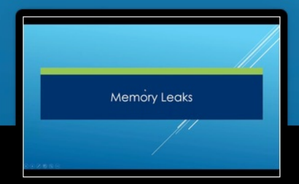
Memory Leaks

When we loose access to memory that is dynamically allocated



Reassignment of stack address to active dynamic address pointer

```
int *p_number {new int{67}}; // Points to some address, let's call that address1

//Should delete and reset here

int number{55}; // lives at address2

p_number = &number; // Now p_number points to address2 , but address1 is still in use by
                    // our program. But our program has lost access to that memory location.
                    //Memory has been leaked.
```

```
//Double allocation
int *p_number1 {new int{55}};

//Use the pointer

//Should delete and reset here.

p_number1 = new int{44}; // memory with int{55} leaked.
```

Pointer in local scope

```
#include <iostream>

int main(int argc, char **argv)
{
    [ ]
    [...]

    {
        int *p_number2 {new int{57}};
    }
    //Memory with int{57} leaked.

    return 0;
}
```