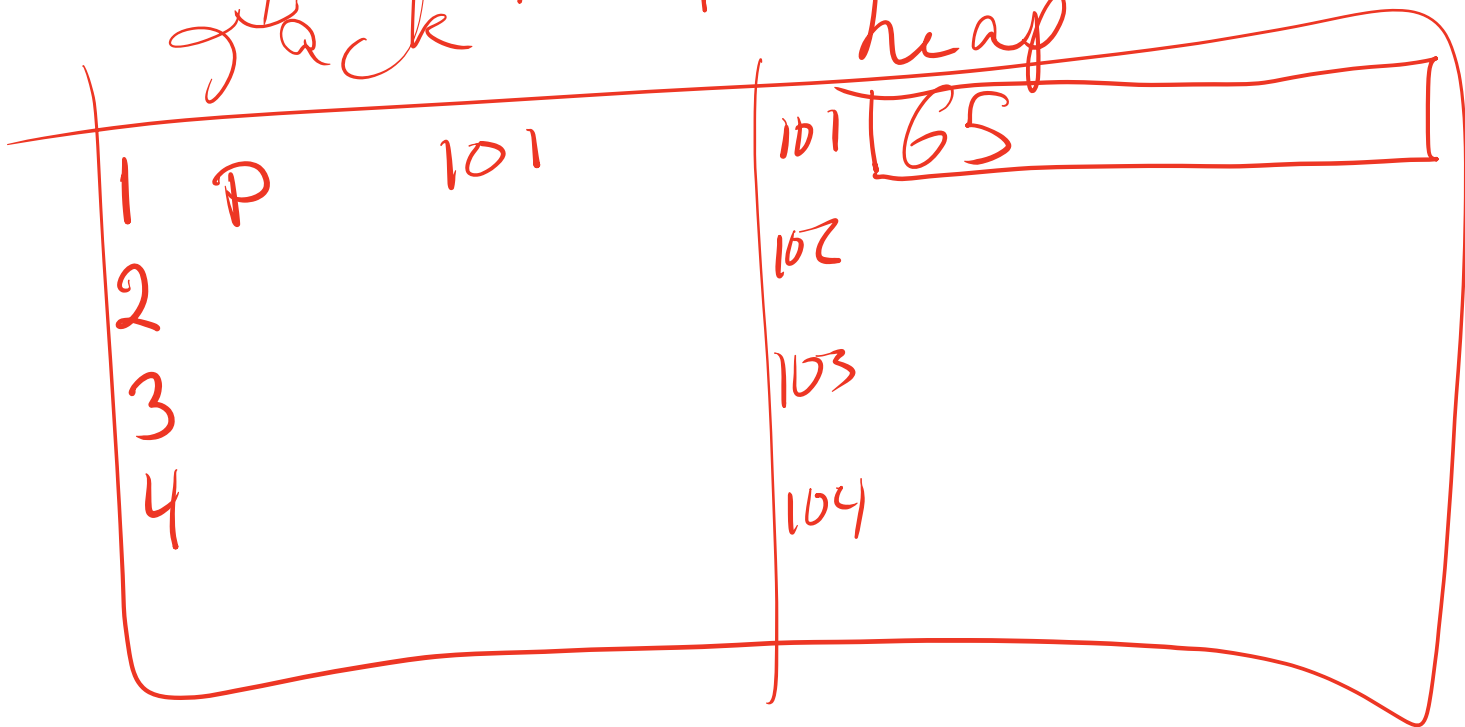
$(\&arr[0])$
2

```
int main()
{
    size_t x = 0;
    scanf("%zu", &x);
    if (x < 100) {
        double arr[x];
        initialize(arr);
    }
    printf("%lf\n", arr[0]);
}
```

| Stack | Heap |
|-------------|------|
| 1 x 77 | 101 |
| 2 arr[0] 75 | 102 |
| 3 [1] 150 | 103 |
| 4 [2] | 104 |
| 5 [3] | 105 |
| 6 [4] | 106 |

Why the name "stack"?

stack int *p = malloc(sizeof(int));
heap

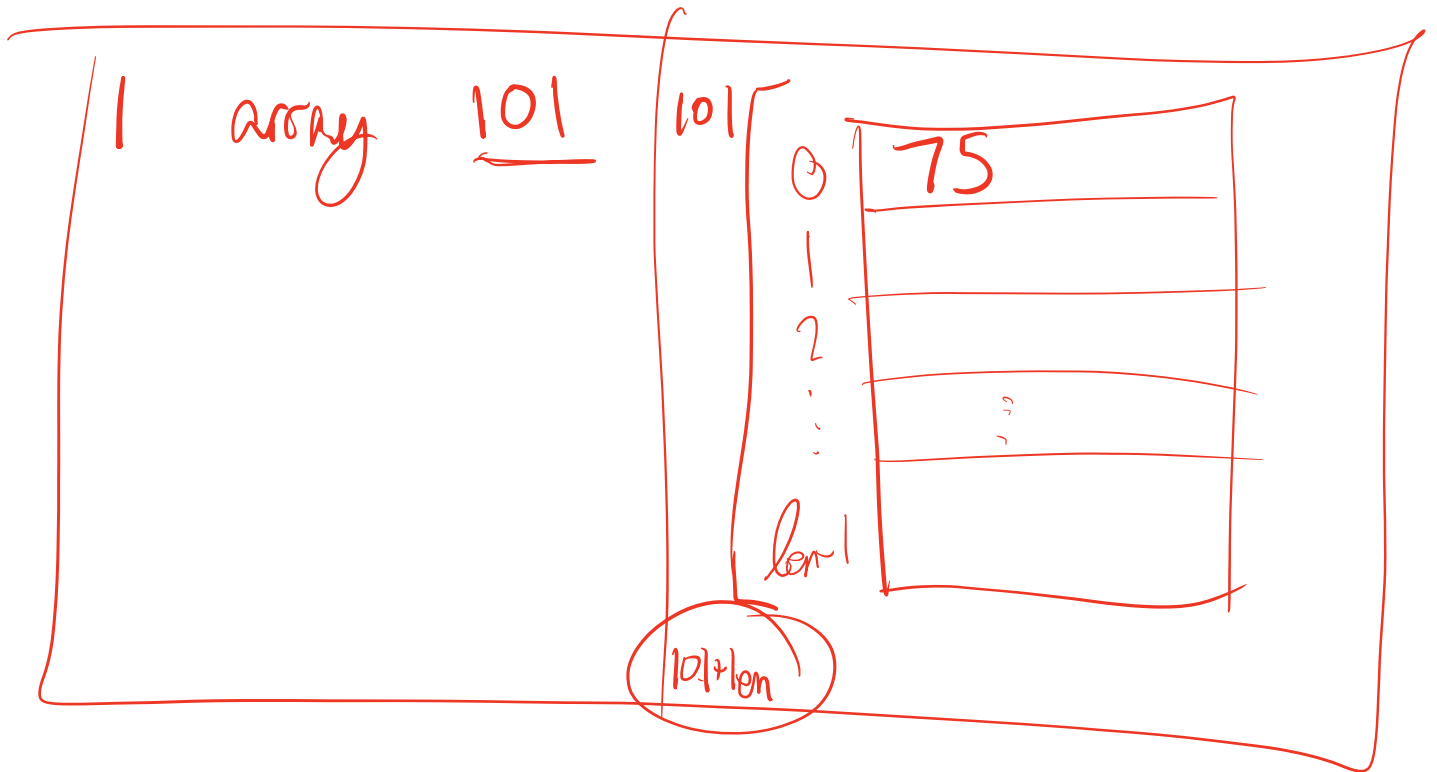


*p → 65 p → 101

&p → 1 &*p → 101 == p

*&p → 101

$\text{int}^* \text{array} = \text{malloc}(\text{len} * \text{sizeof}(\text{int}));$



$\text{array}[0] = 75;$

$*(\text{array} + n) = 75$

int * array = malloc(... -4)

for (... -) {

arr[i] = i * 3;

}

| | | | | |
|---|-----|-----|-----|---|
| 1 | arr | 101 | 101 | 0 |
| | | | 102 | 3 |
| | | | 103 | 6 |
| | | | 104 | 9 |

for (... -)

printf(..., arr[i]);

`int * arr = malloc(... 4);`

`arr = malloc(... 8);`

