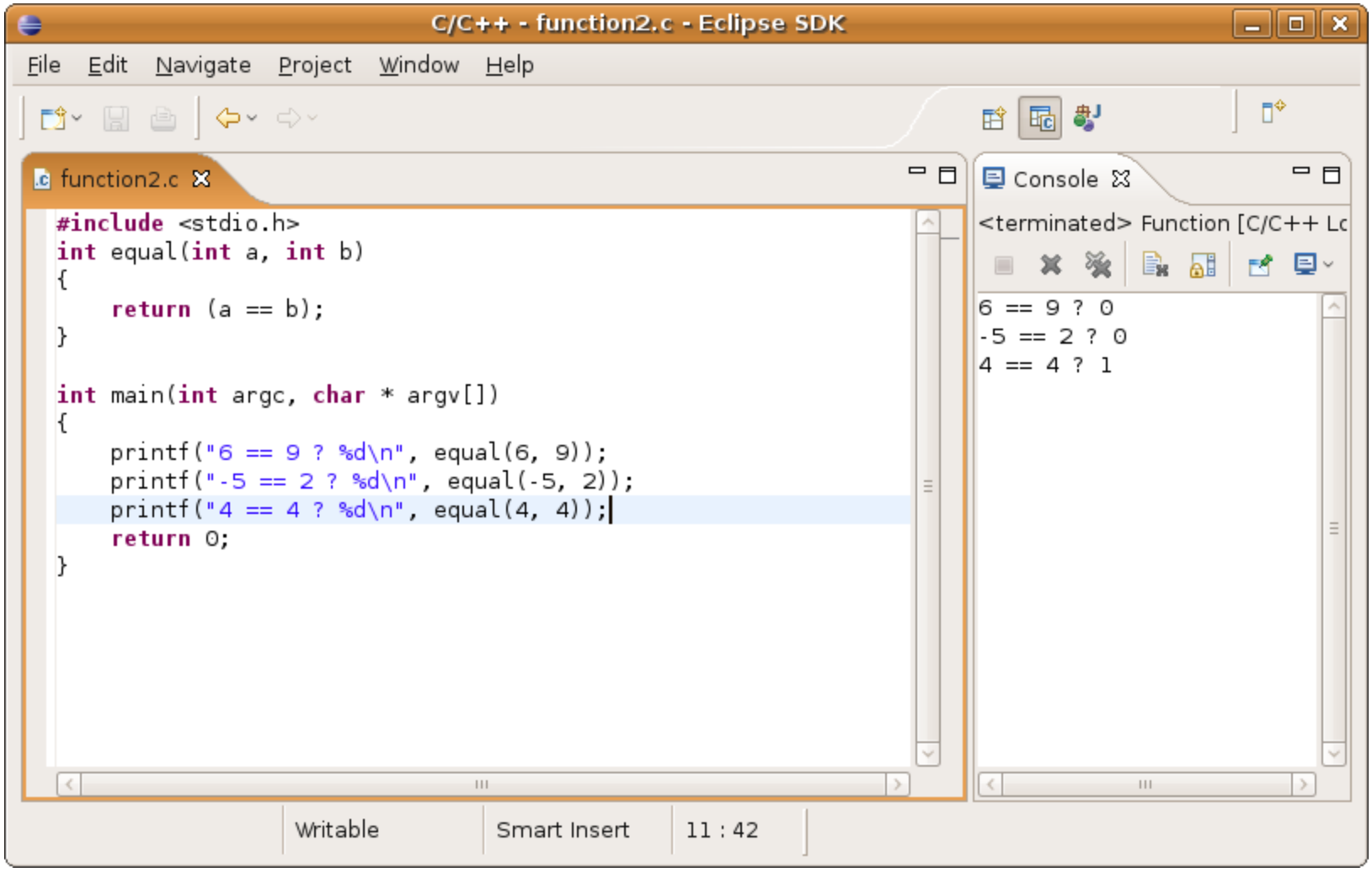
Functions

Yung-Hsiang Lu

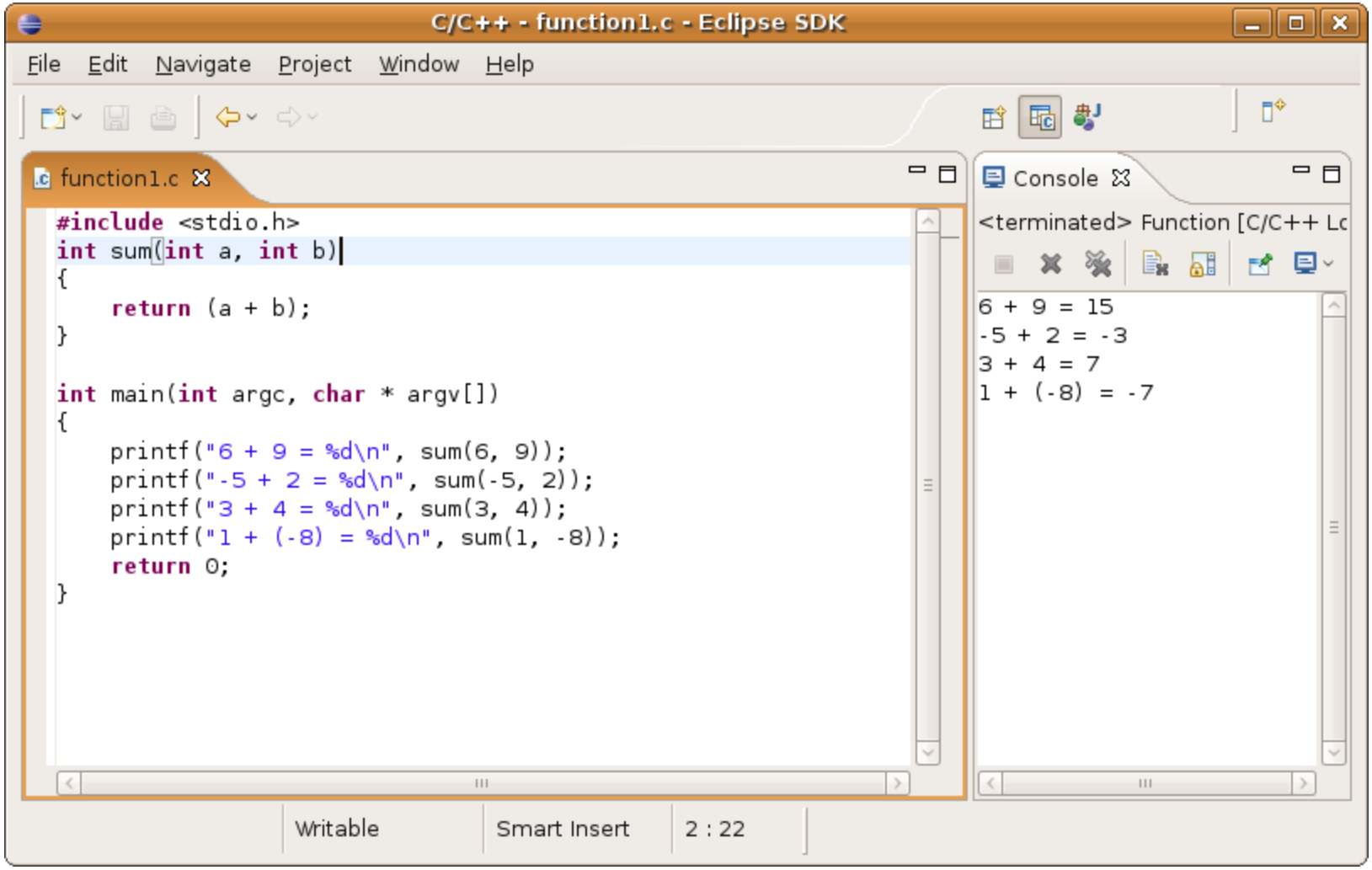
Functions in C Programs

- Functions are fundamental building blocks.
- A C program starts at the “main” function.
- Functions are essential for **code reuse**.
- Code reuse: perform similar tasks, differences controlled by the input arguments. Do **not** copy-paste code in multiple places in a program.
- We have used some functions
  - printf
  - scanf
  - strtol

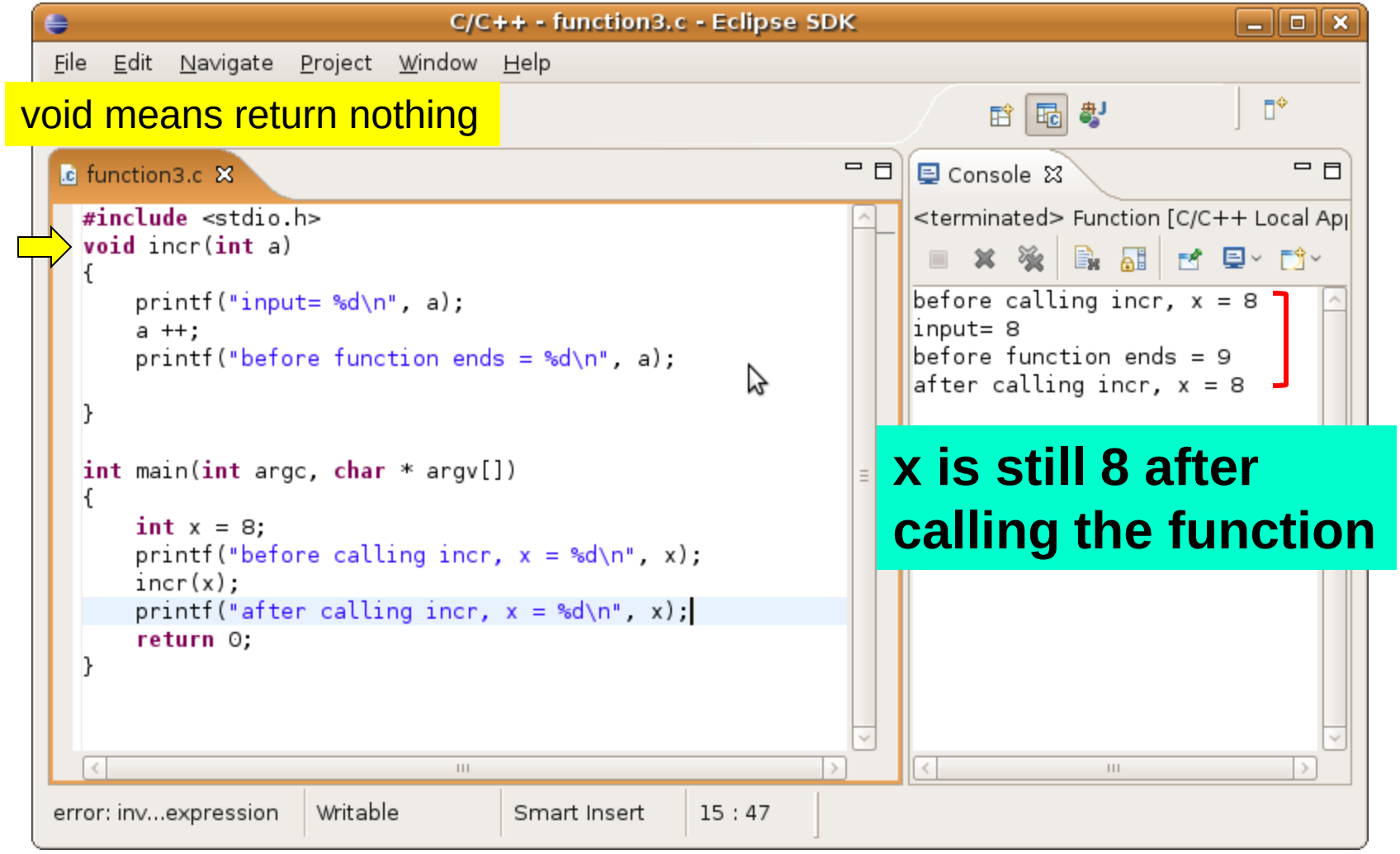
Compare Two Arguments



Function Return Value



Modify the Argument



Call by Value

A function **copies** the **value** to the function's argument.

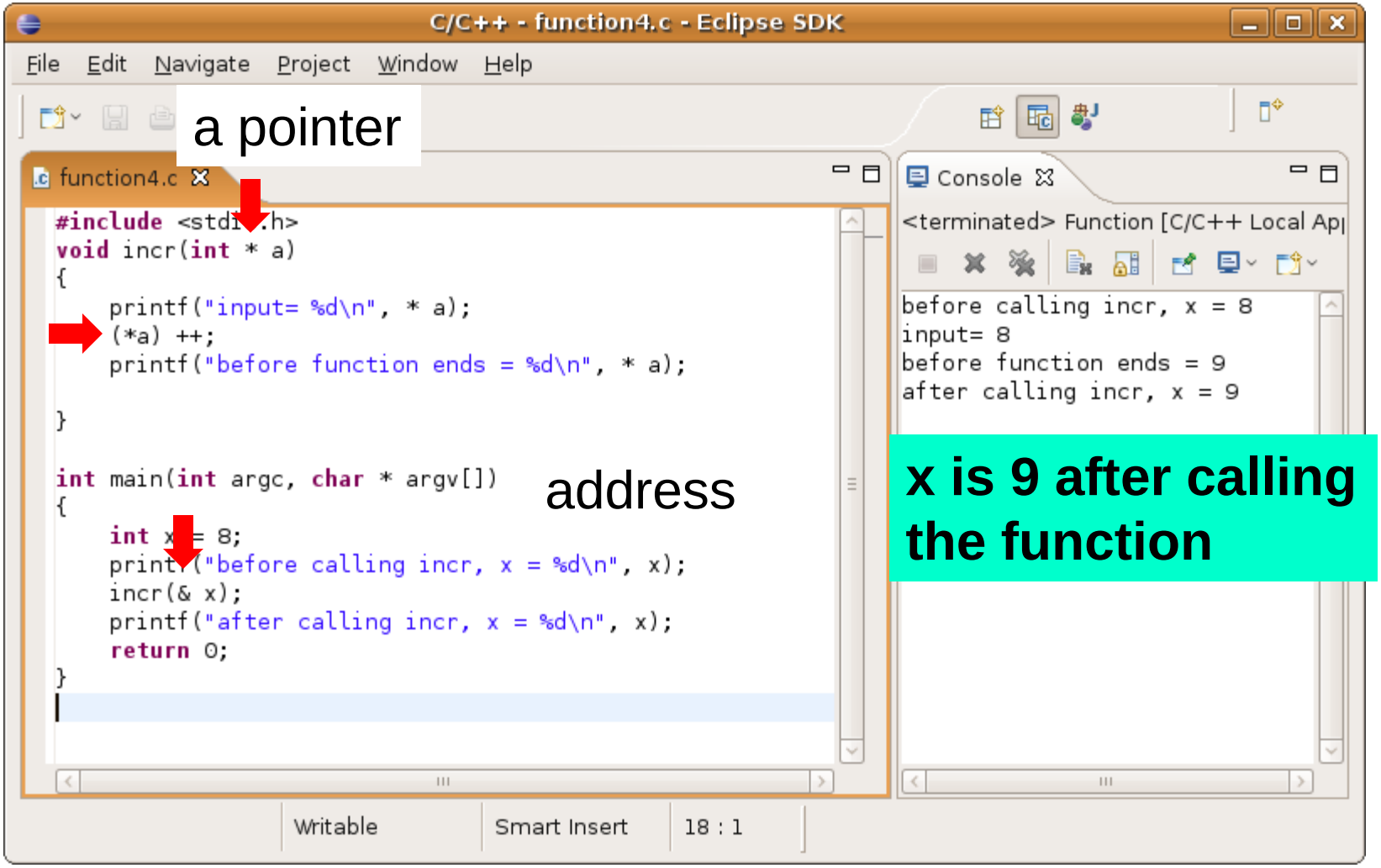
```
void incr(int a) …

int x = 8;
incr(x);        /* copy x's value to a */
a ++;           /* a becomes 9, x is still 8 */
```

similar to

```
int x = 8;
int y = x;
y ++;           /* y is 9, x is still 8 */
```

Address	**Data**
somewhere (&x)	8
somewhere (&a)	8 → 9

Address as Argument



Call by Address

A function **copies** the **address** to the argument.

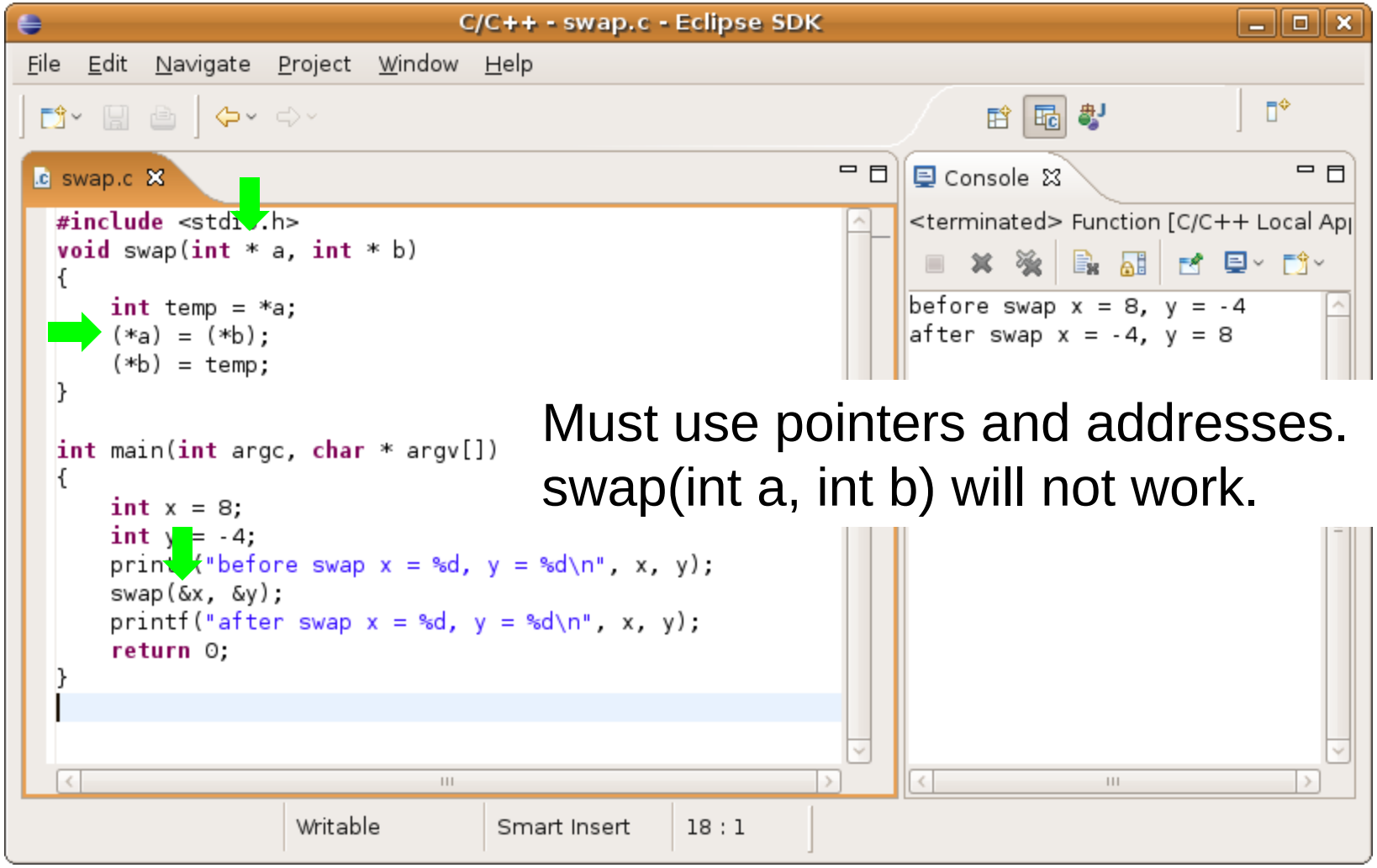
```
void incr(int * a) …

int x = 8;
incr(& x);      /* copy x's value to a */
(*a) ++;                /* a becomes 9, so is x */
```

similar to

```
int x = 8;
int * y = & x;
(*y) ++;        /* x is 9 */
```

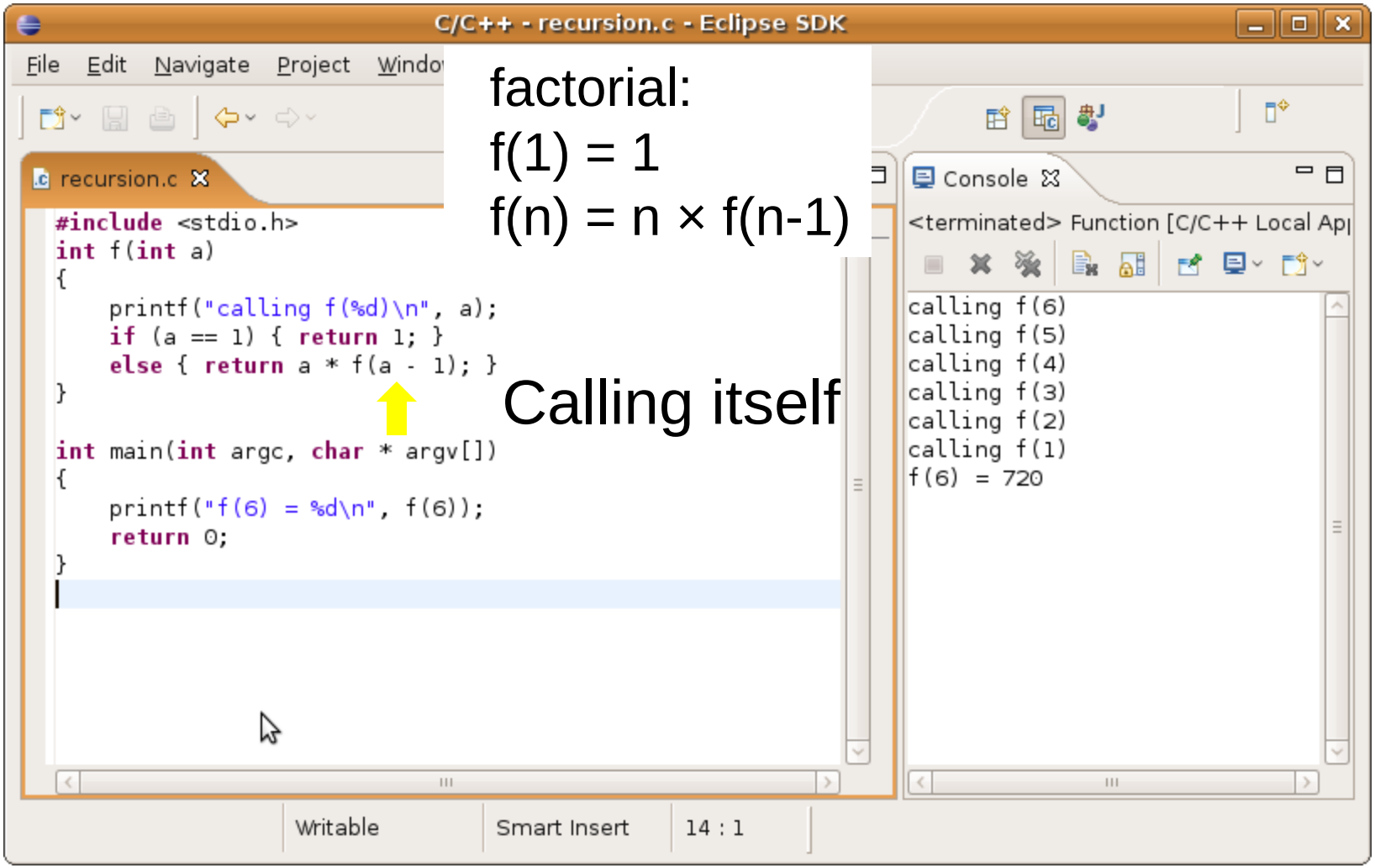
Address	**Data**
somewhere (&x)	8
somewhere (&a)	& x

Swap Function



Must use pointers and addresses.
swap(int a, int b) will not work.

Recursion (Function calling Itself)



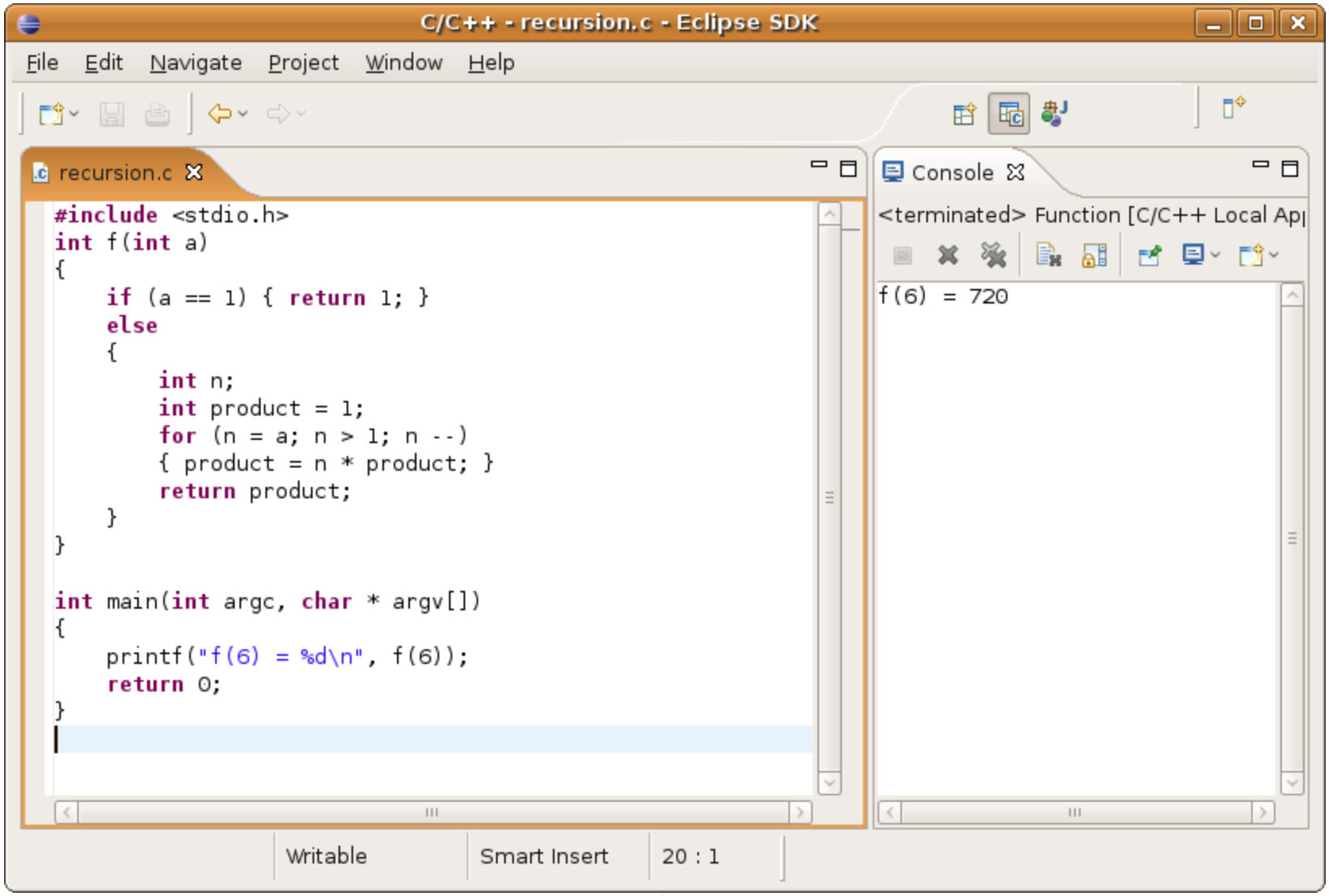
Structure of Recursive Calls

- function with arguments
  1. If the terminal condition is satisfied, solve the problem and return the result
  2. Otherwise, divide the problem into smaller parts and solve individual parts by calling the function with new arguments

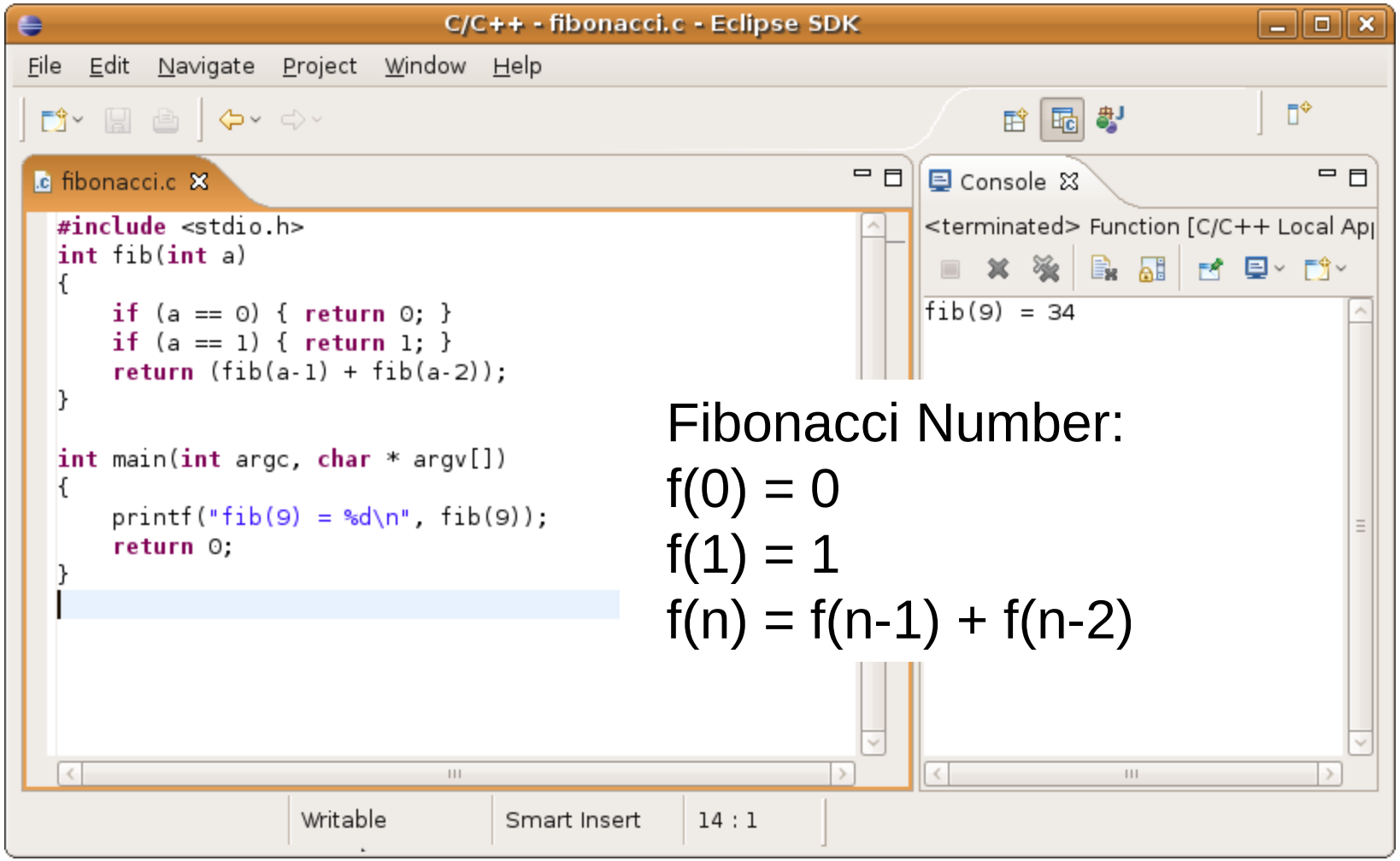
```
if (a == 1) { return 1; }          /* step 1 */
else { return a * f(a-1); }        /* step 2 */
```

- This is an example of a problem-solving strategy called **divide and conquer**.

Recursion ⇒ Iteration



Fibonacci Number



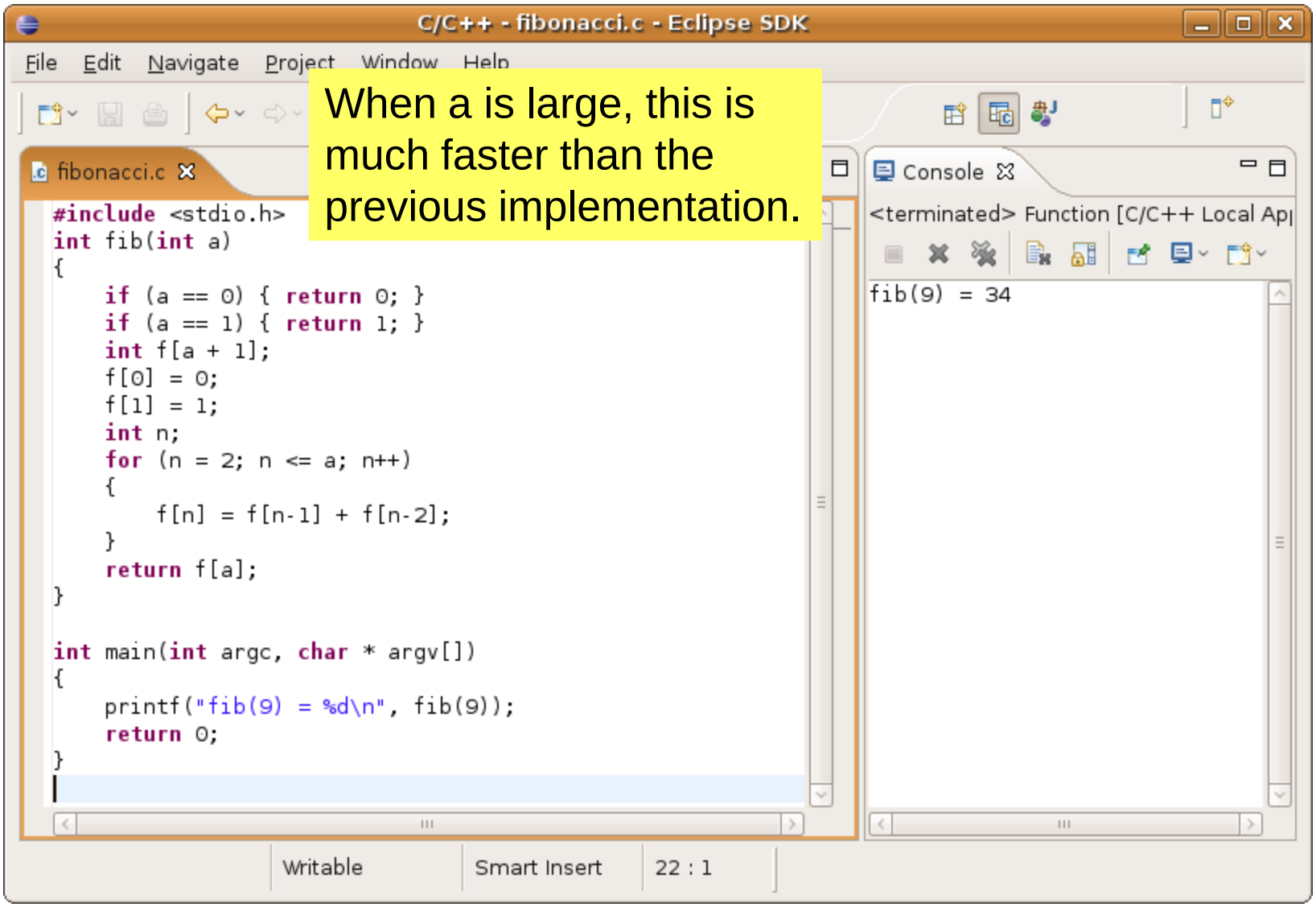
Fibonacci Number:

f(0) = 0

f(1) = 1

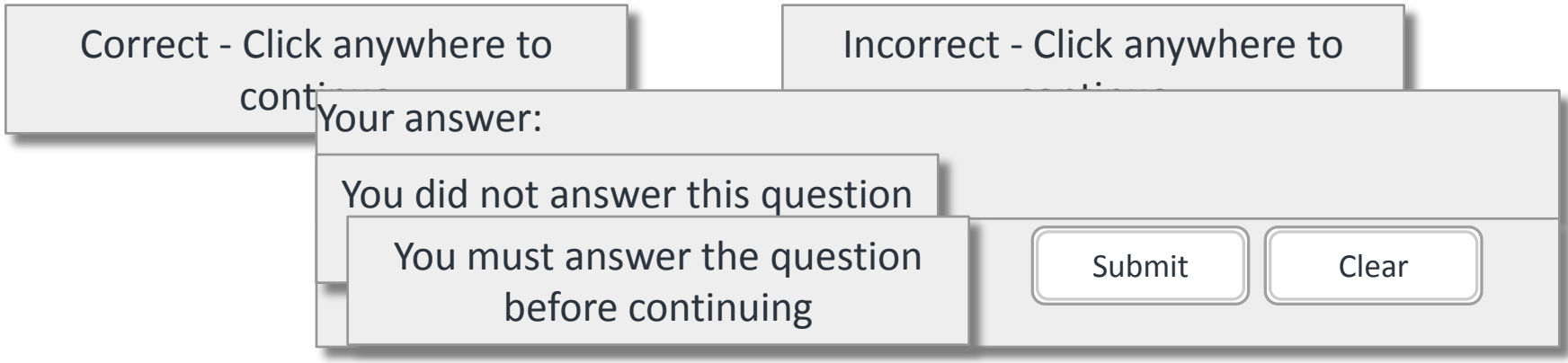
f(n) = f(n-1) + f(n-2)

Fibonacci Number by Iteration



Which statement is correct? You may choose multiple answers.

- [x] A) A function may have zero, one, two, or more arguments.
- [x] B) A function may return a value.
- [x] C) A function may call itself.
- [x] D) A function may call another function.



What is the value of x after calling the function?

```
void f(int a)
 {
        a -= 2;
}
int x = 7;
f(x);
```

x's value is

Correct - Click anywhere to cont

Incorrect - Click anywhere to

Your answer:

You did not answer this question

You must answer the question before continuing

Submit

Clear

Function

Your Score	{score}
Max Score	{max-score}
Number of Quiz Attempts	{total-attempts}

Question Feedback/Review Information Will Appear Here

Continue | Review Quiz