

A job ready bootcamp in C++, DSA and IOT

DMA



Saurabh Shukla (MySirG)

Agenda

- ① SMM vs DMA
- ② malloc()
- ③ Type casting
- ④ calloc()
- ⑤ Memory leak
- ⑥ free()
- ⑦ realloc()

SMA

Static Memory Allocation

```
int x;  
float y;  
int *p;  
int a[10];  
struct Book b1;
```

DMA

Dynamic Memory Allocation

```
malloc()  
calloc()
```

malloc()

float *P;

P = malloc(4);



Standard Practice

P = malloc(sizeof(float));

Void* malloc(unsigned int s)

{

return address;

3

Type casting

```
int *P;
```

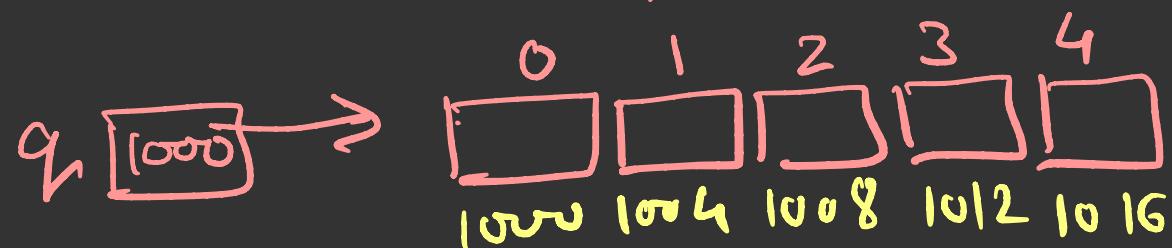
```
P=(int *)malloc(sizeof(int))
```

(int *) 1000

calloc()

int *q;

q = calloc(5, 4);



Standard practice

q = (int *)calloc(5, sizeof(int));

Malloc vs Calloc

① one argument

② Garbage value

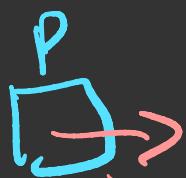
③ Single block

① two arguments

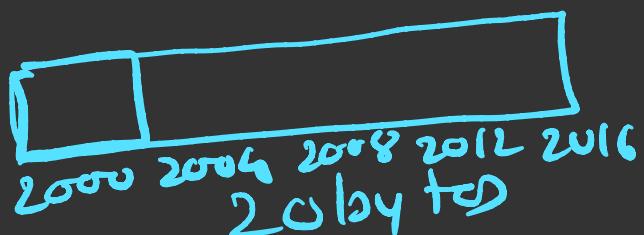
② 0 zero

③ Array of blocks

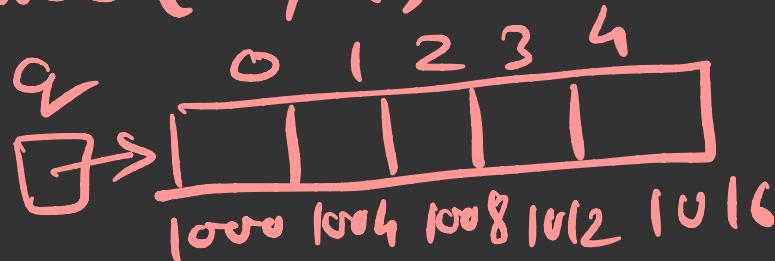
$\text{int} * p$



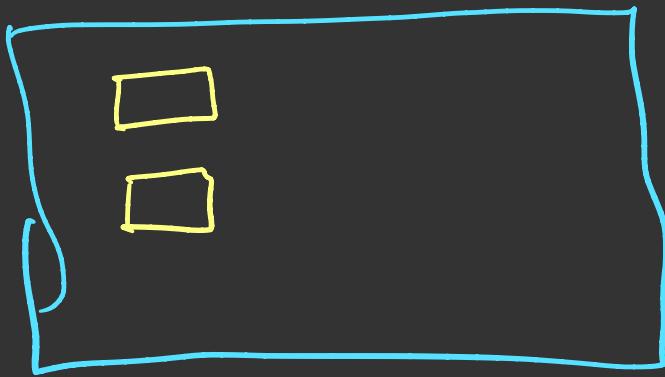
malloc (20)



calloc (5, 4)



Memory Leak



Program's
memory



f1() {
 {

 int *P;

 P = malloc(4);

 *P=5;

 :::
 :::

Total memory = consumed + free

}

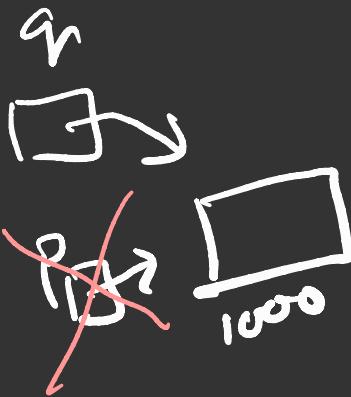
{

int *q;

q=f1();

{

int* f1()



int* p;

p=malloc(4);

==

return p;

}

free()

free() is used to release memory of DMA variables only

free(address);

```
void f() { int *p;  
    p = malloc(4);  
    ...  
    free(p);  
    p = NULL;
```

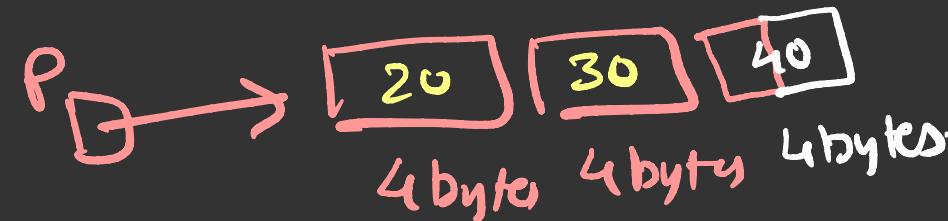
{}

realloc()

realloc(pointer, newsize)

ptr = malloc(6);
q = realloc(ptr , 10);

```
int *p = malloc(10);
```



```
*p = 20;
```

```
* (p+1) = 30;
```

```
* (p+2) = 40; ← wrong
```