

# #defines

Last updated 6/13/23

These slides describe the #define statement

# #define

- # - indicates a pre-processor command
  - Placed in the global area of the code
- #define – indicates a find/replace action in your code, before any other processing
  - Allows the value to be changed in one place (the #define) even though it is used in dozens of places in your code
  - You do not see the substitution (see warning slide)

# #define

- Example
  - Change the delay to 4s

4 changes required

```
...  
Sleep(3000);  
...  
Sleep(3000);  
...  
Sleep(3000);  
...  
Sleep(3000);
```

VS

```
#define DELAY 3000 // (3sec)
```

1 change → 4 changes

```
...  
Sleep(DELAY);  
...  
Sleep(DELAY);  
...  
Sleep(DELAY);  
...  
Sleep(DELAY);
```

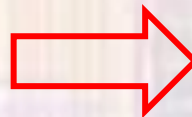
# #define

- **Warning**
  - Accidentally adding a ; can create hard to find errors

```
#define DELAY 3000; // (3sec)
```

...  
Sleep(DELAY);

This is what you see



...  
Sleep(3000;);

This is what is used



```
Error:  
..\intro_to_prog.c:41:8: note: in expansion of macro 'DELAY'  
Sleep(DELAY);
```

Error is not clear