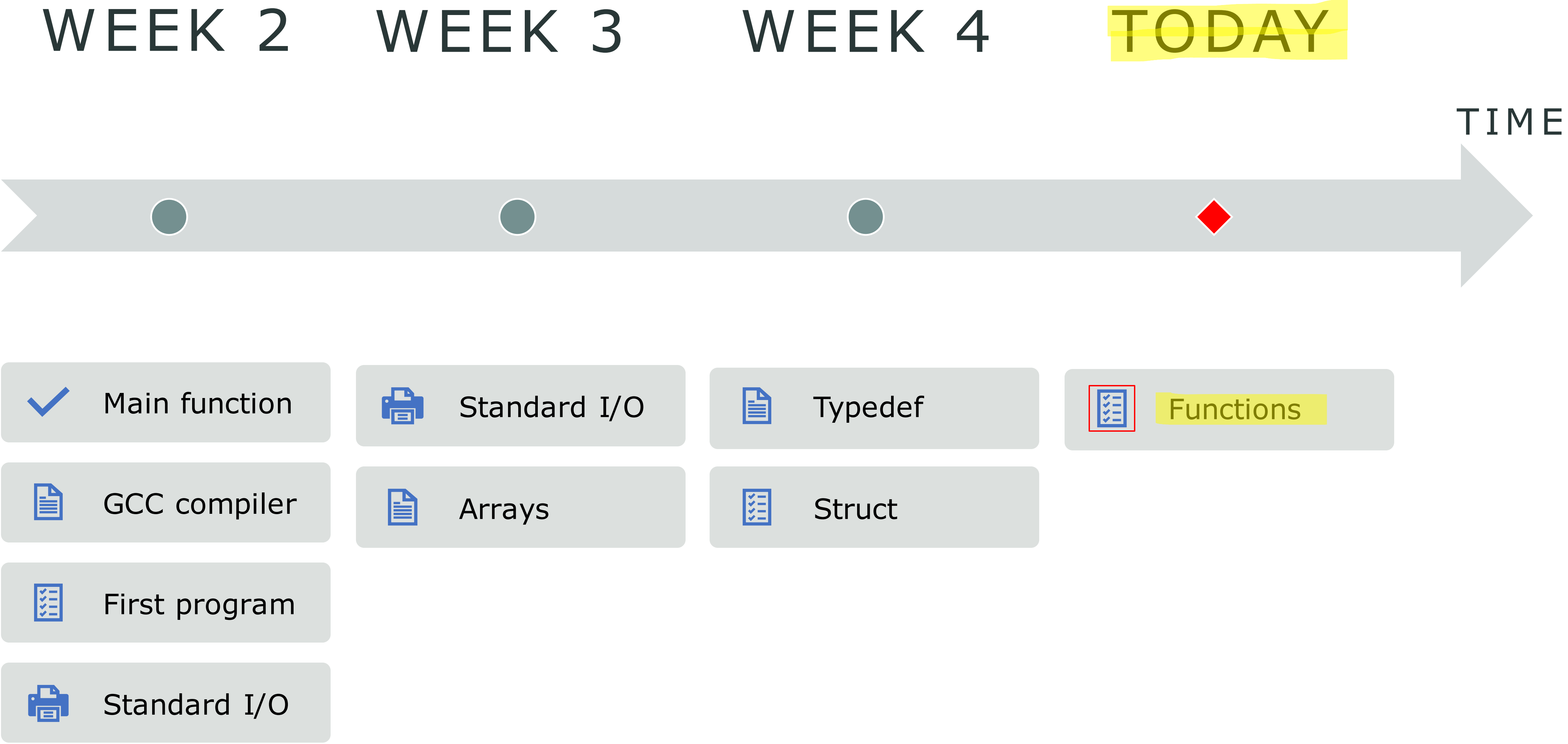
WELCOME TO CSCUV4

WEEK 5

Cristian Lepore

PLAN FOR TODAY

15 mins	–	Presentation
10 mins	–	Q&A
35 mins	–	Exercises & checkpoints



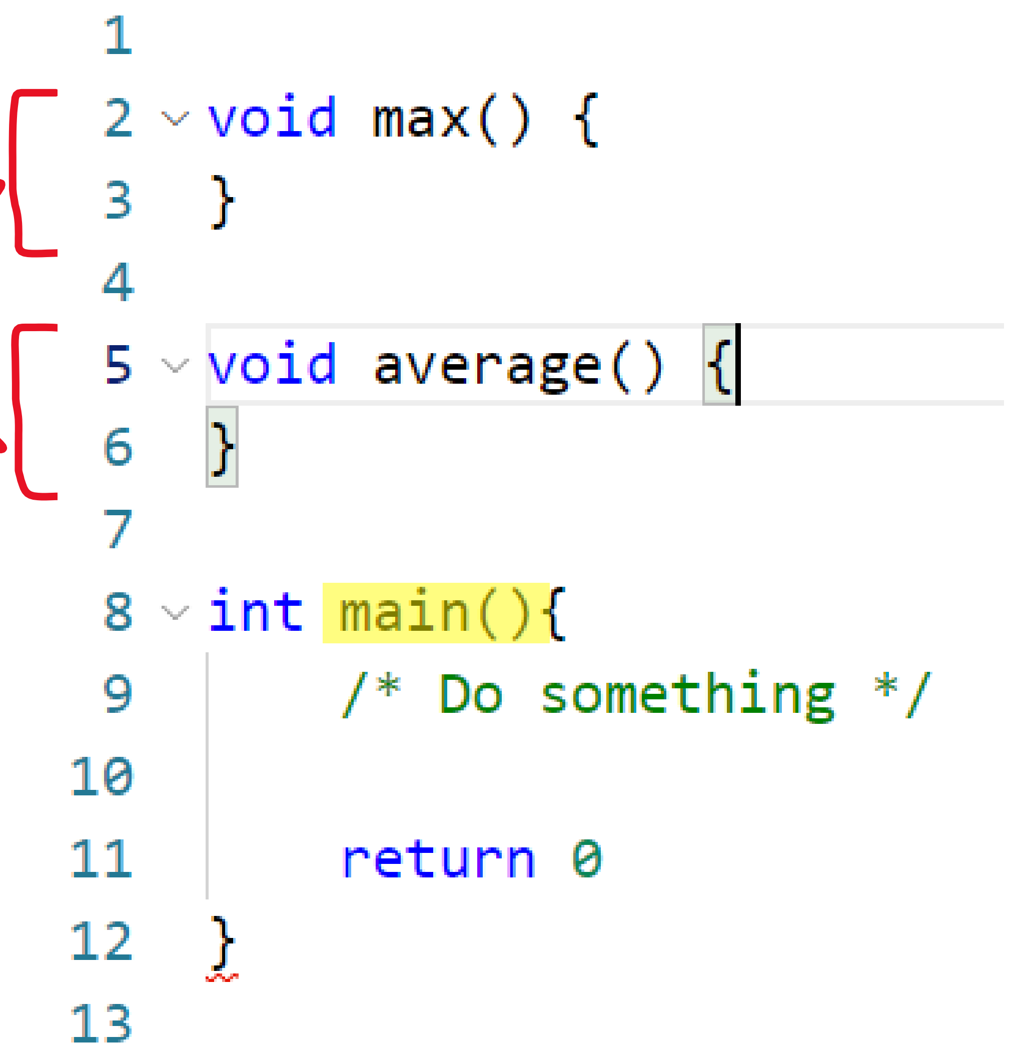
TIME

WEEK 2	WEEK 3	WEEK 4	TODAY
✓ Main function	Standard I/O	Typedef	Functions
GCC compiler	Arrays	Struct	
First program			
Standard I/O			

ABOUT FUNCTIONS

1. Defining a Function
2. Function Declarations
3. Calling a Function
4. Function Arguments

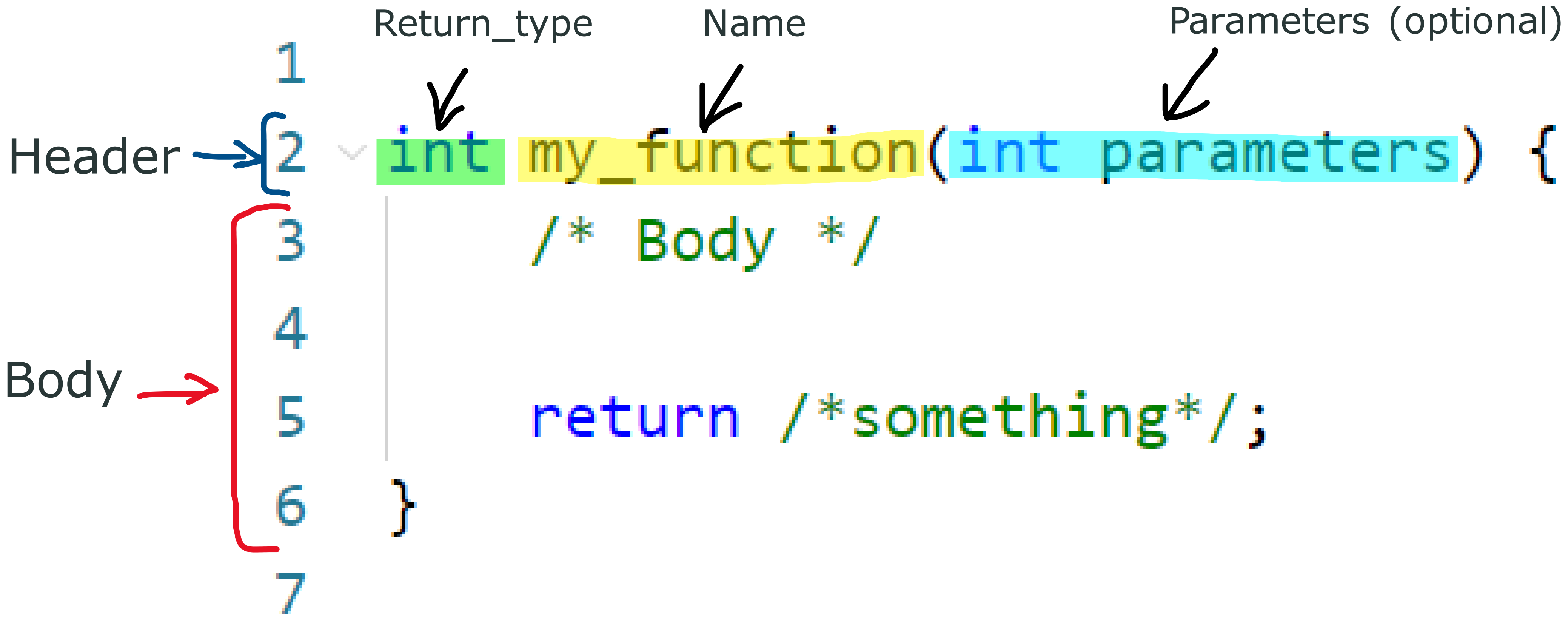
WHAT IS A FUNCTION



```

void max() {
}

void average() {
}

int main(){
    /* Do something */

    return 0
}

```

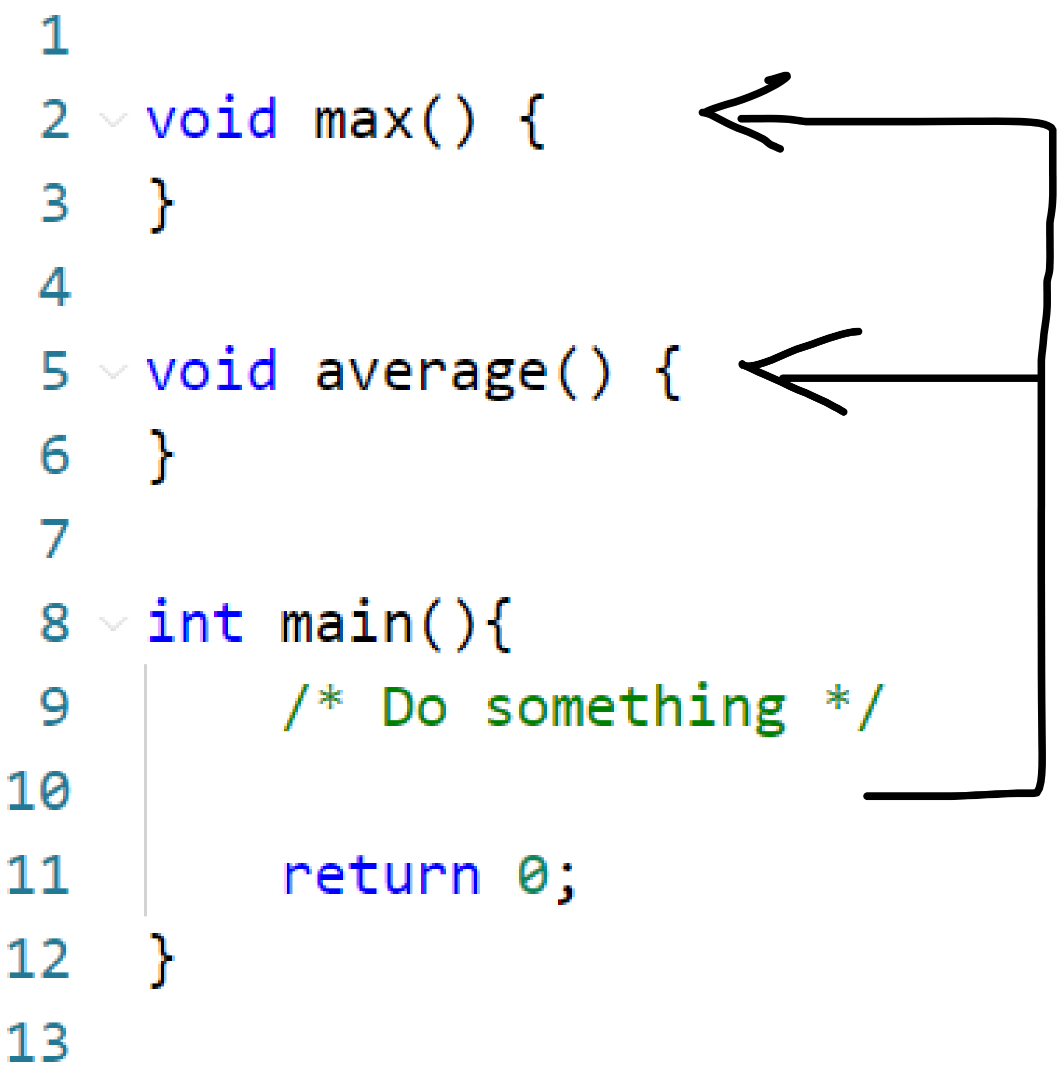
Other functions

DEFINING A FUNCTION



```
int my_function(int parameters) {
    /* Body */

    return /*something*/;
}
```

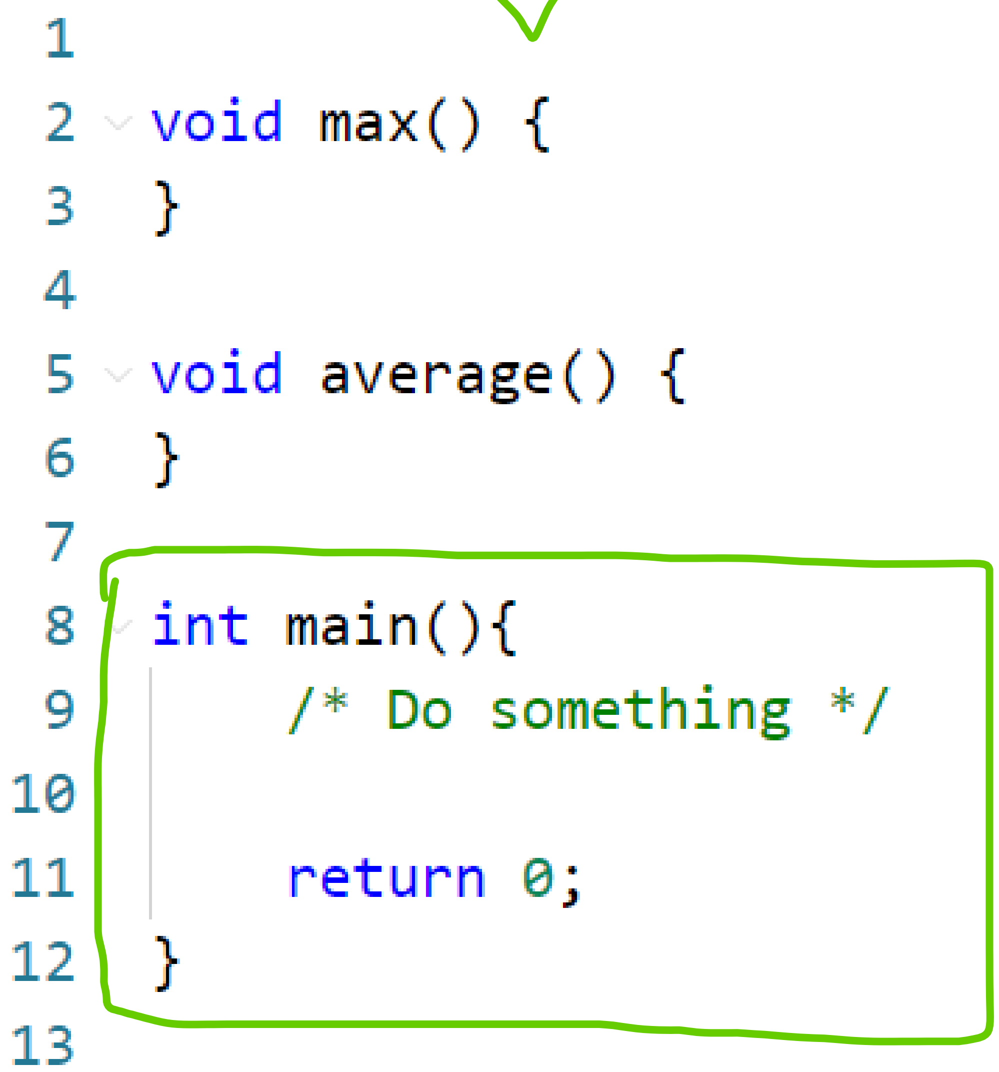
POSITION IN OUR CODE



```

void max() {
}

void average() {
}

int main(){
    /* Do something */

    return 0;
}

```

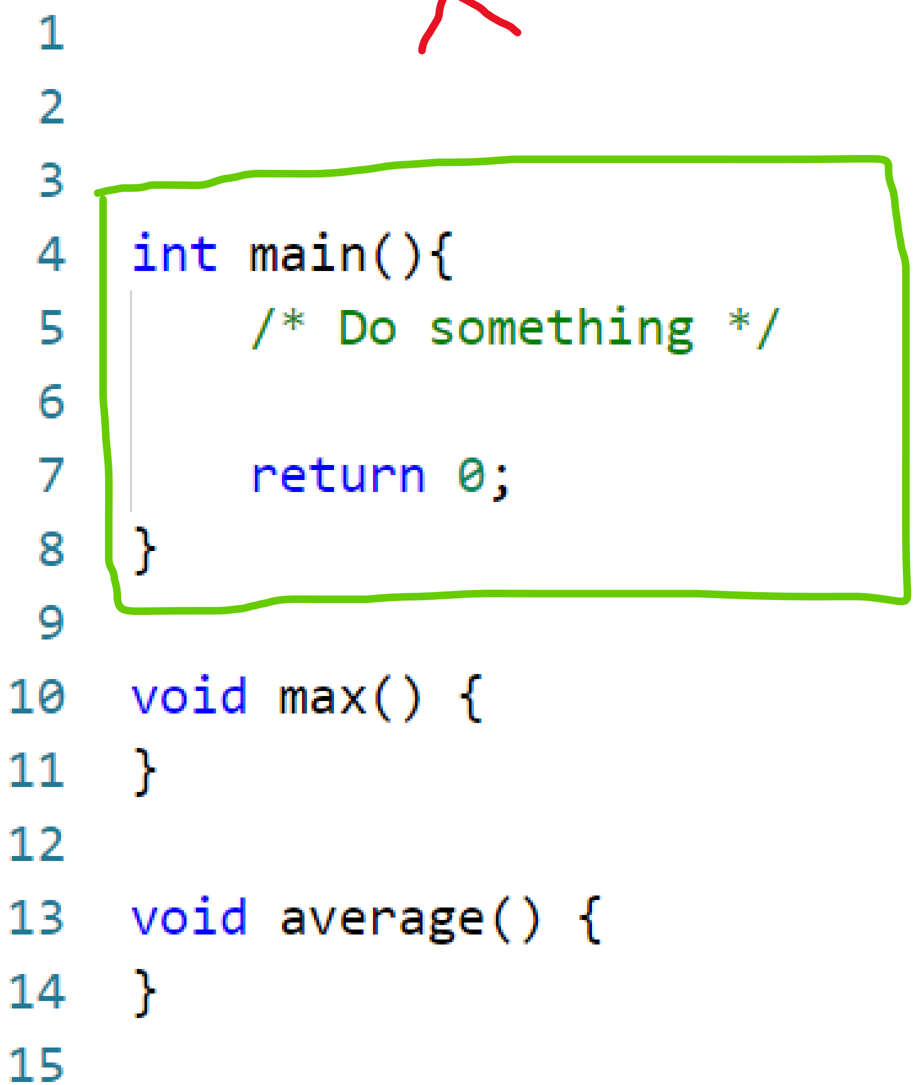
POSITION IN OUR CODE

✓



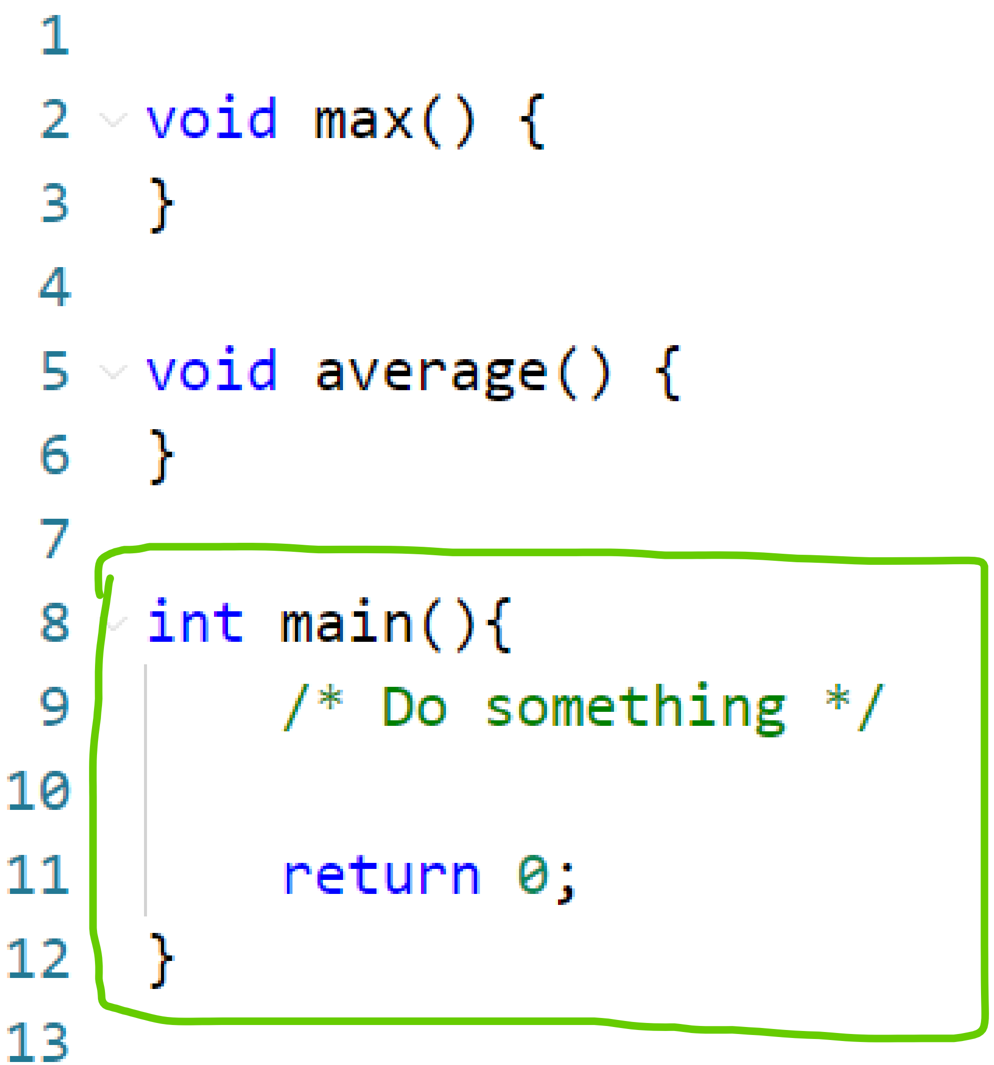
```

void max() {
}

void average() {
}

int main(){
    /* Do something */

    return 0;
}

```

✗



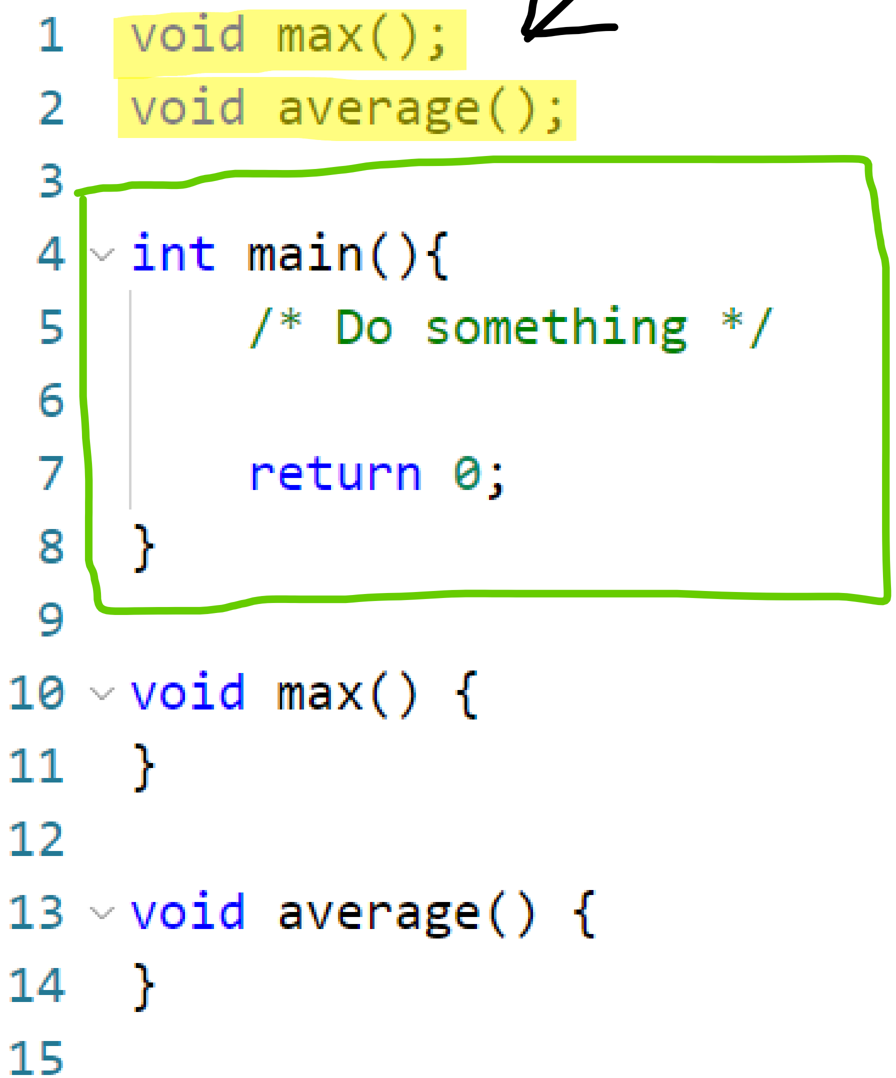
```


int main(){
    /* Do something */

    return 0;
}

void max() {
}

void average() {
}

```

POSITION IN OUR CODE



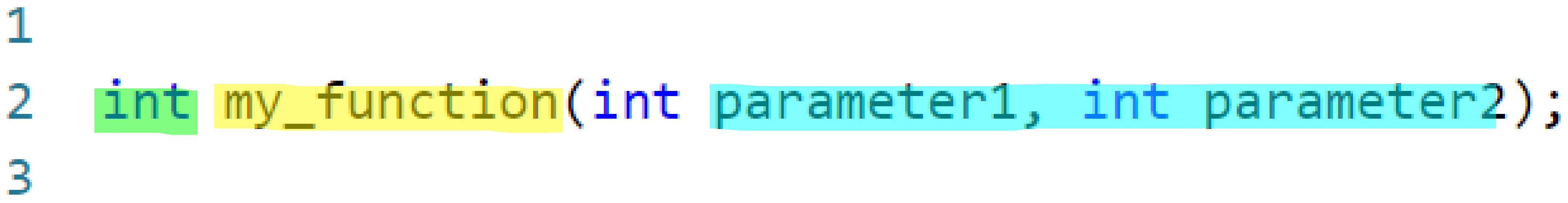
```

void max() {
}

void average() {
}

int main(){
    /* Do something */

    return 0;
}

```

Declaration

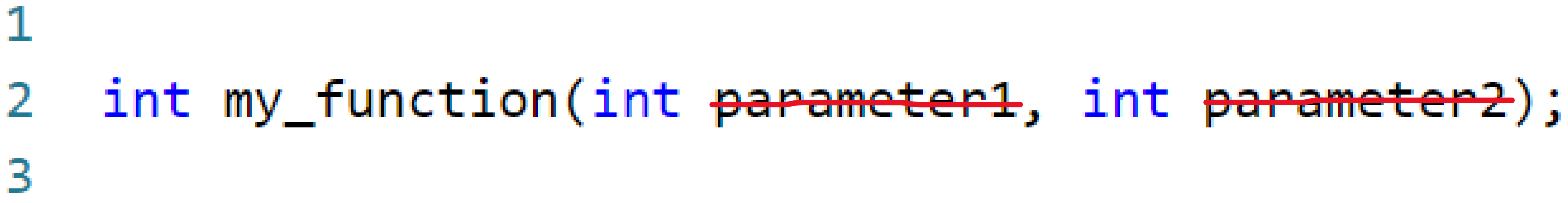


```
void max();
void average();

int main(){
    /* Do something */

    return 0;
}

void max() {
}

void average() {
}

```

FUNCTION DECLARATION

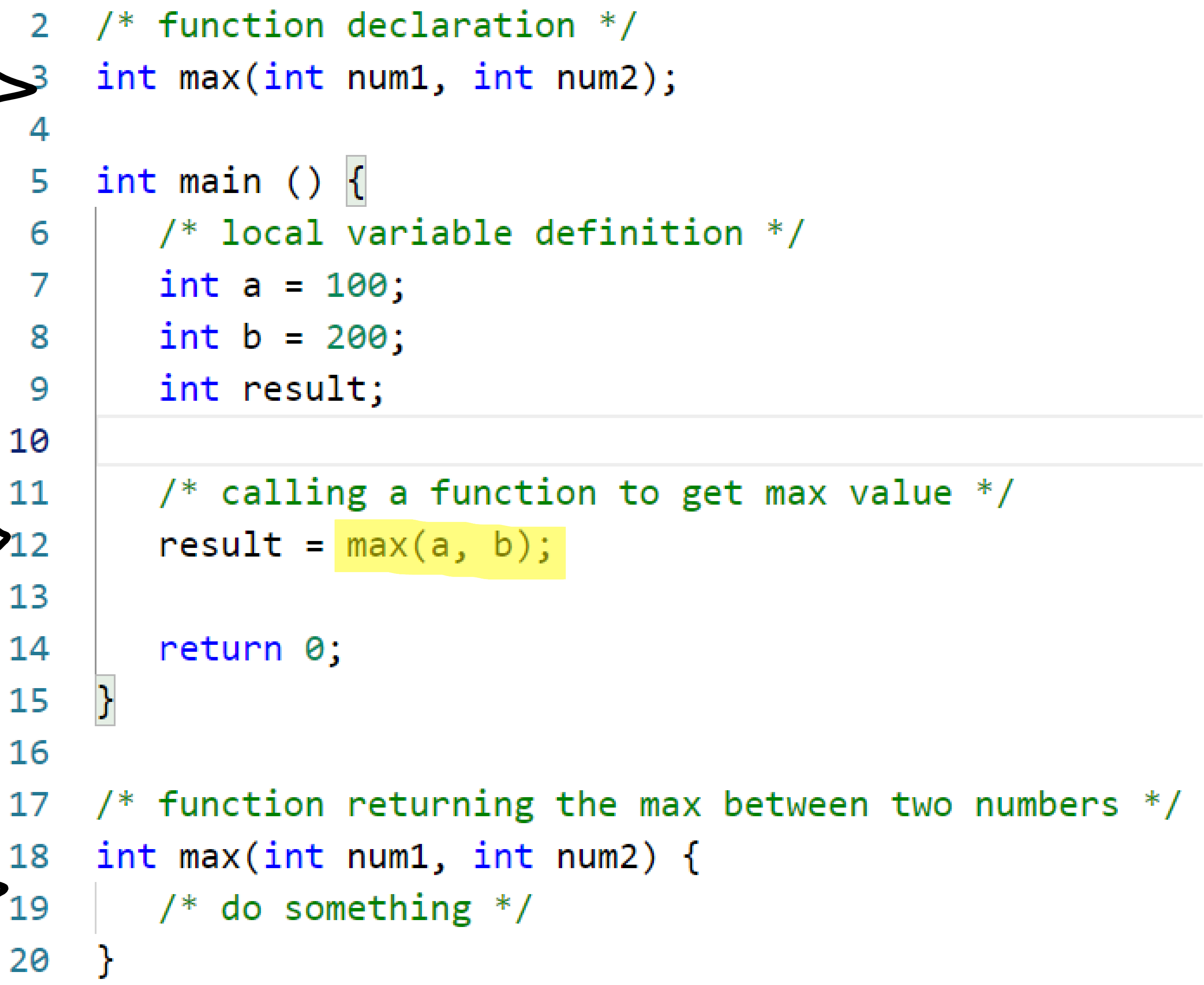


```
int my_function(int parameter1, int parameter2);
```



```
int my_function(int ~~parameter1~~, int ~~parameter2~~);
```

CALLING A FUNCTION



```c
/* function declaration */
int max(int num1, int num2);

int main () {
   /* local variable definition */
   int a = 100;
   int b = 200;
   int result;

   /* calling a function to get max value */
   result = max(a, b);

   return 0;
}

/* function returning the max between two numbers */
int max(int num1, int num2) {
   /* do something */
}
```

Declaration

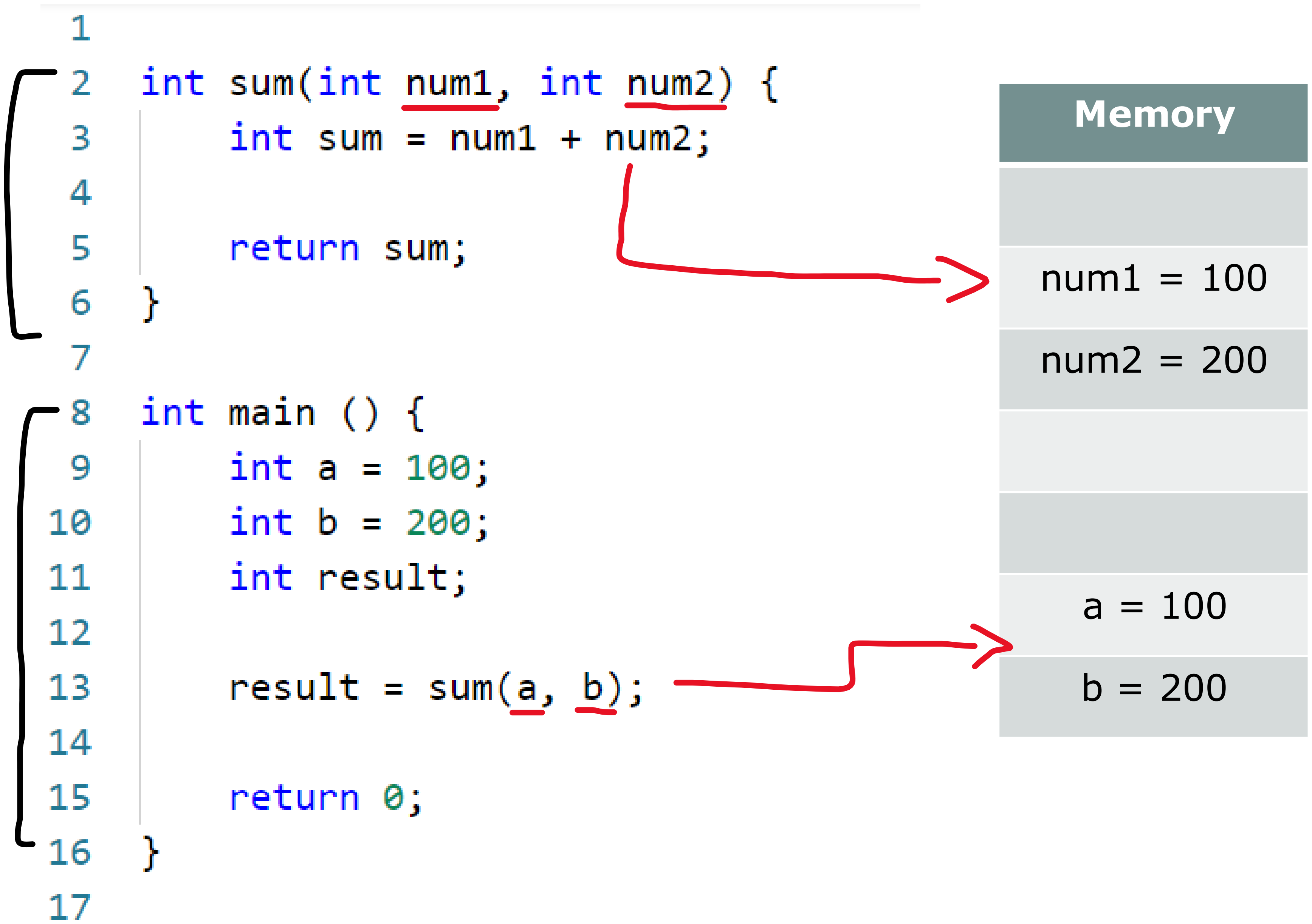
Call

Definition

FUNCTION ARGUMENTS BY VALUE

The scope of num1, num2 and sum is within the sum function.

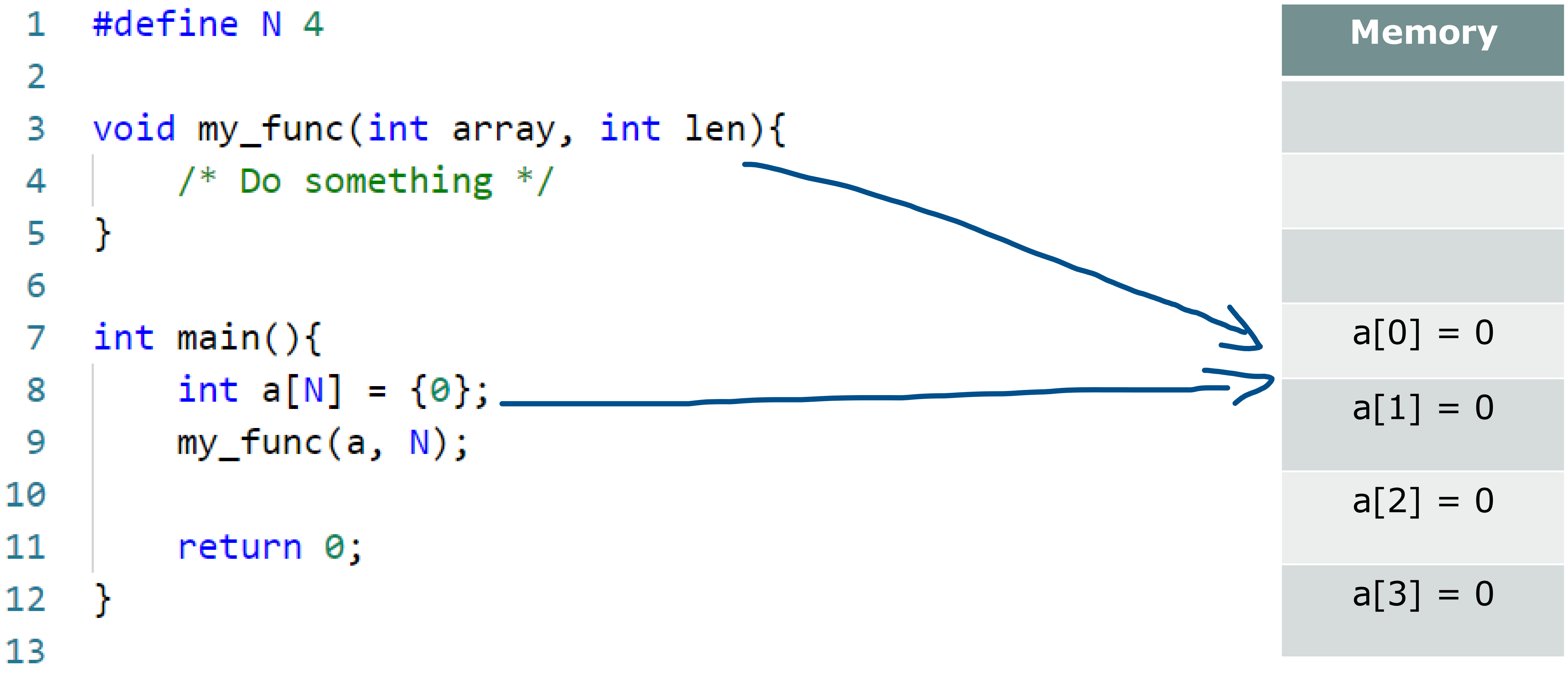
The scope of a, b and result is within the Main function.



```c

int sum(int num1, int num2) {
    int sum = num1 + num2;

    return sum;
}

int main () {
    int a = 100;
    int b = 200;
    int result;

    result = sum(a, b);

    return 0;
}

```

Memory
num1 = 100
num2 = 200
a = 100
b = 200

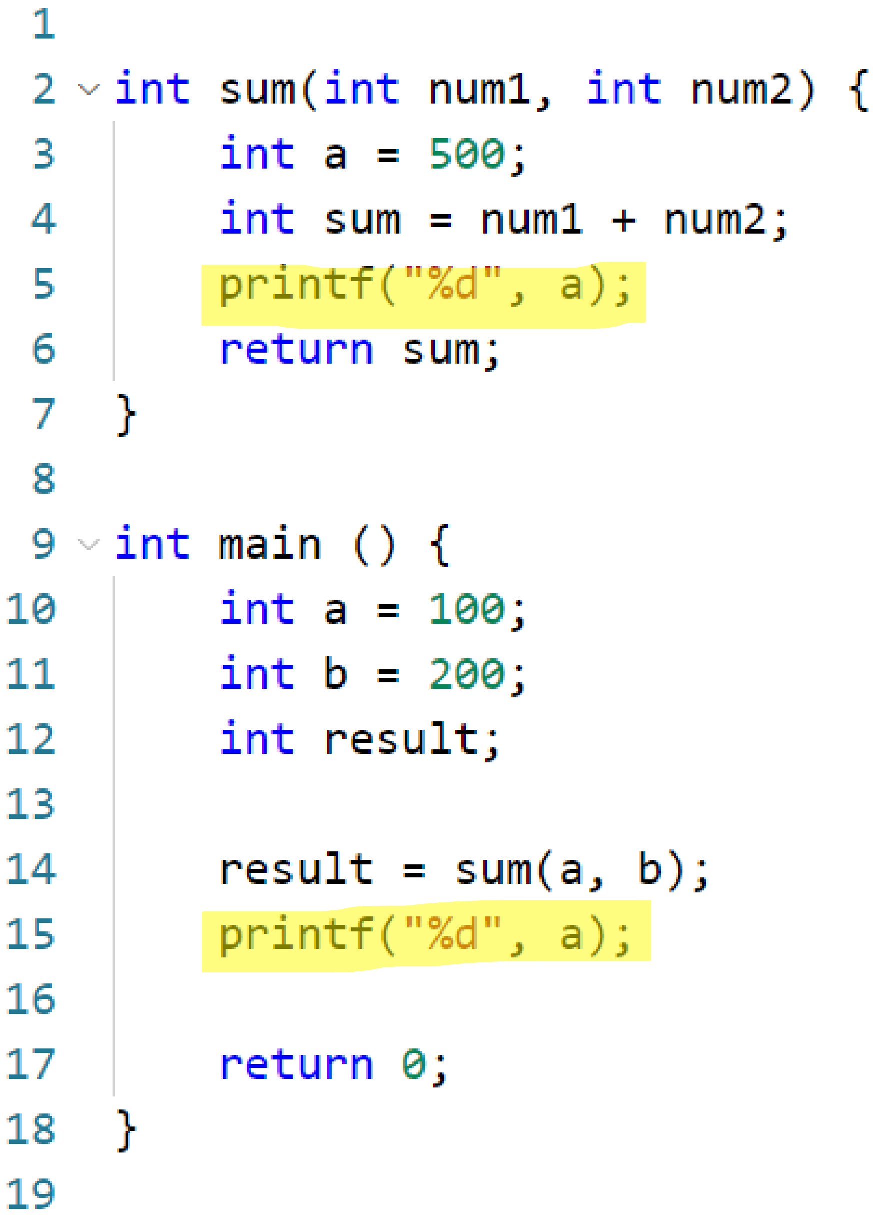
FUNCTION ARGUMENTS BY REFERENCE



```c
#define N 4

void my_func(int array, int len){
    /* Do something */
}

int main(){
    int a[N] = {0};
    my_func(a, N);

    return 0;
}

```

Memory
a[0] = 0
a[1] = 0
a[2] = 0
a[3] = 0

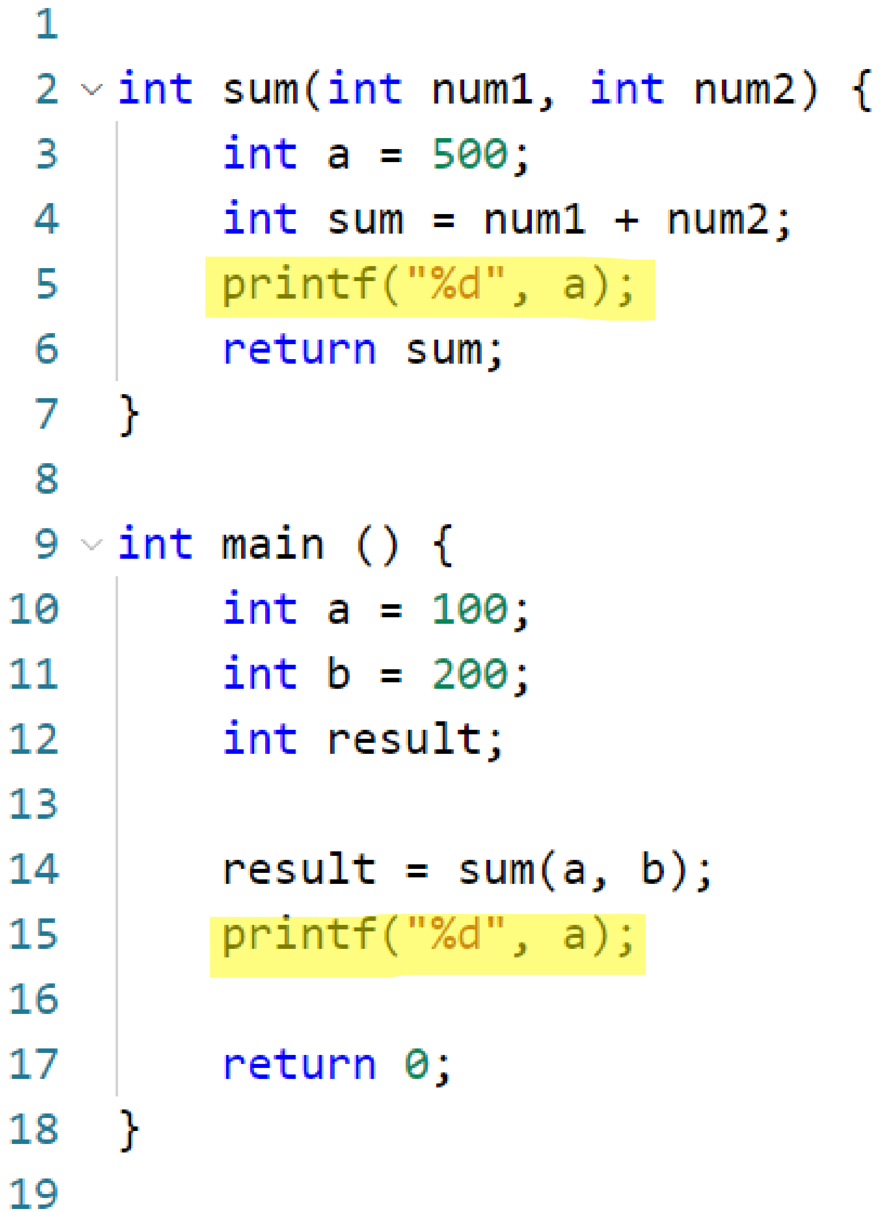
QUESTION



```c

int sum(int num1, int num2) {
    int a = 500;
    int sum = num1 + num2;
    printf("%d", a);
    return sum;
}

int main () {
    int a = 100;
    int b = 200;
    int result;

    result = sum(a, b);
    printf("%d", a);

    return 0;
}

```

Output ???

QUESTION



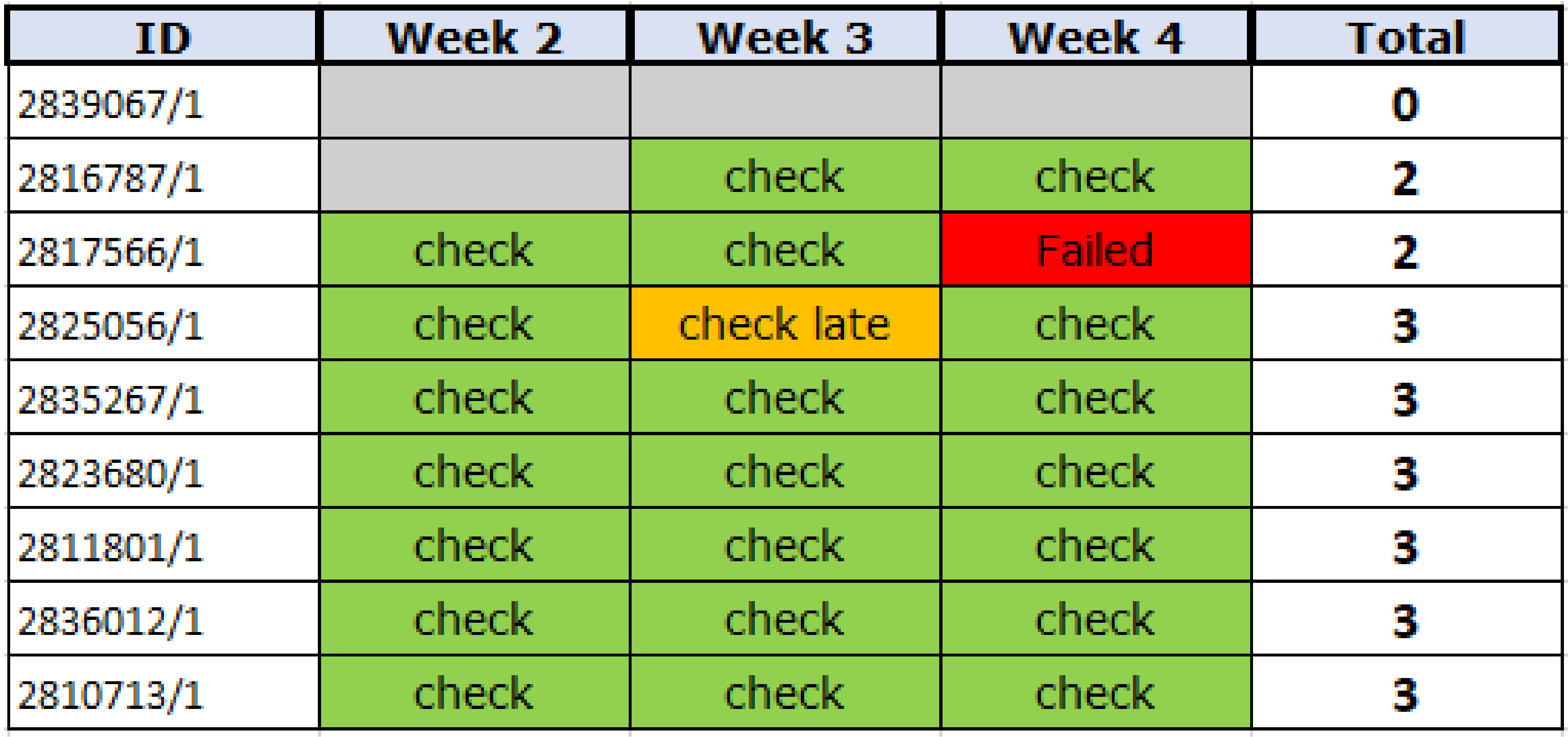
```c

int sum(int num1, int num2) {
    int a = 500;
    int sum = num1 + num2;
    printf("%d", a);
    return sum;
}

int main () {
    int a = 100;
    int b = 200;
    int result;

    result = sum(a, b);
    printf("%d", a);

    return 0;
}

```

Output ???

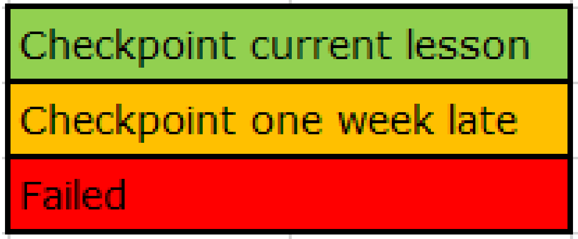
a = 500

a = 100

CHECKPOINTS GROUP A

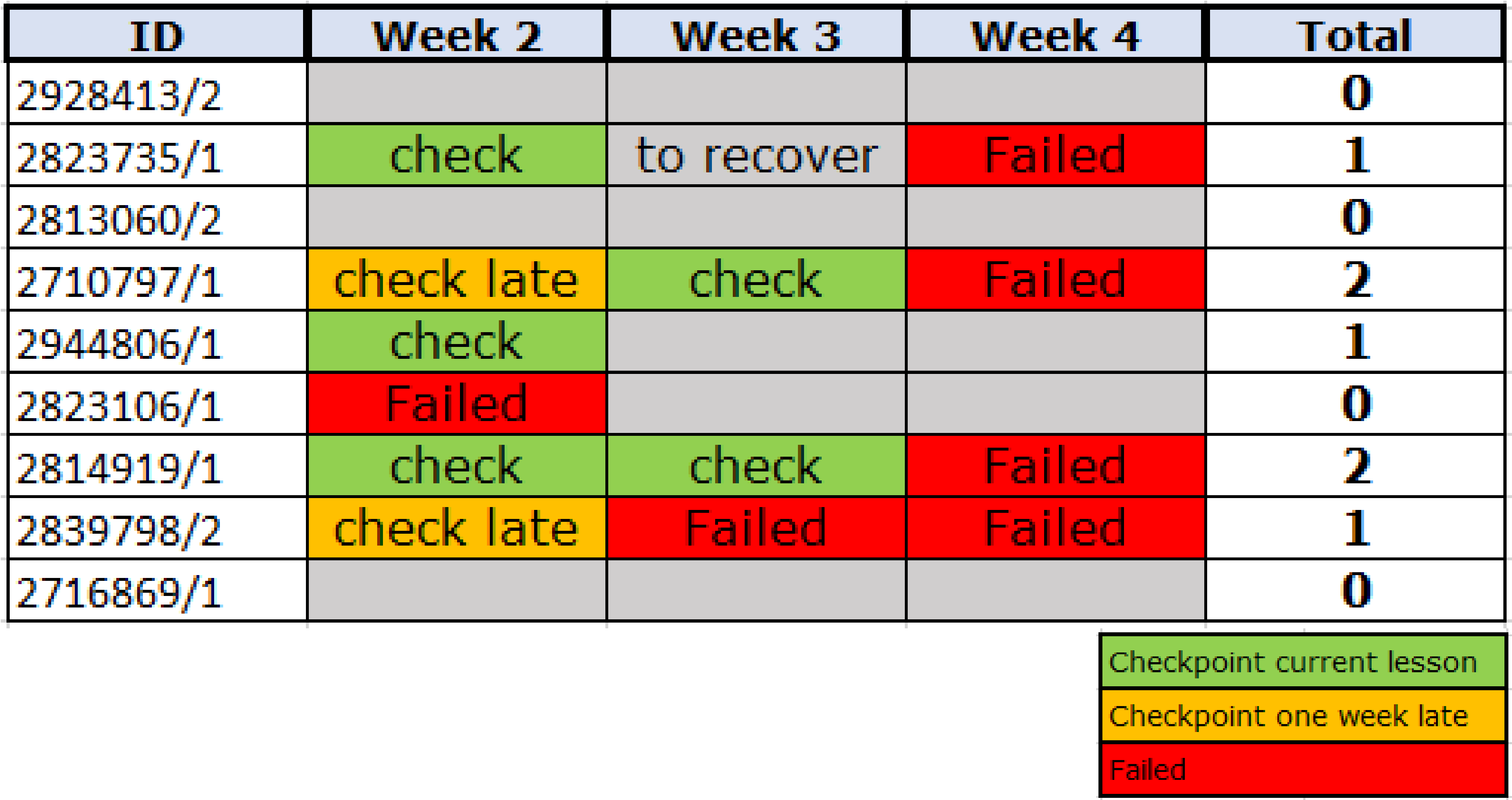


ID	Week 2	Week 3	Week 4	Total
2839067/1				0
2816787/1		check	check	2
2817566/1	check	check	Failed	2
2825056/1	check	check late	check	3
2835267/1	check	check	check	3
2823680/1	check	check	check	3
2811801/1	check	check	check	3
2836012/1	check	check	check	3
2810713/1	check	check	check	3



Checkpoint current lesson
Checkpoint one week late
Failed

CHECKPOINTS GROUP B



ID	Week 2	Week 3	Week 4	Total
2928413/2				0
2823735/1	check	to recover	Failed	1
2813060/2				0
2710797/1	check late	check	Failed	2
2944806/1	check			1
2823106/1	Failed			0
2814919/1	check	check	Failed	2
2839798/2	check late	Failed	Failed	1
2716869/1				0

🟩 Checkpoint current lesson

🟨 Checkpoint one week late

🟥 Failed

Any Questions?

Thank you