This is CS50

Week 1

Scan your HUID for attendance at the back table.
Open `code.cs50.io` and log in!

Carter Zenke

Preceptor

carter@cs50.harvard.edu

Think, Pair, Share

- What are you excited about from this week's lecture?
- What do you want to learn more about?

https://carterzenke.me/section

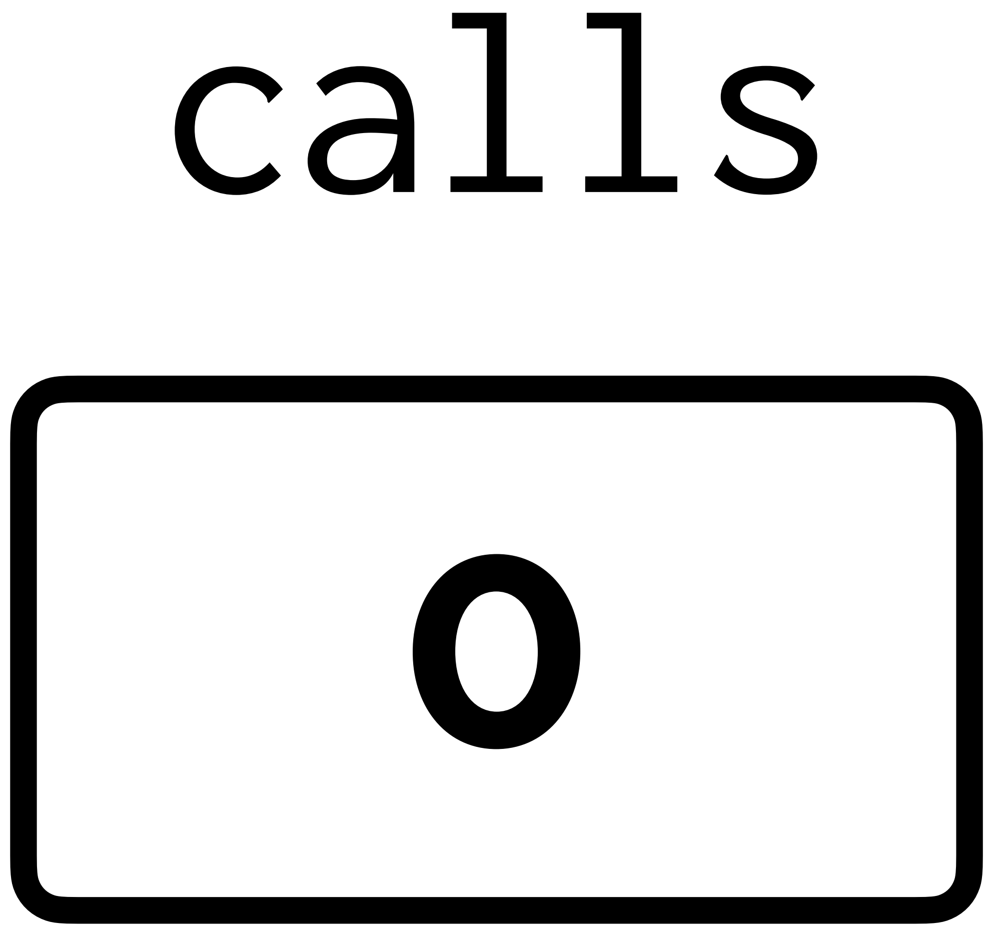
Today

- Variables and Types
- Input and Printing
- Functions, Loops, and Conditionals
- Problem Set 1

Variables and Types

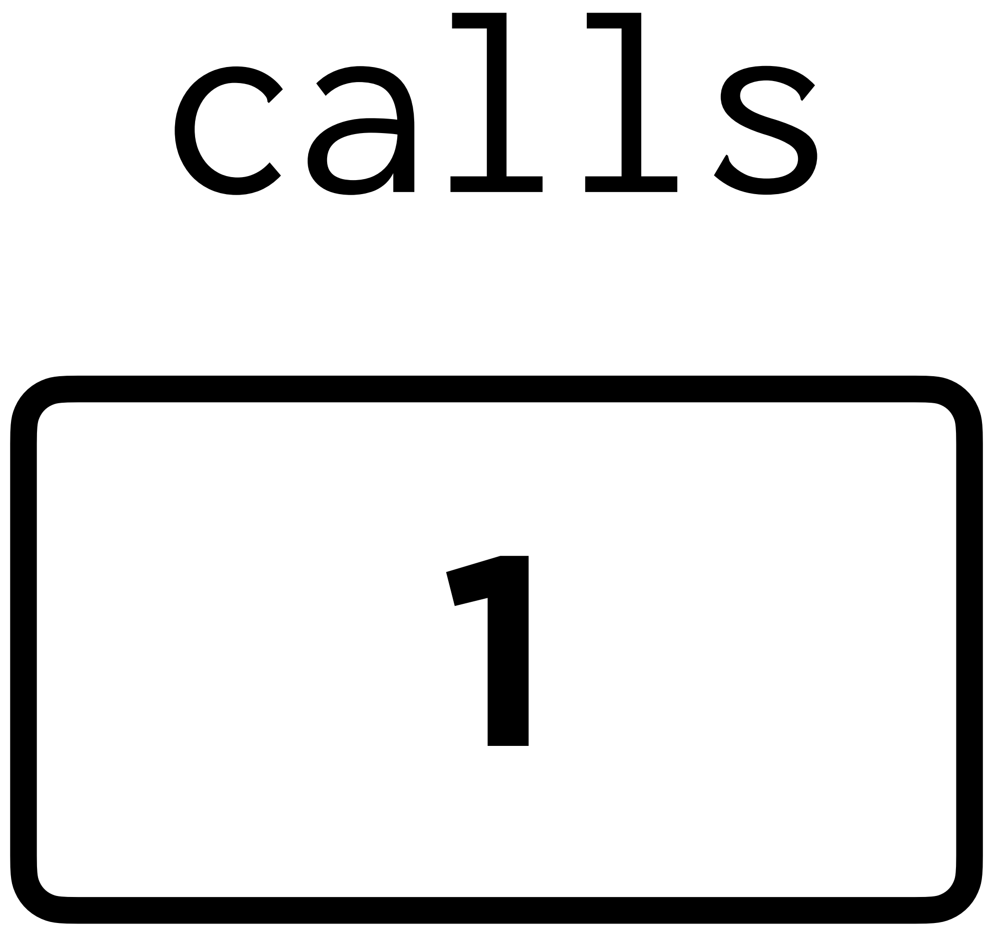
How would you explain what a **variable** is, in a single sentence?

0



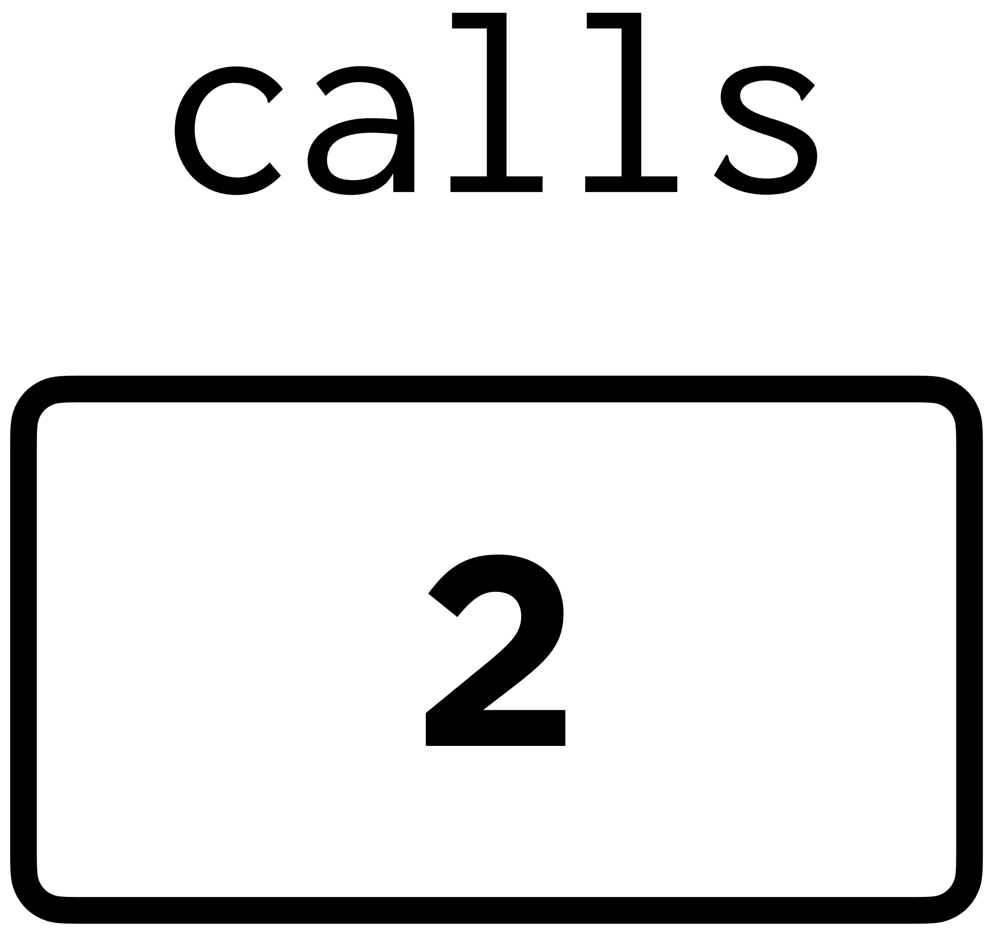
calls

0



calls

1

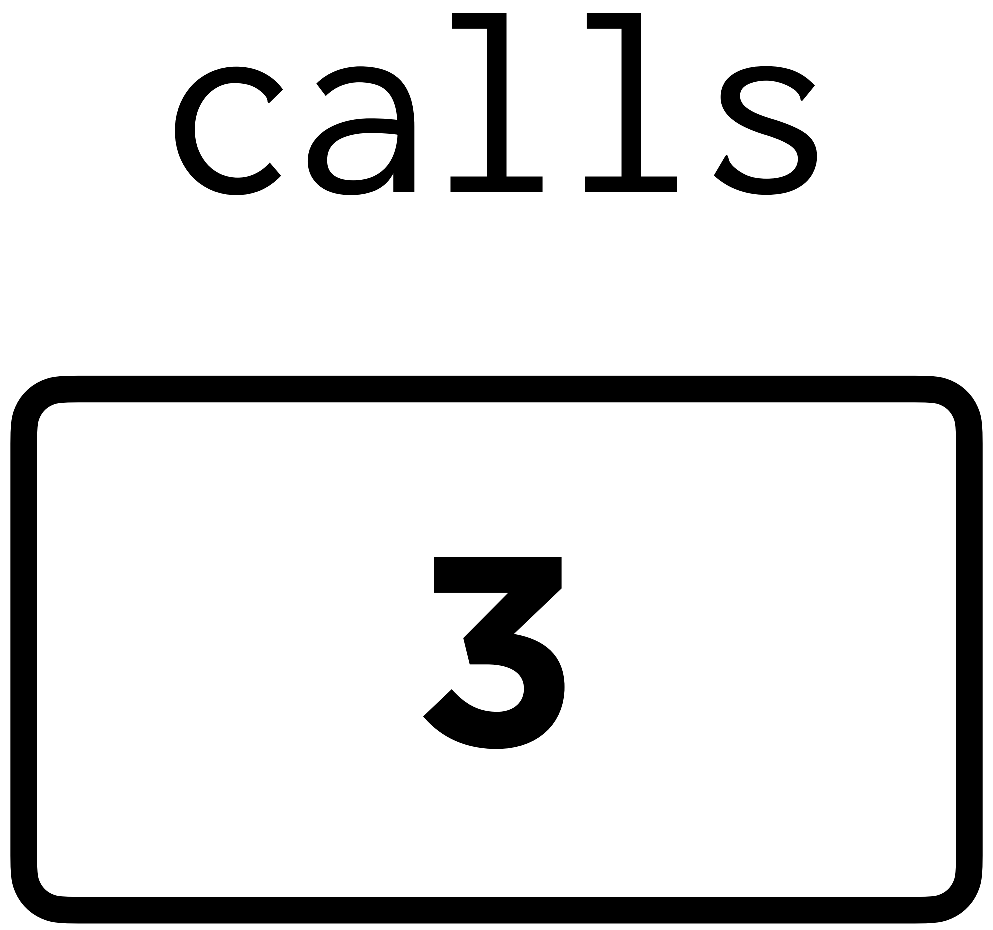


calls

2

calls

3

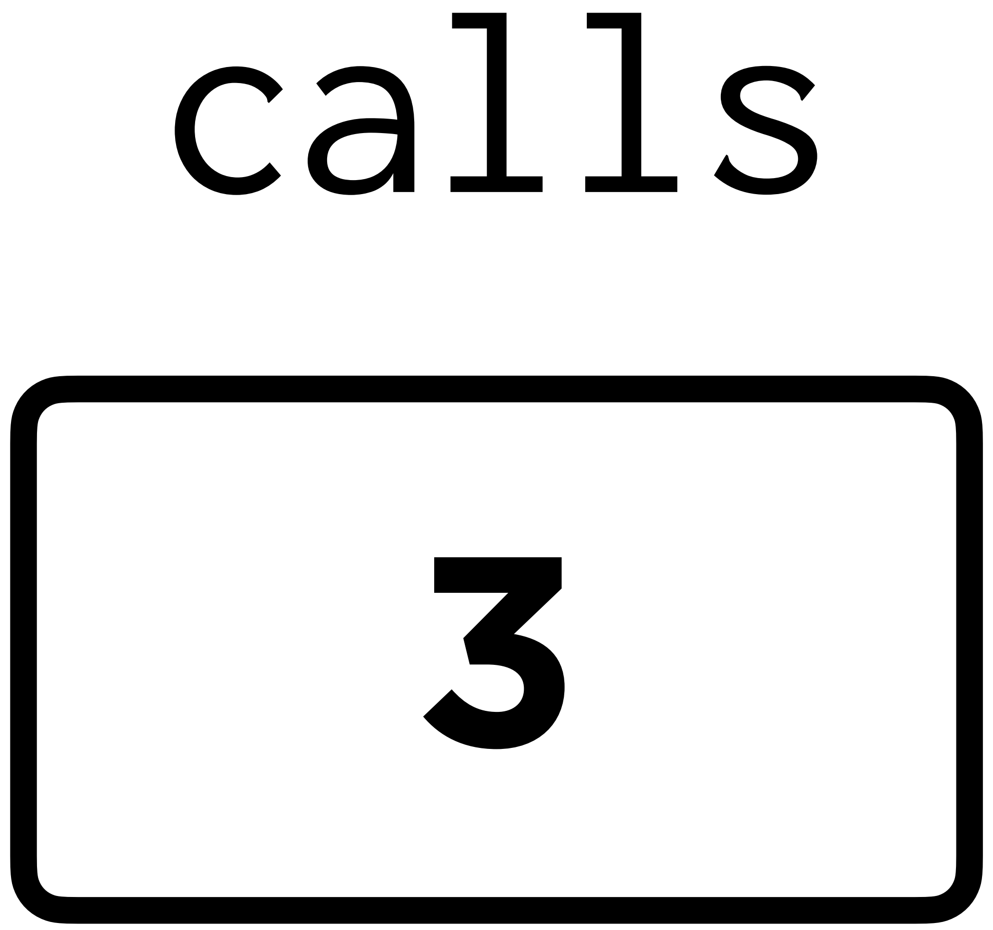


`calls`

3

A **variable** is a name for some value that can change.

```
int calls = 3;
```

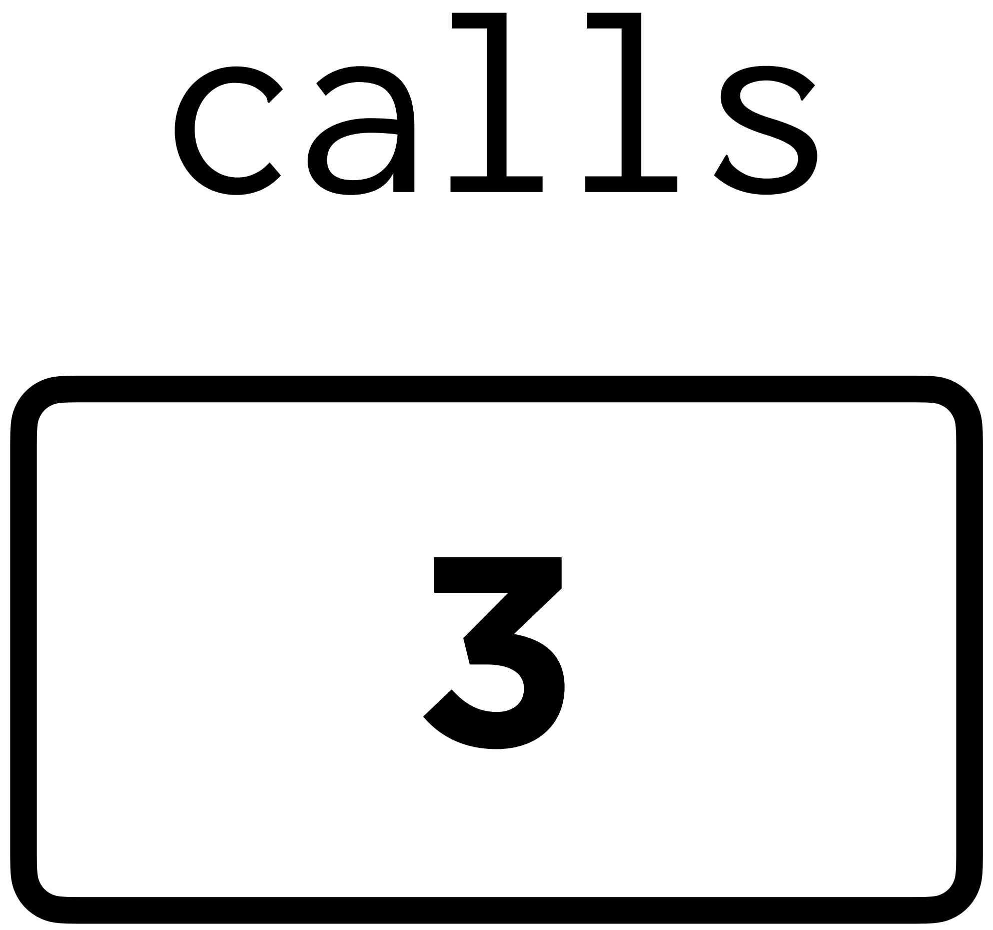


calls

3

```
int calls = 3;
```

name

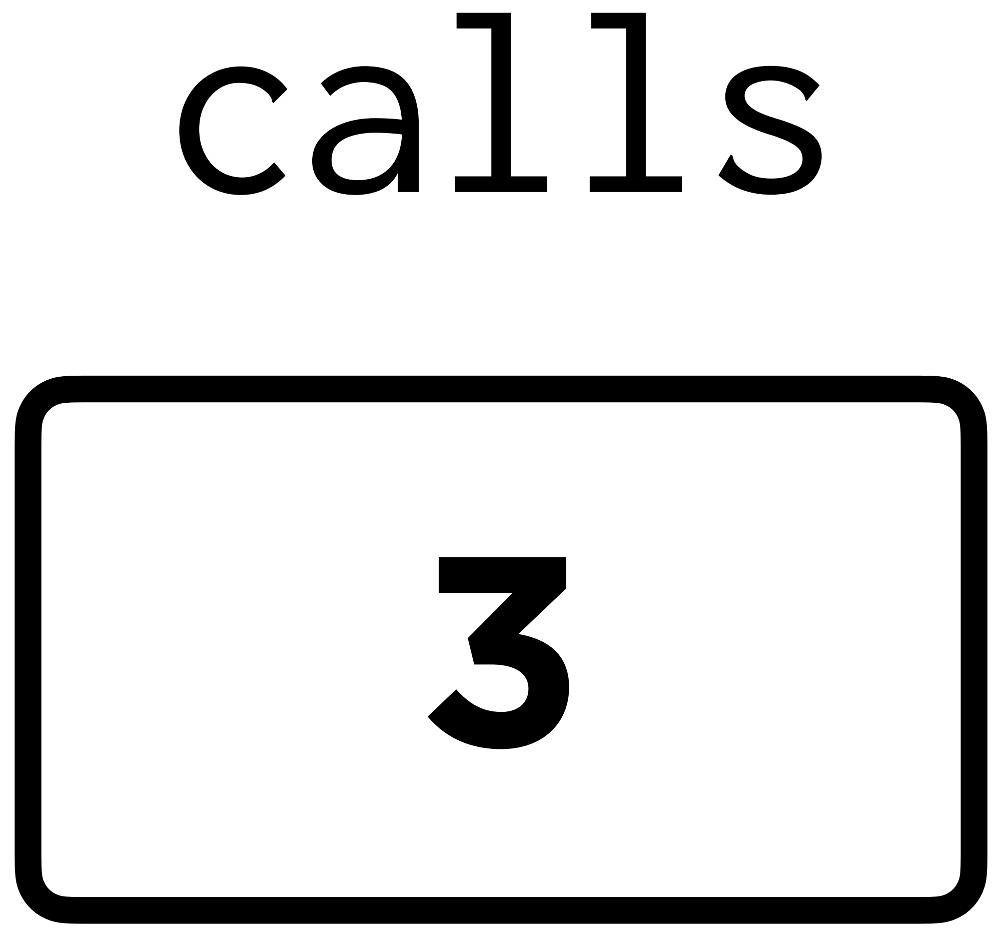


calls

3

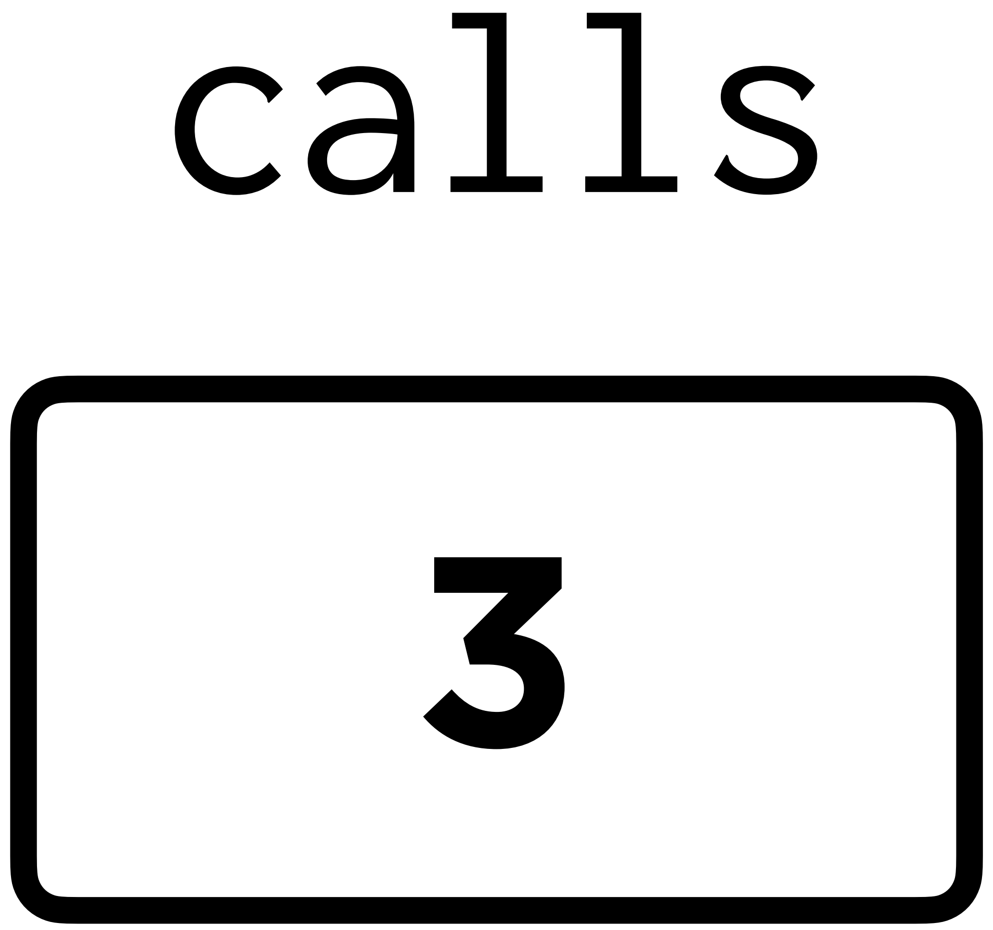
```
int calls = 3;
```

type



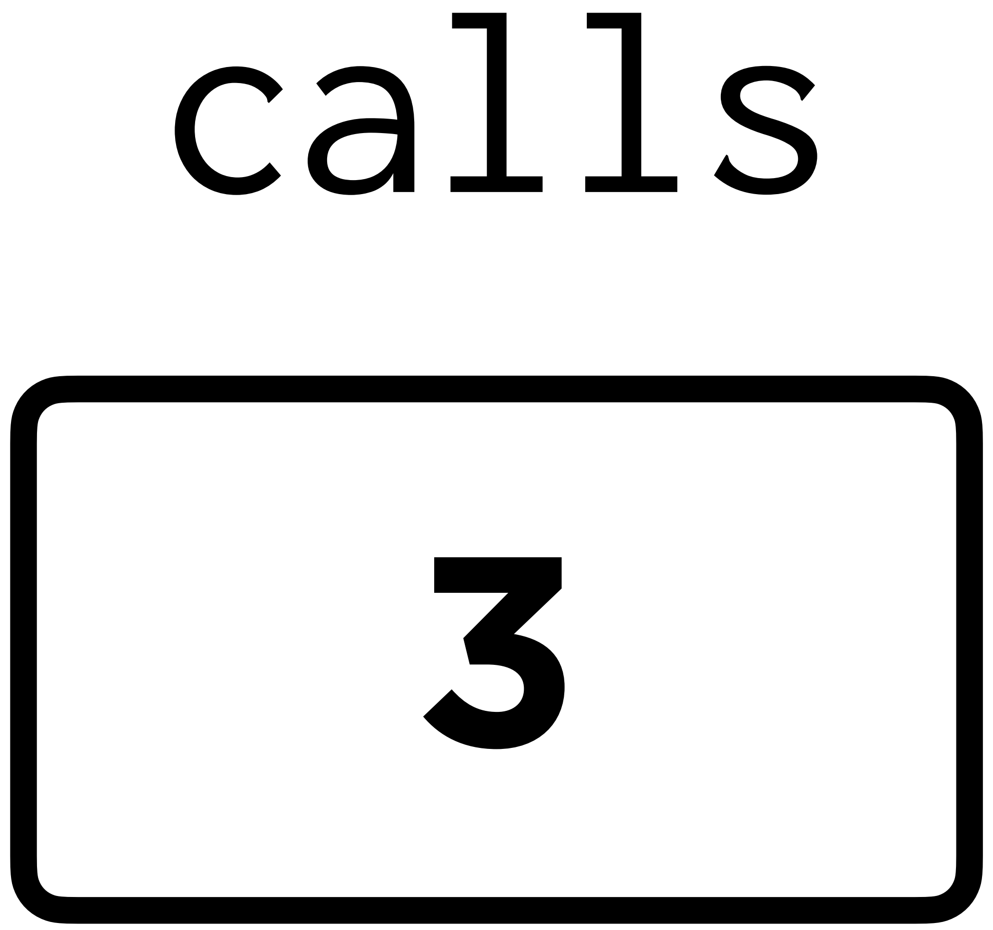
```
int calls = 3;
```

value



```
int calls = 3;
```

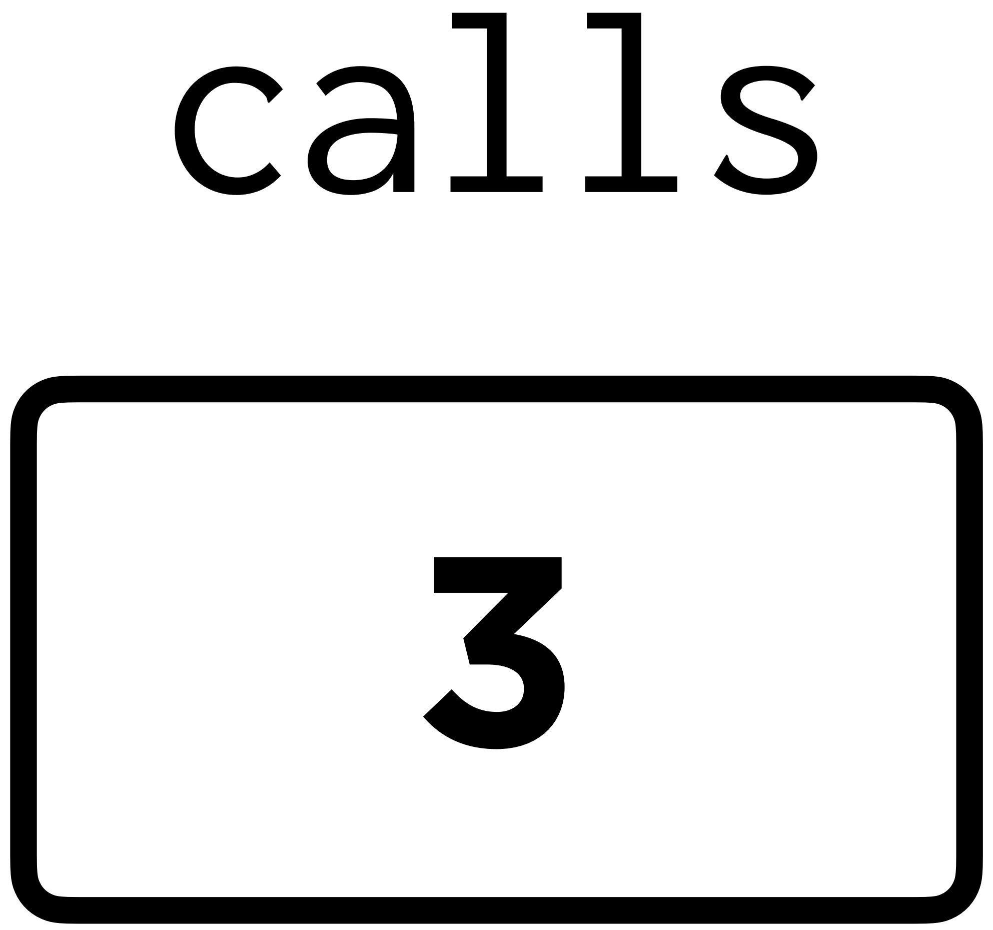
assignment operator



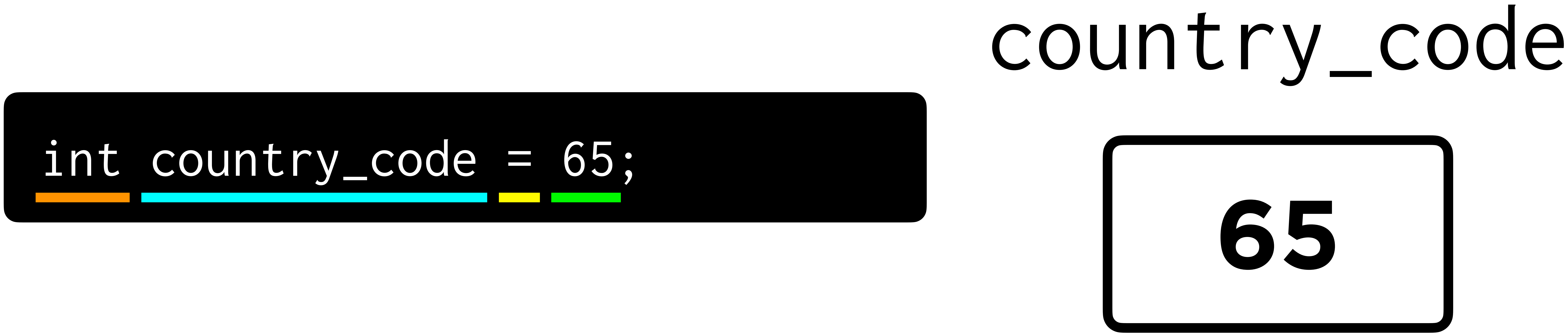
```
int calls = 3;
```

type name | value

assignment operator



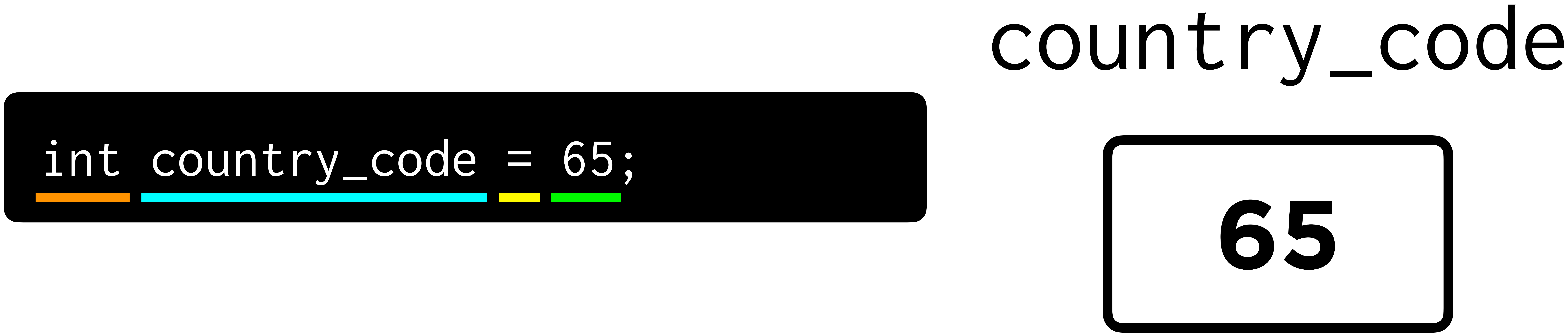
"Create an ***integer*** *named* ***calls*** *that* ***gets*** *the value* ***3****."*



```
int country_code = 65;
```

country_code

65

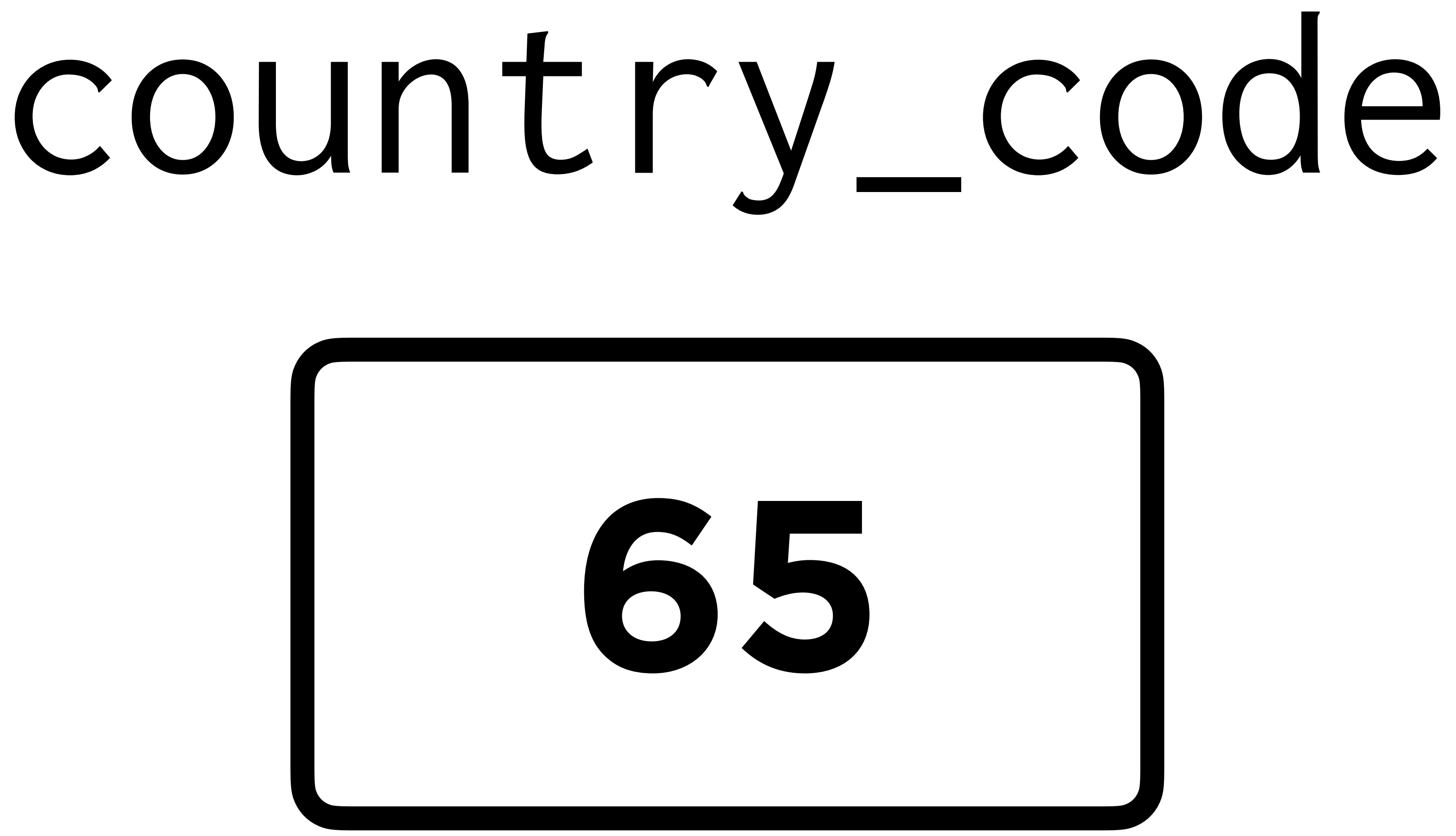


```
int country_code = 65;
```

"Create an ***integer*** *named* ***country_code*** *that* ***gets*** *the value* ***65****."*

```
int country_code = 65;
```

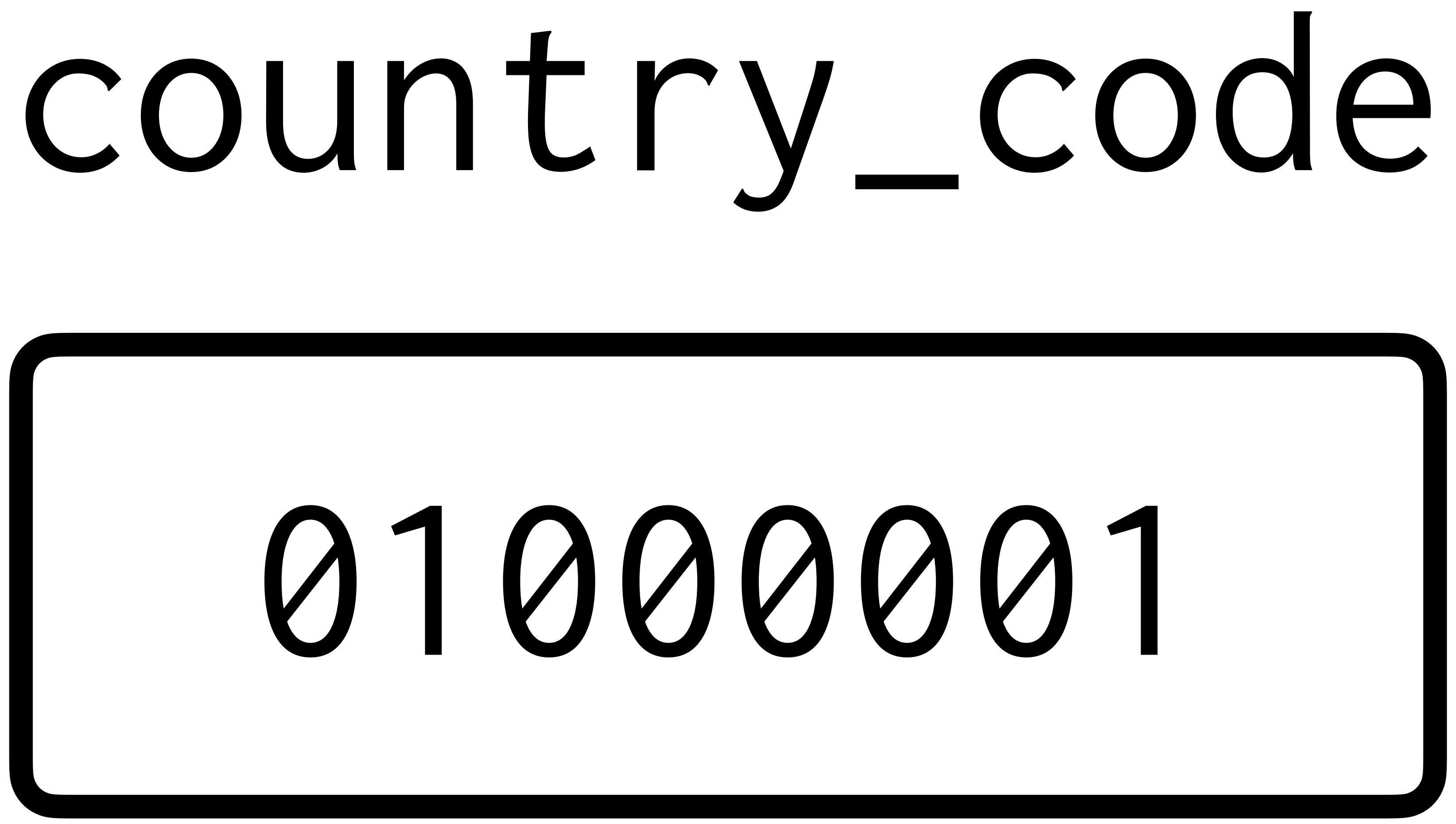
Why does C care about data types?



country_code

65

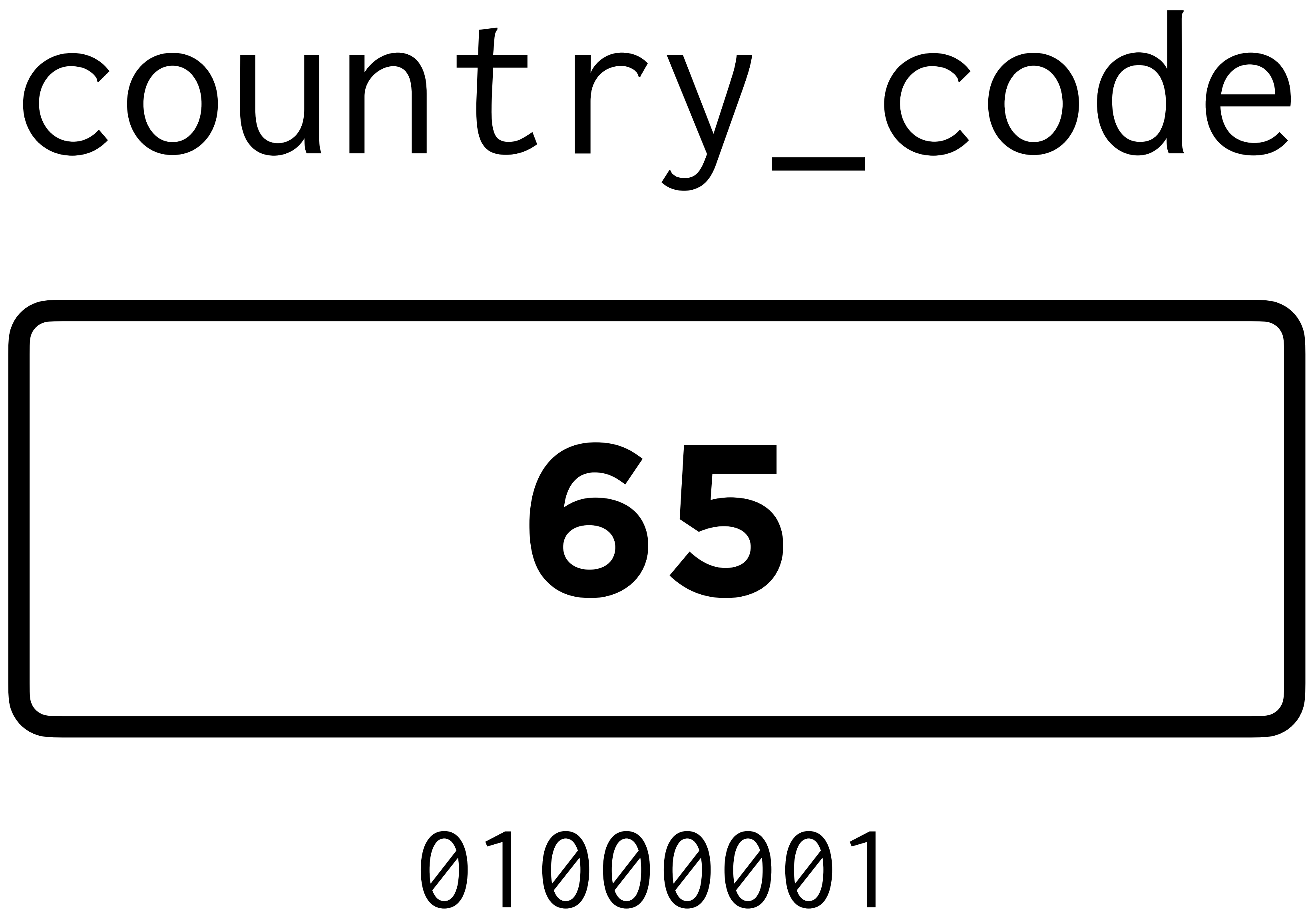
```
int country_code = 65;
```



country_code

01000001

```
int country_code = 65;
```

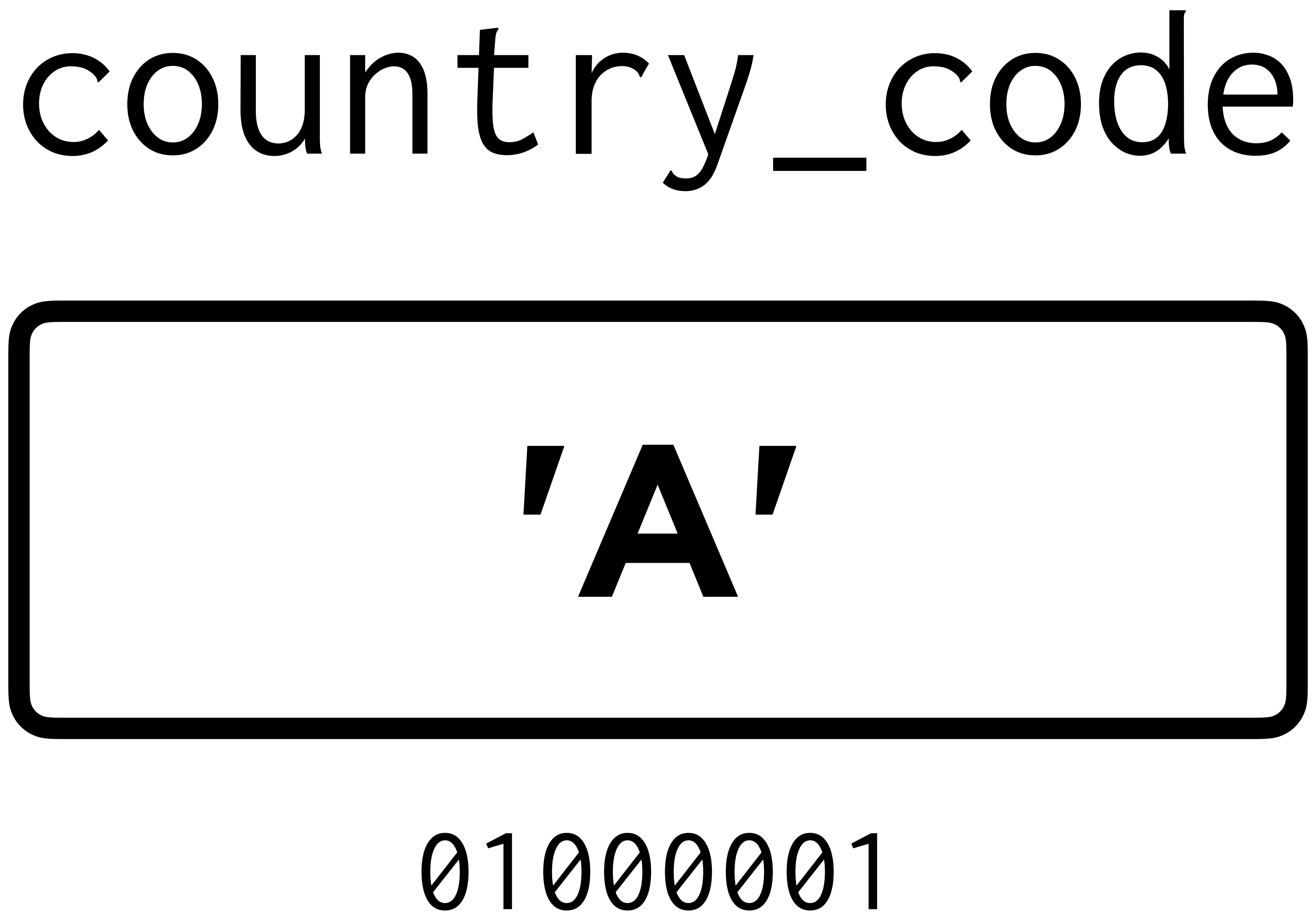


country_code

65

01000001

```c
char country_code = 65;
```



country_code

'A'

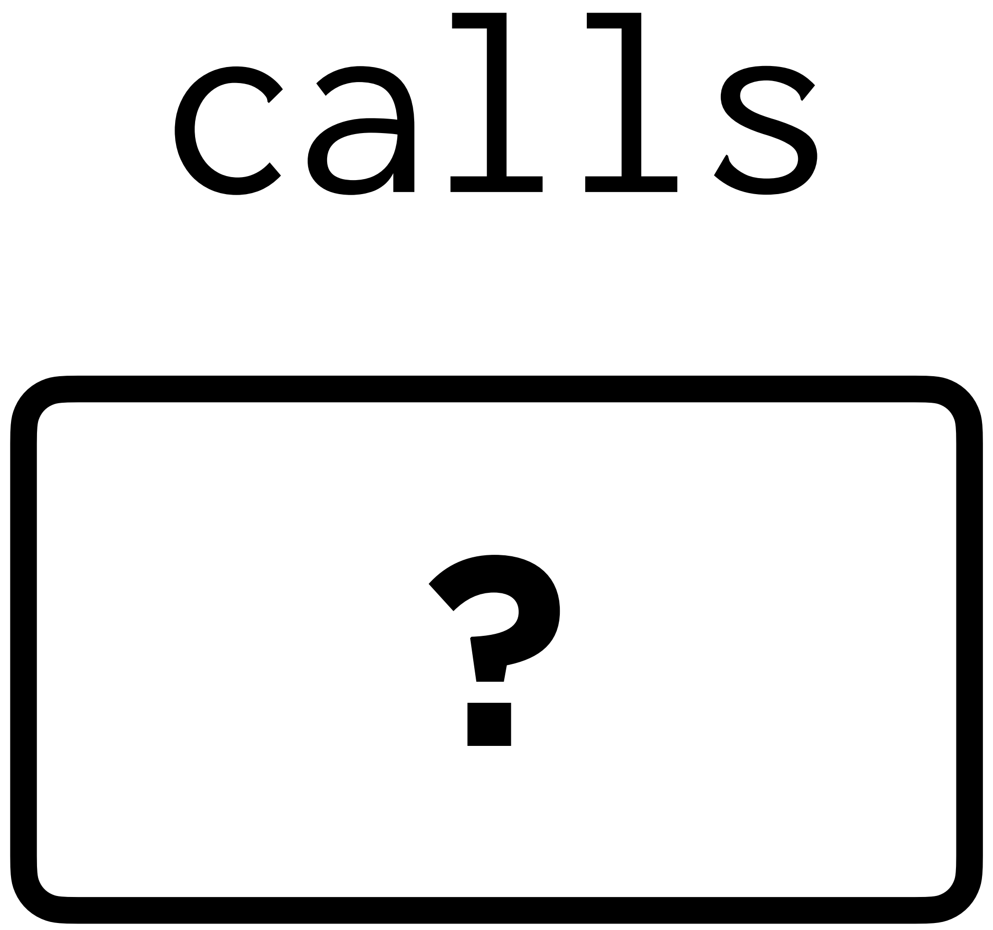
01000001

```
int calls = 4;
calls = calls + 2;
calls = calls - 1;
calls = calls * 2;
calls = calls / 2;
```

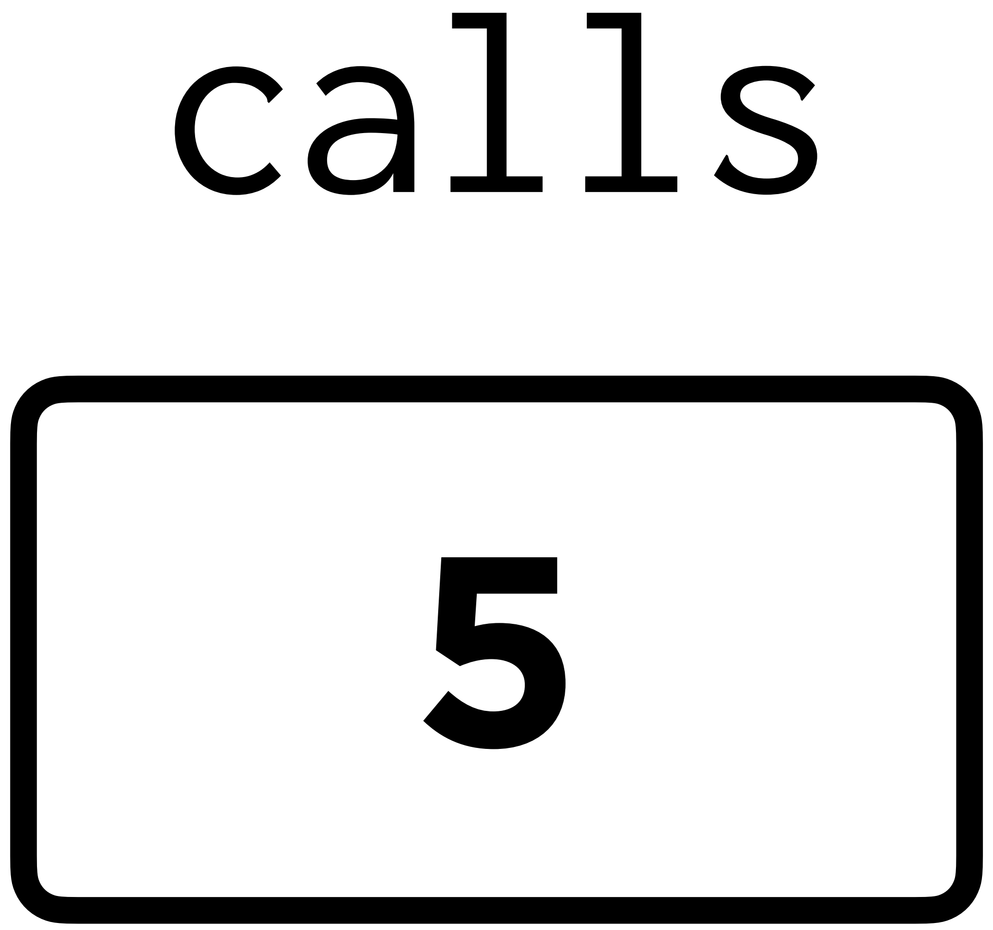
```
int calls = 4;
calls += 2;
calls -= 1;
calls *= 2;
calls /= 2;
```

"Syntactic sugar"

```
int calls = 4;
calls = calls + 2;
calls = calls - 1;
calls = calls * 2;
calls = calls / 2;
```



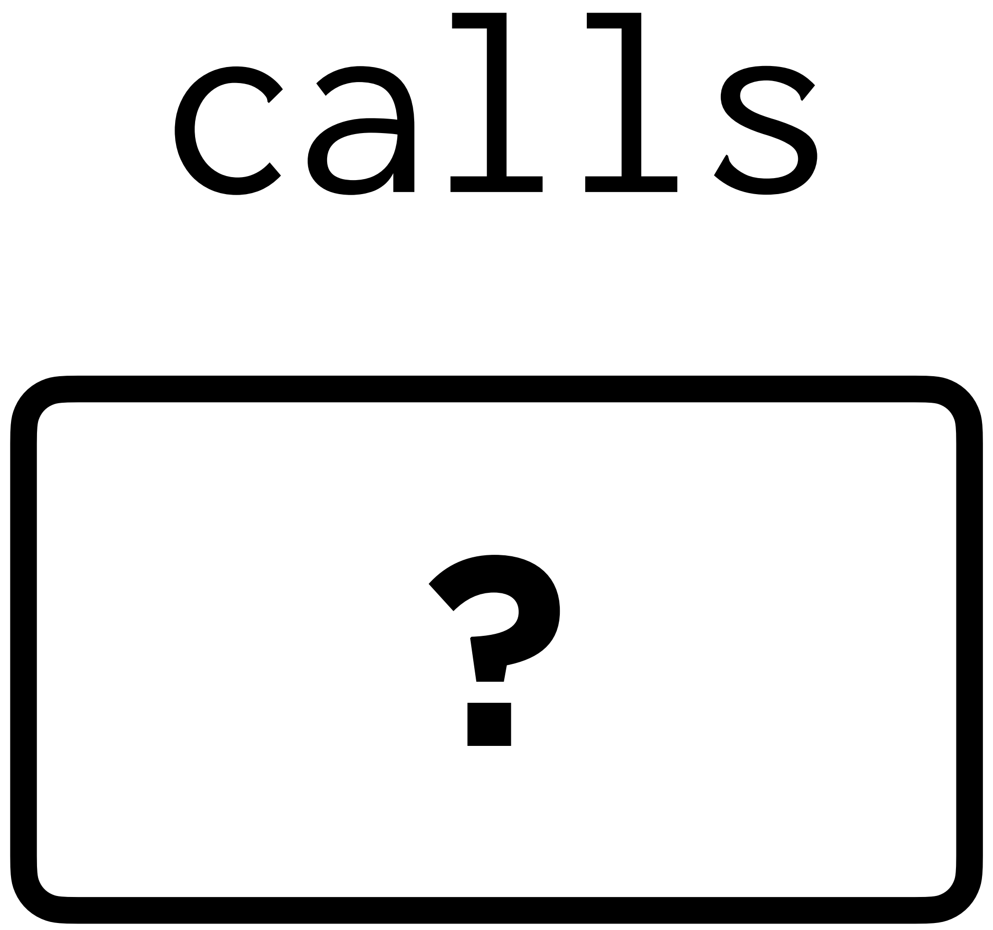
```
int calls = 4;
calls = calls + 2;
calls = calls - 1;
calls = calls * 2;
calls = calls / 2;
```



calls

5

```
int calls = 4;
calls = calls + 1;
calls = calls - 2;
calls = calls * 3;
calls = calls / 2;
```

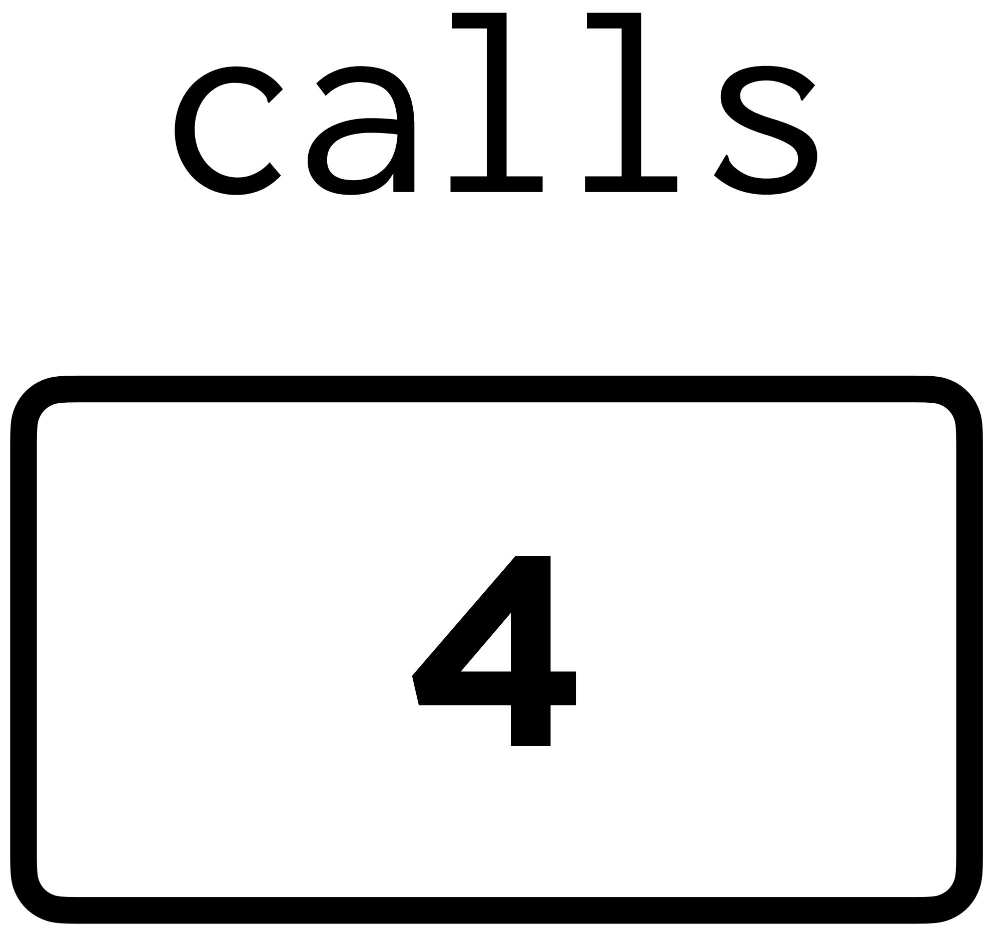


calls

?

```
int calls = 4;
calls = calls + 1;
calls = calls - 2;
calls = calls * 3;
calls = calls / 2;
```

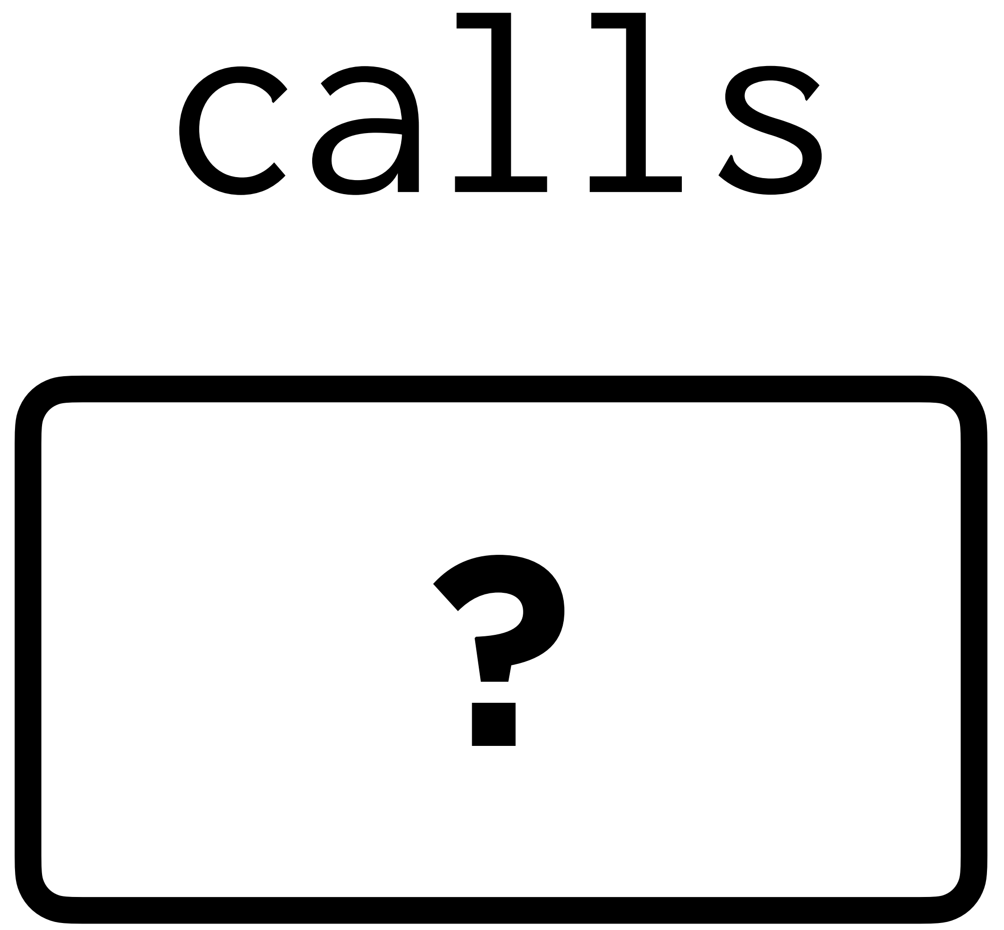
"Truncation"



Input and Printing

```
int calls = get_int("Calls: ");
int calls = get_int("Calls: ");
```

function call

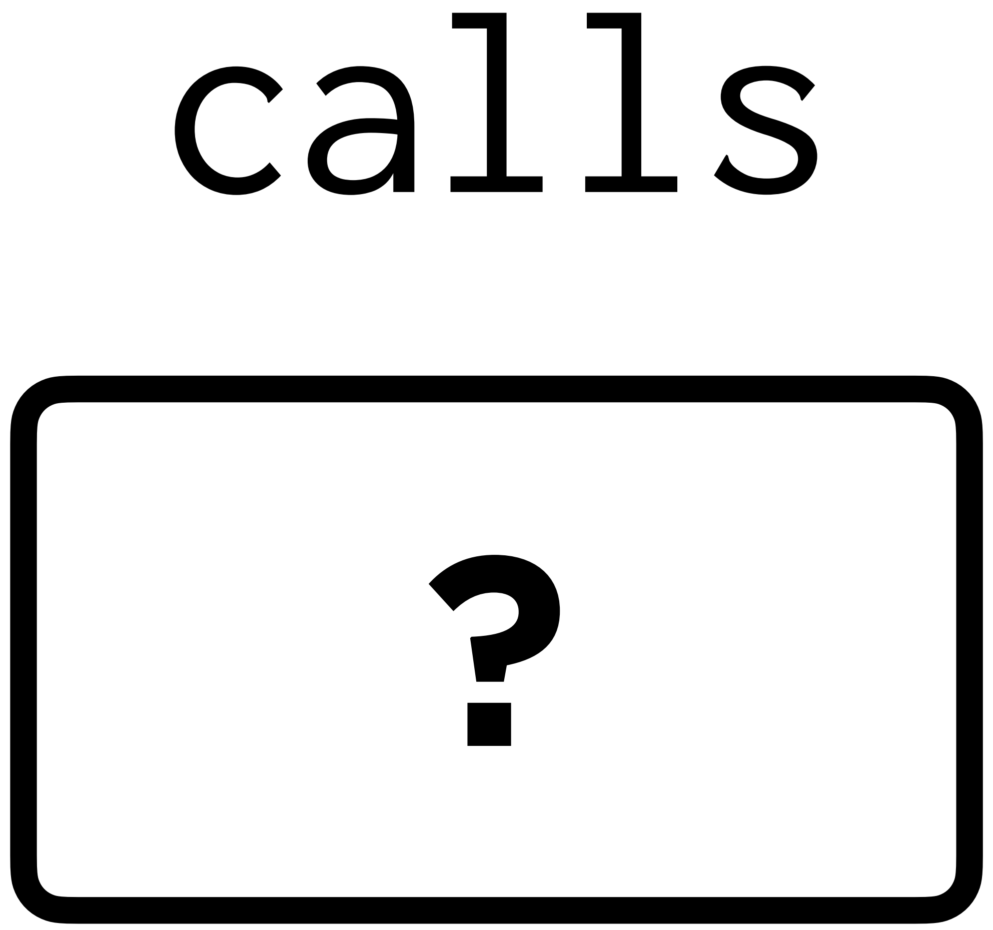


calls

?

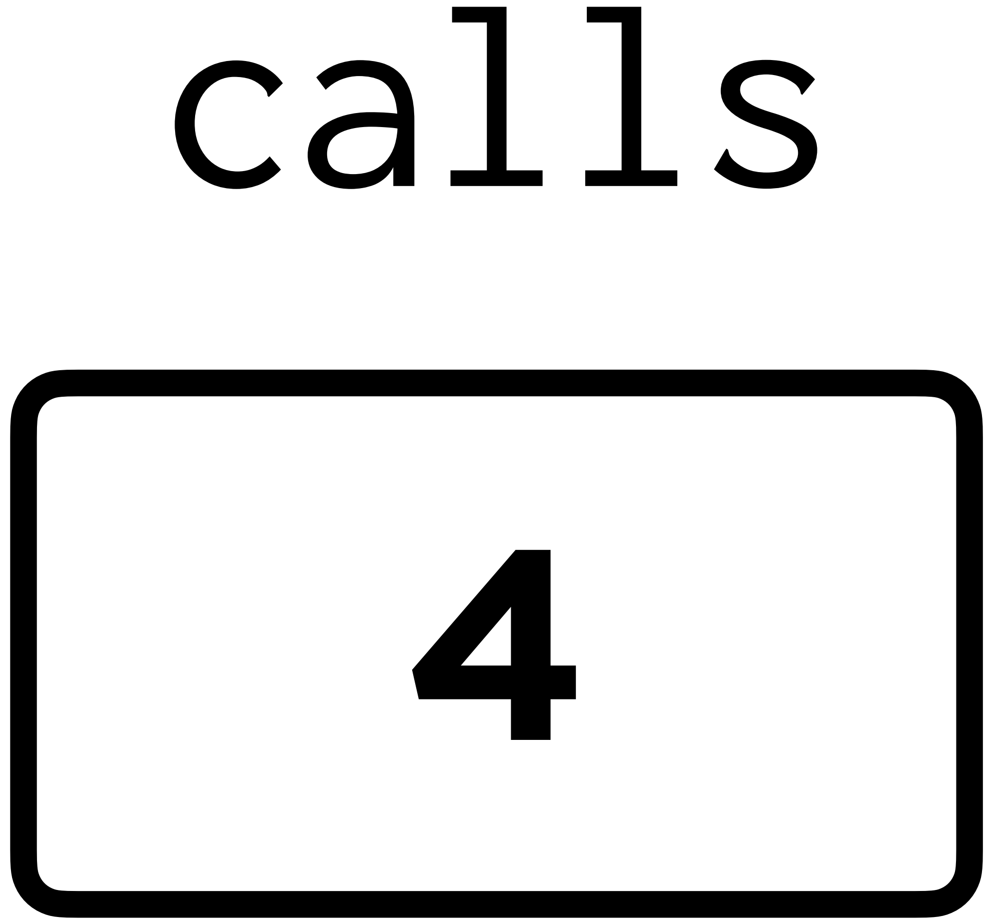
```
int calls = 4;
```

value



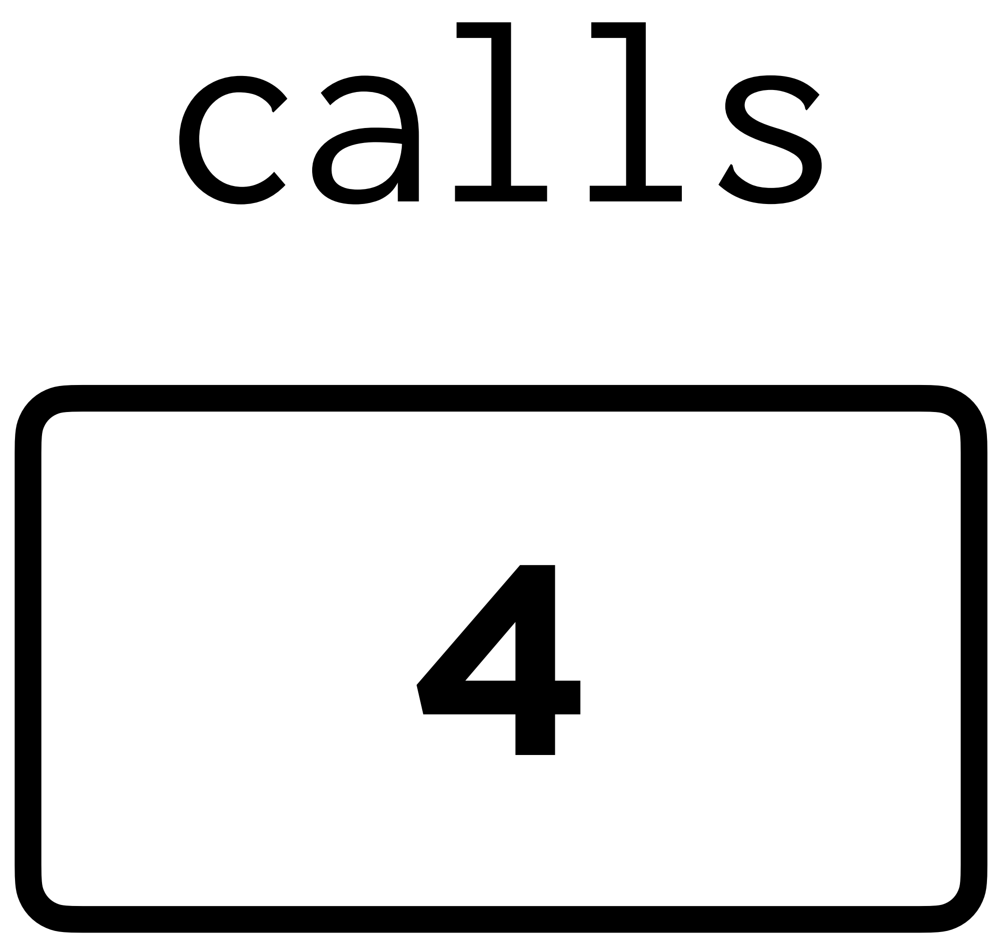
```
int calls = 4;
```

value



```c
int calls = 4;
printf("calls is %i\n", calls);
```

"calls is 4"

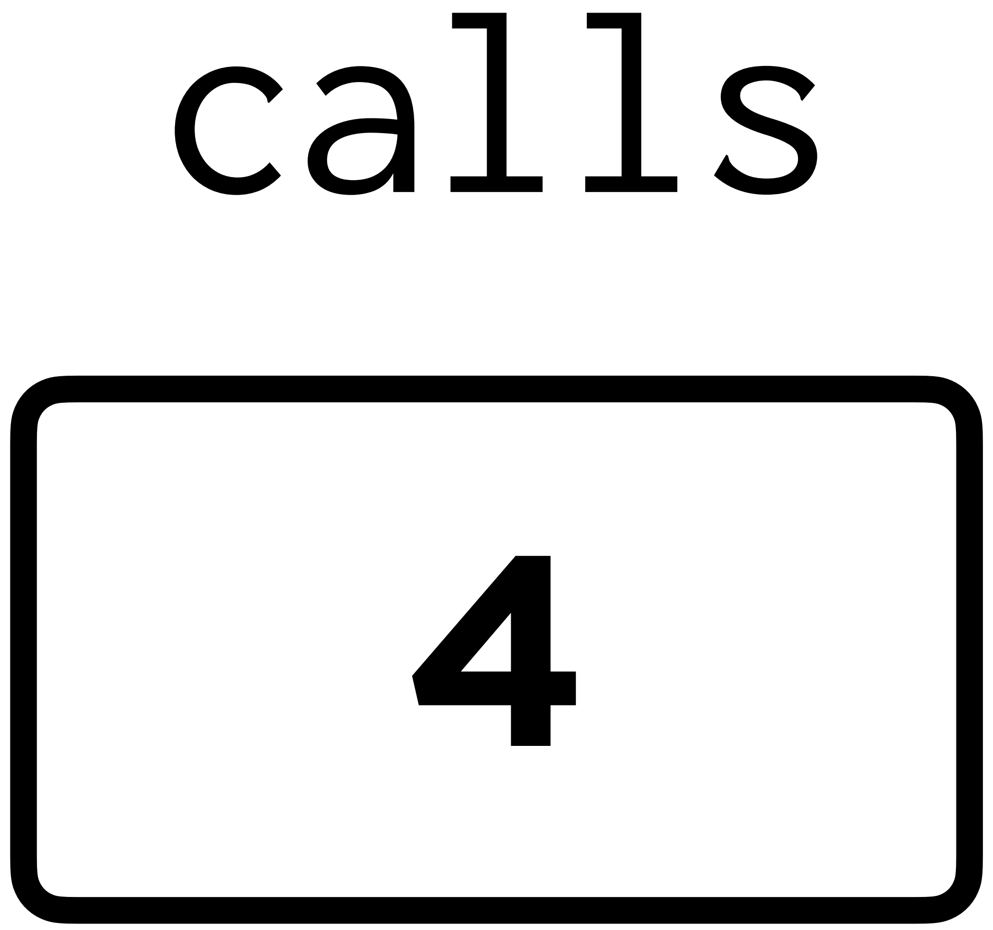


calls

4

```
int calls = 4;
printf("calls is %i\n", calls);
```

format code



calls

4

Types and format codes

- `int (%i)`
- `float (%f)`
- `char (%c)`
- `string (%s)`

Hello, world!

- Let's write a "Hello, world" program to complete the first step of Problem Set 1.
- Visit the link at https://carterzenke.me/sections.

Hello, me!

- Let's write a "Hello, me" program to complete the second step of Problem Set 0.

Functions, Loops, and Conditionals

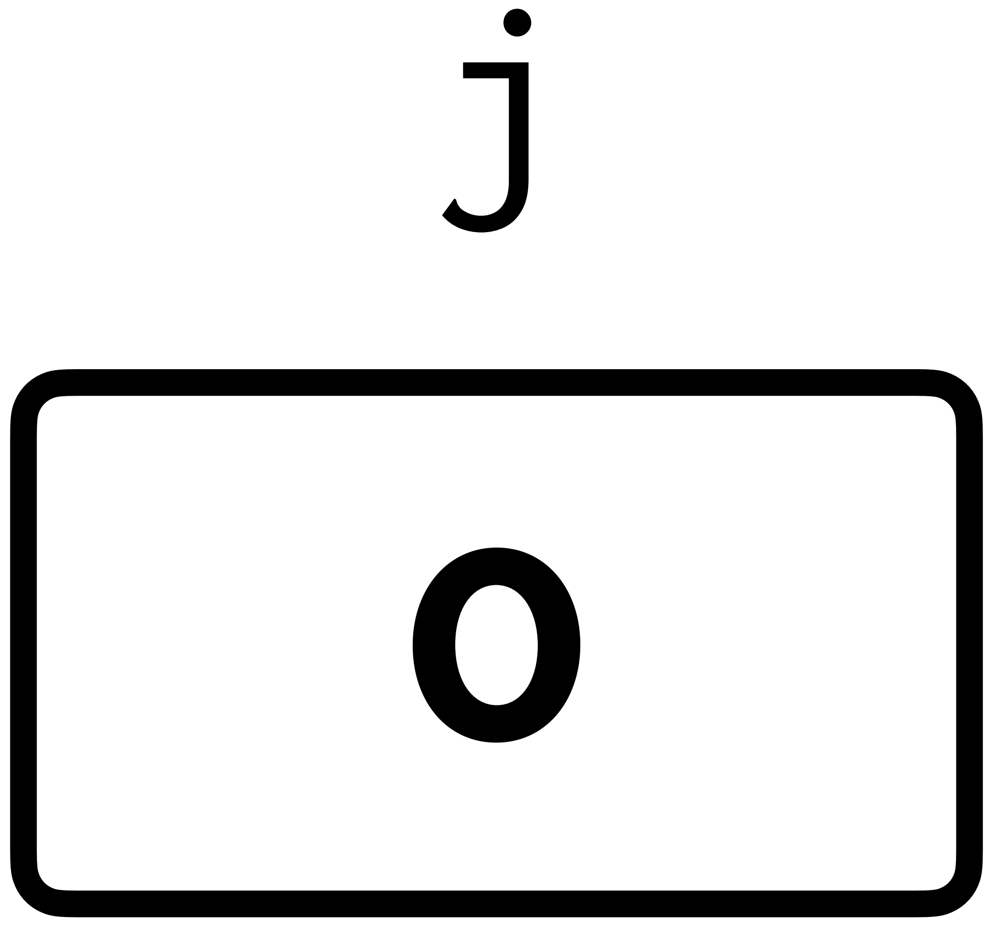
While Loops

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```

```c
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```

```
####
```

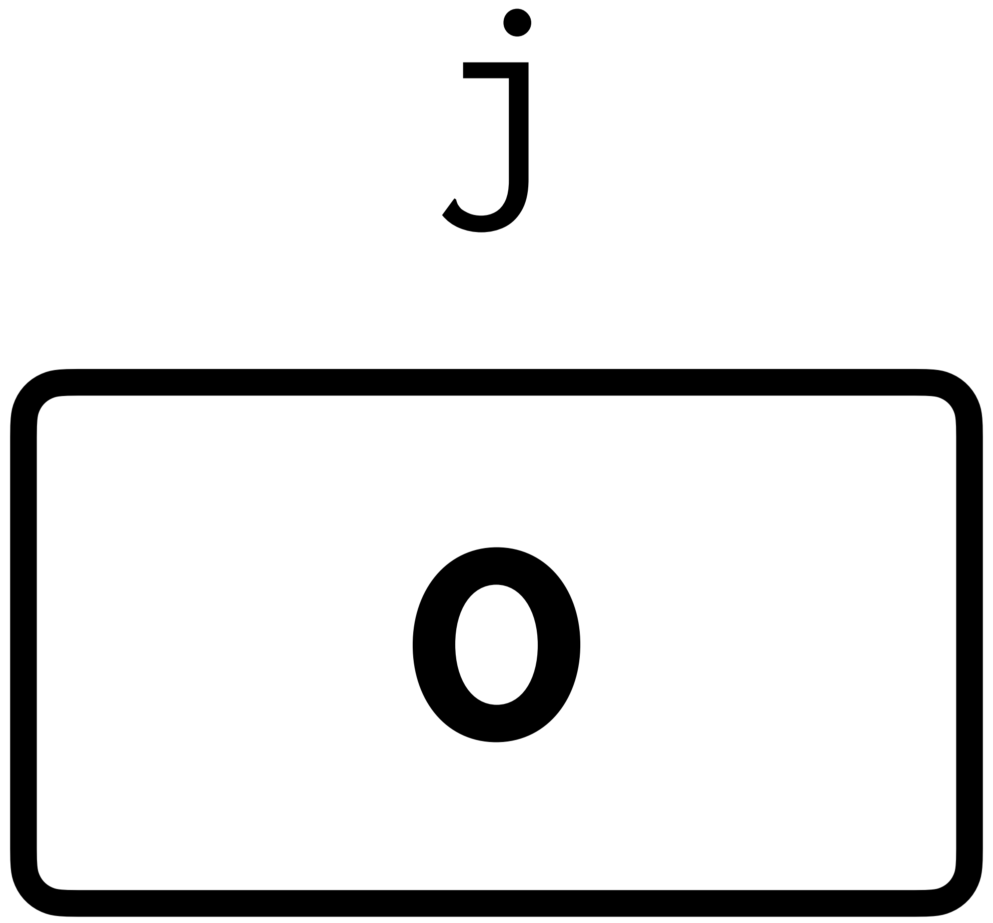
```c
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



j

0

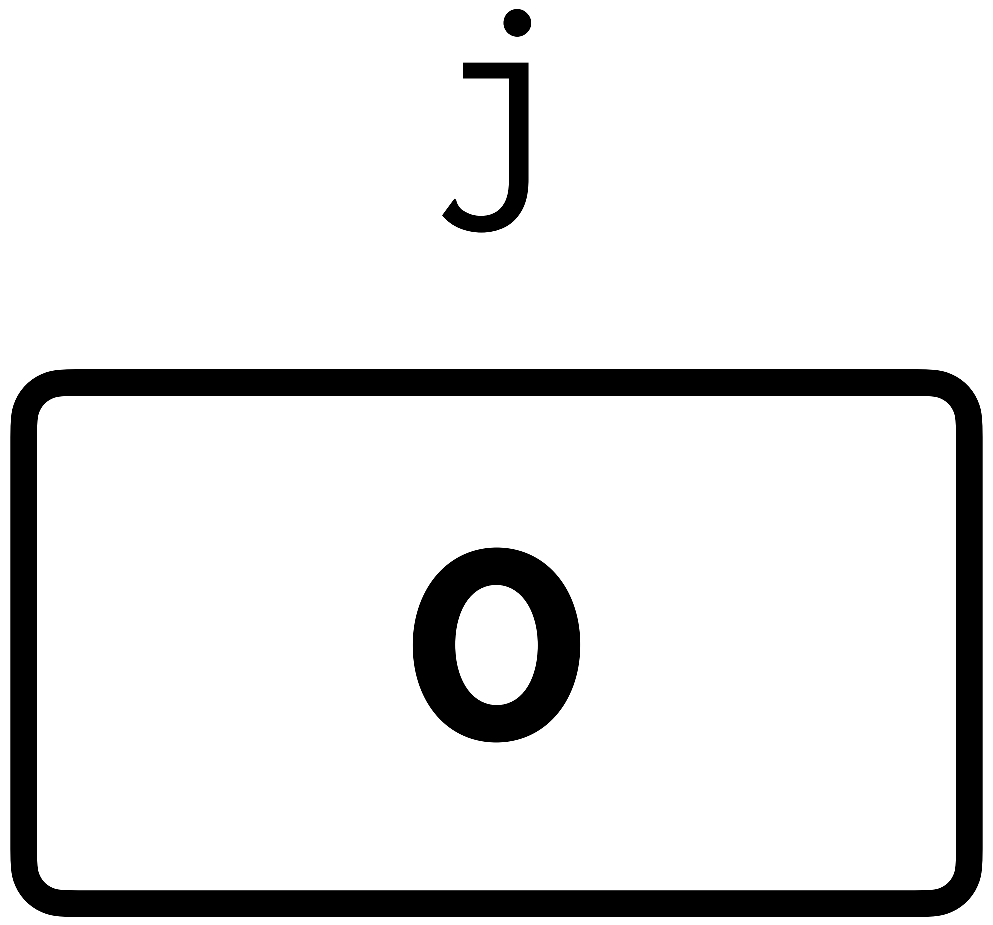
```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



j

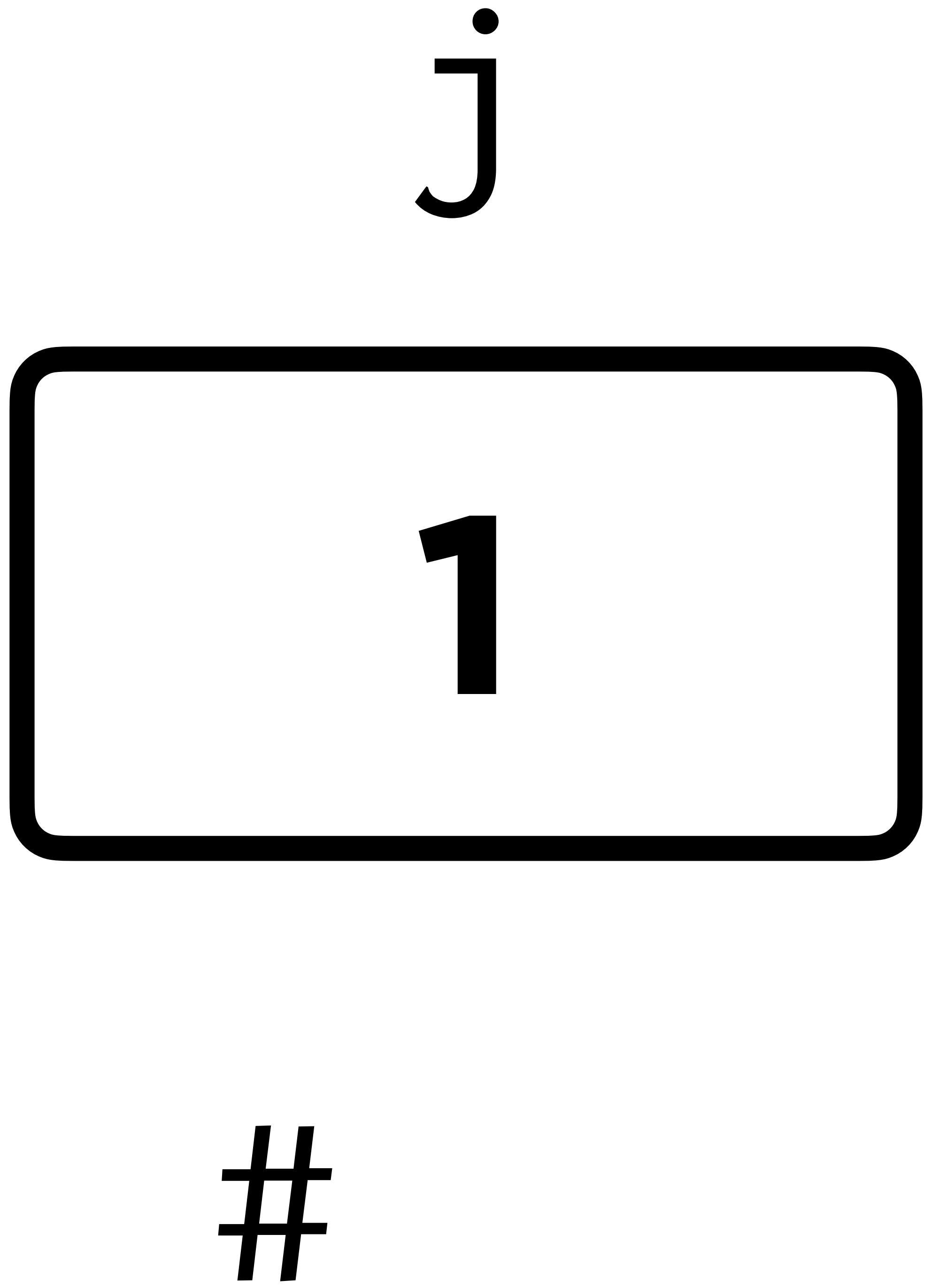
0

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



```
#
```

```c
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```

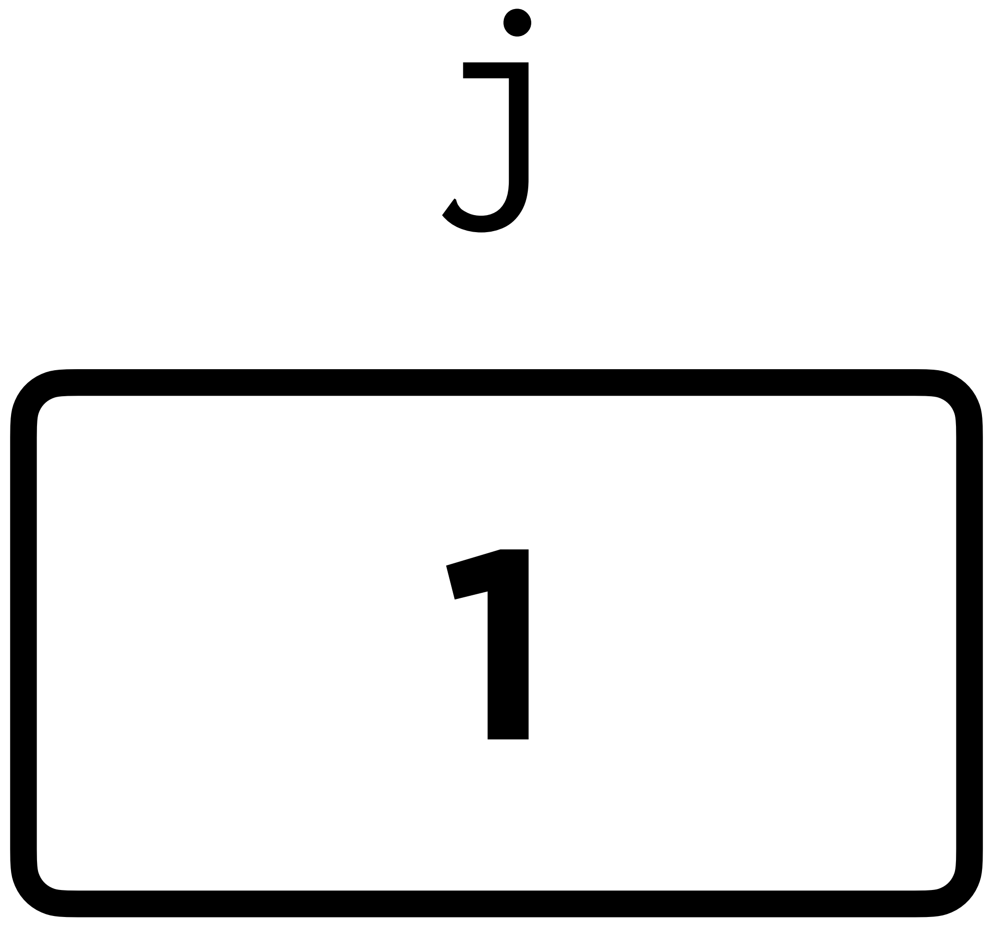


j

1

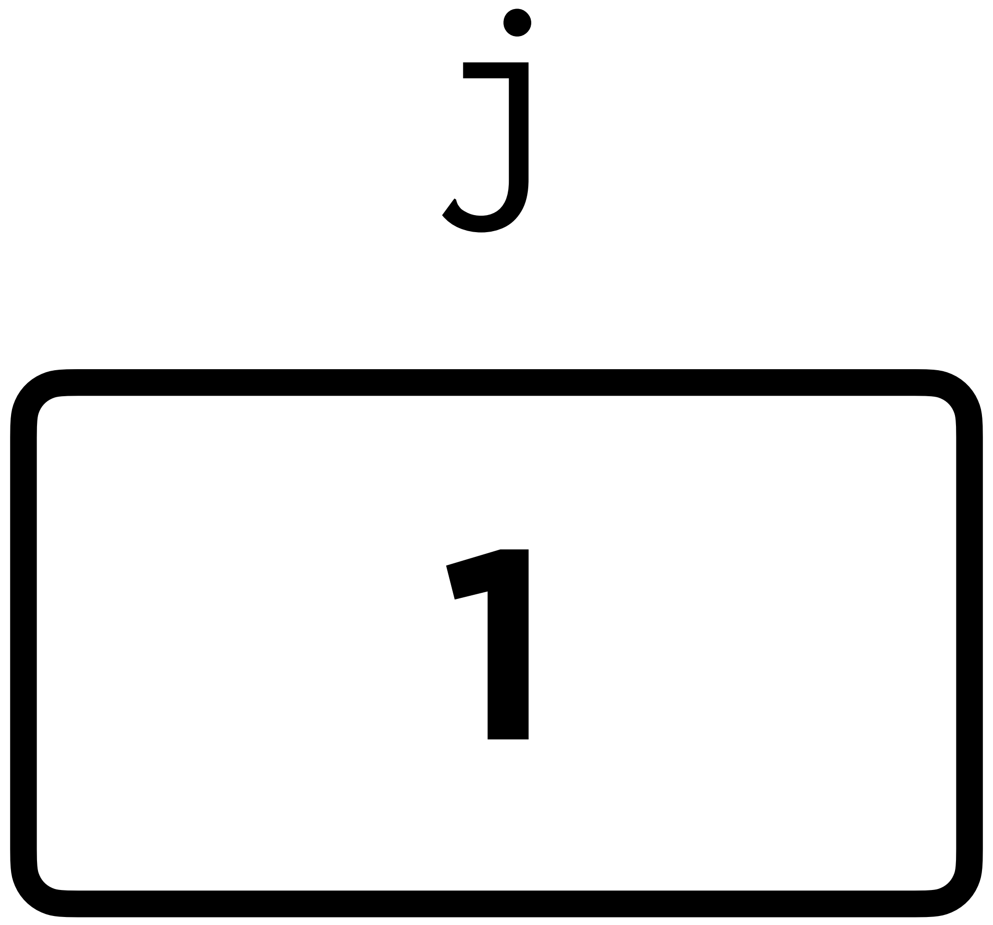
#

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



#

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```

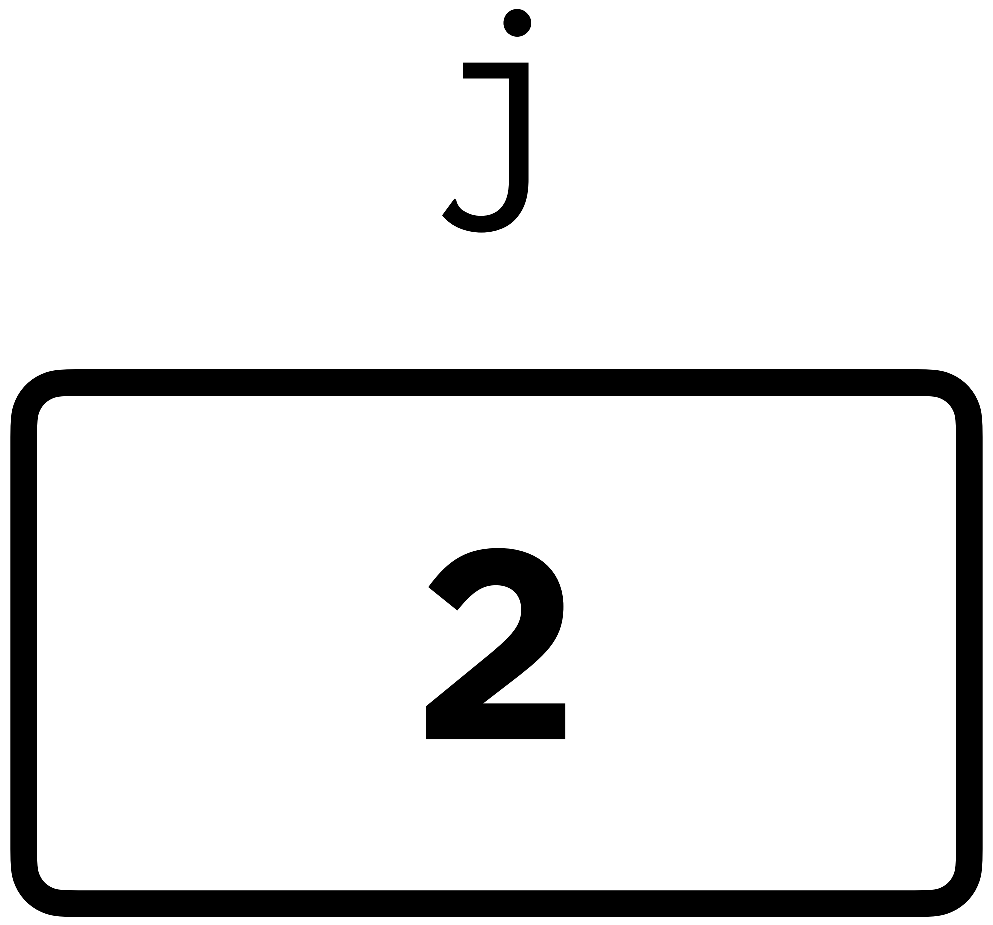


j

1

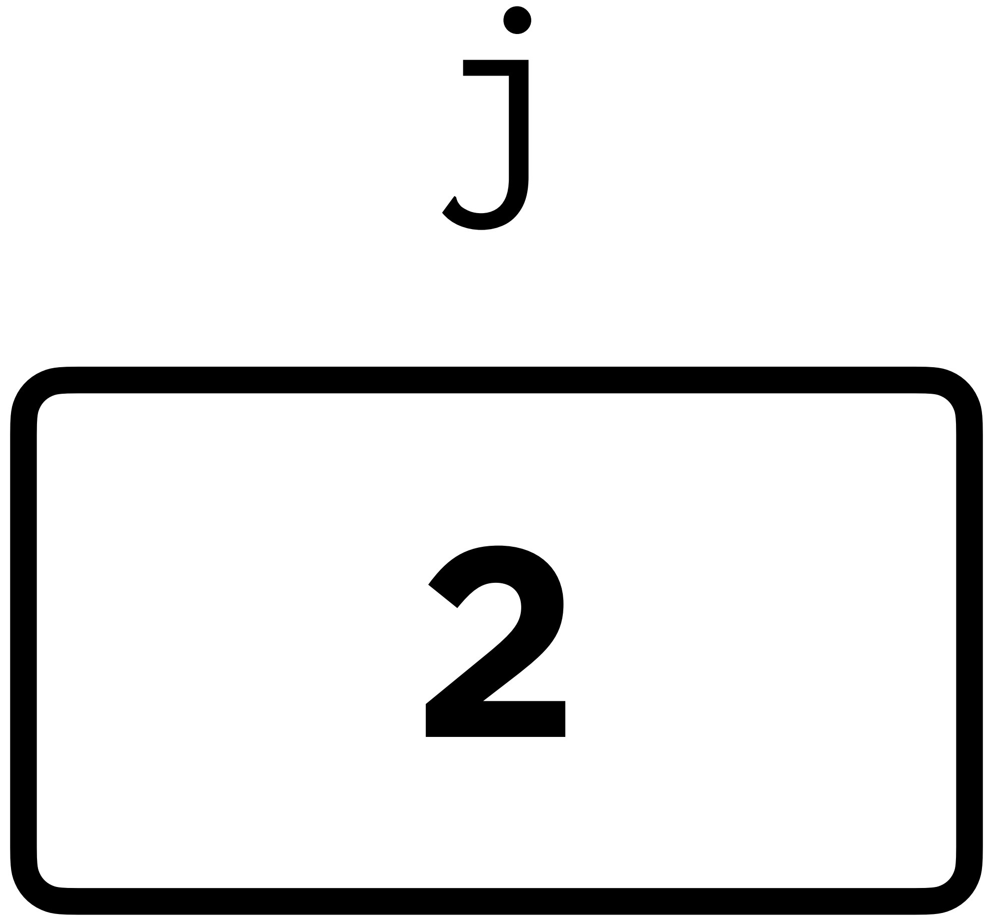
##

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



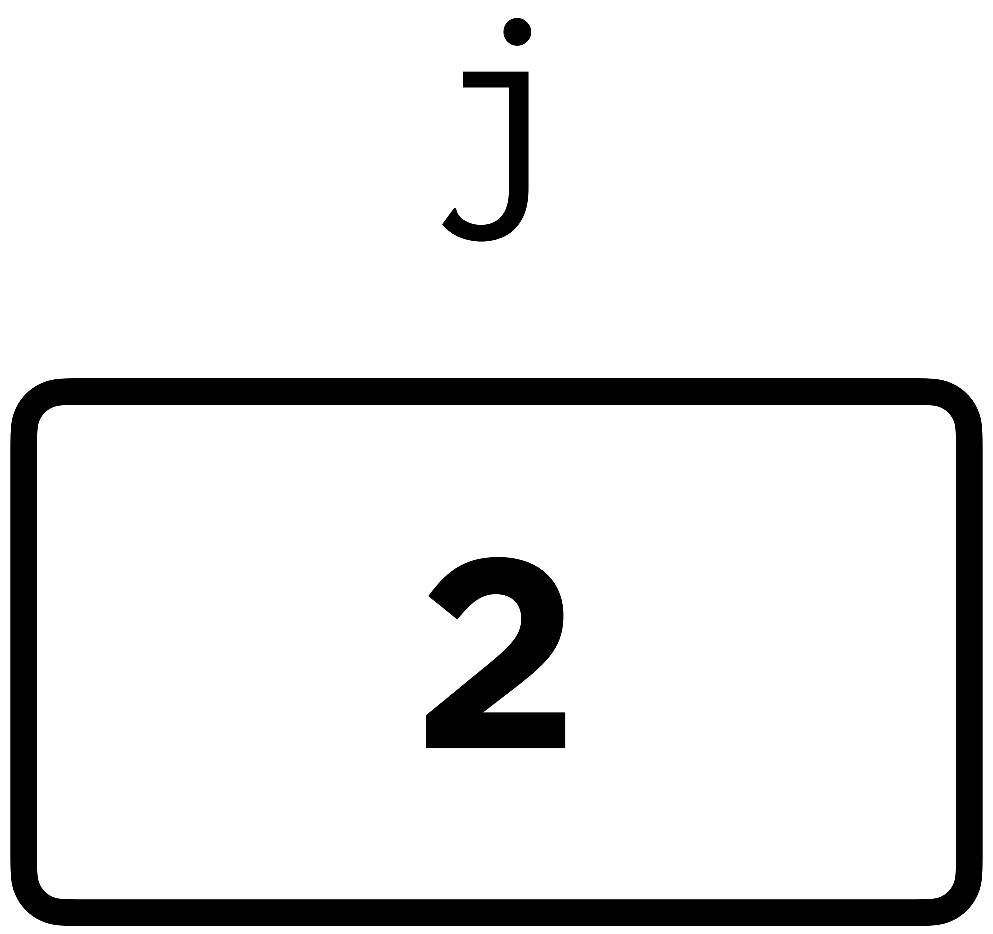
```
##
```

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



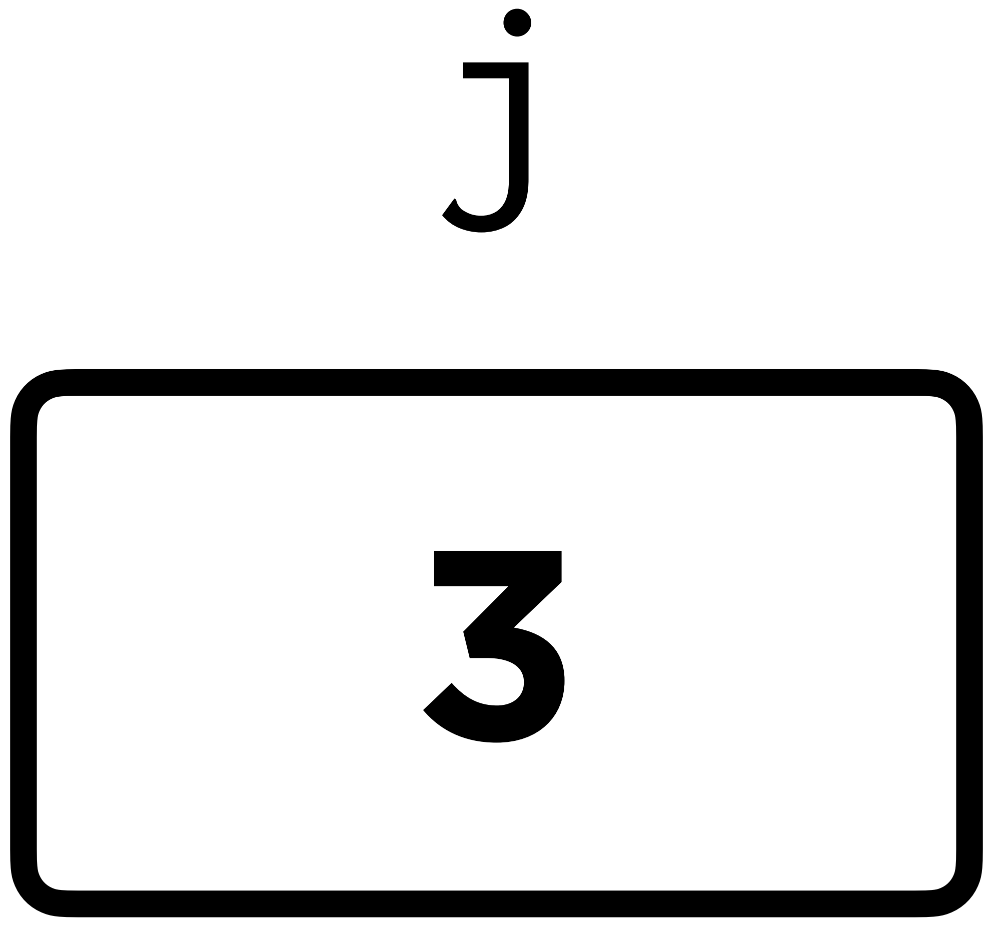
```
##
```

```c
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



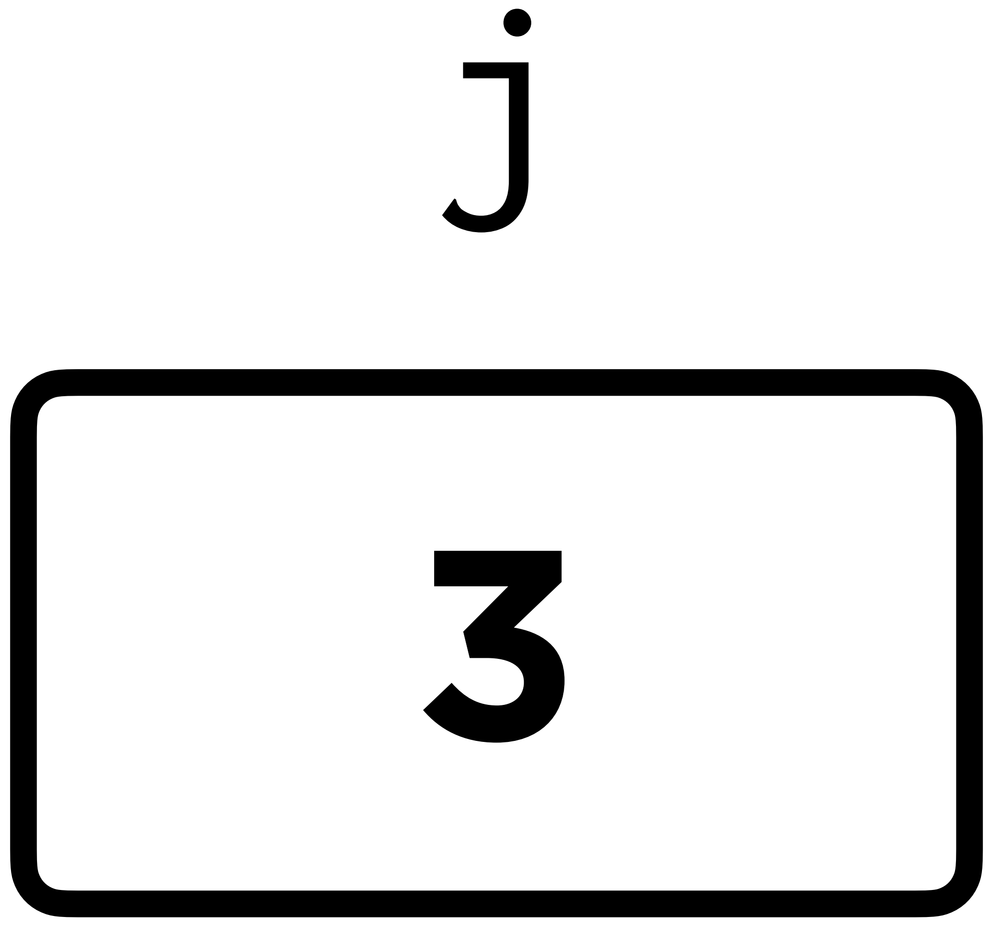
```
###
```

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



```
###
```

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



j

3

###

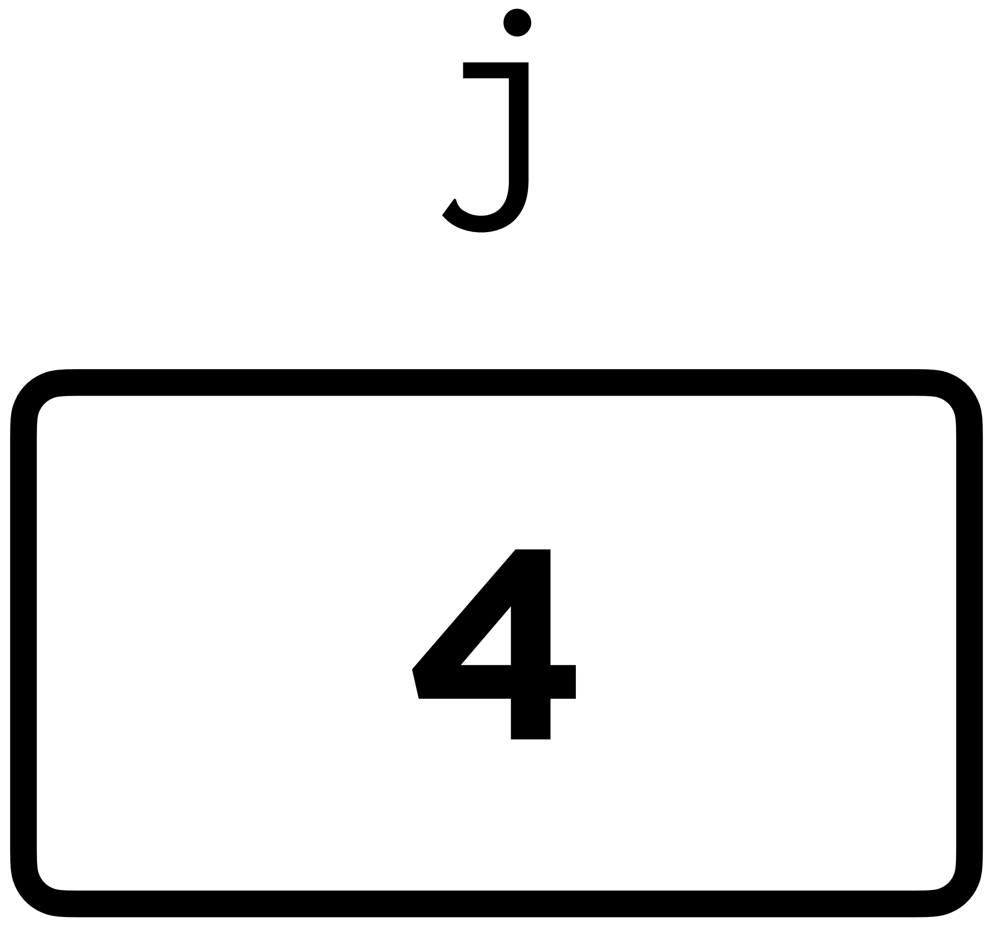
```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```

j

3

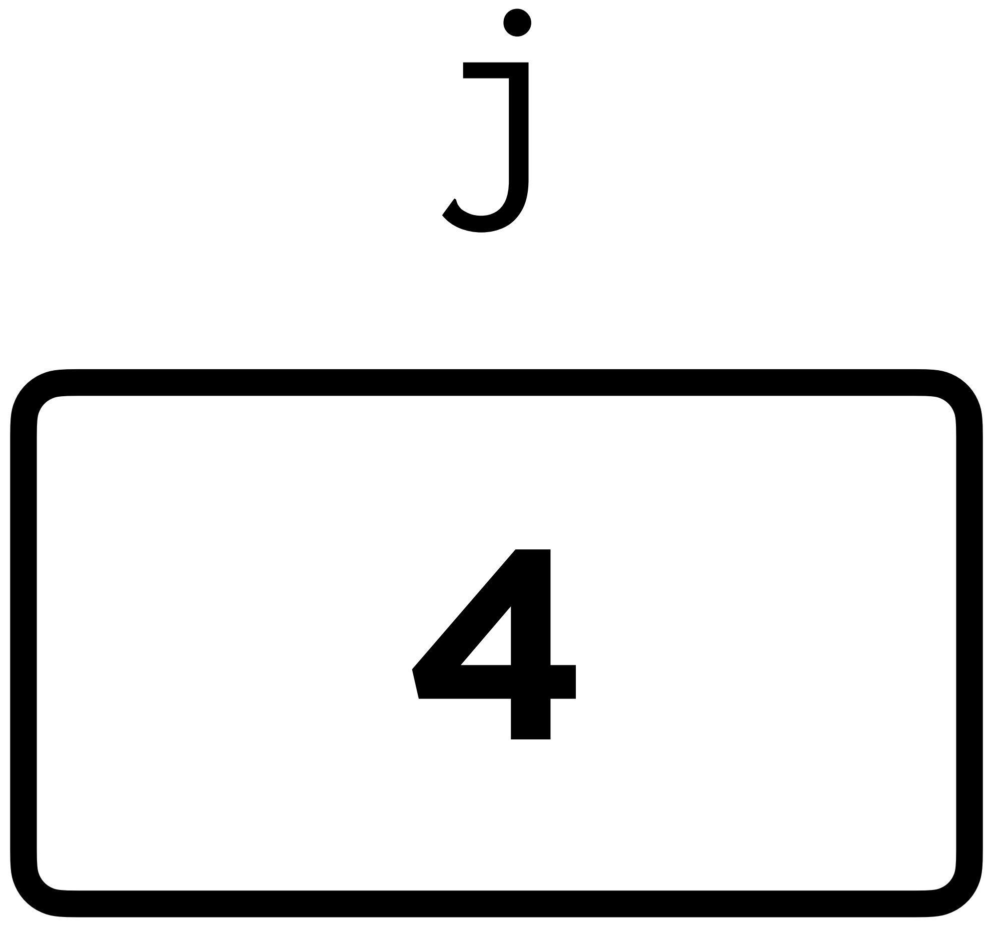
####

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



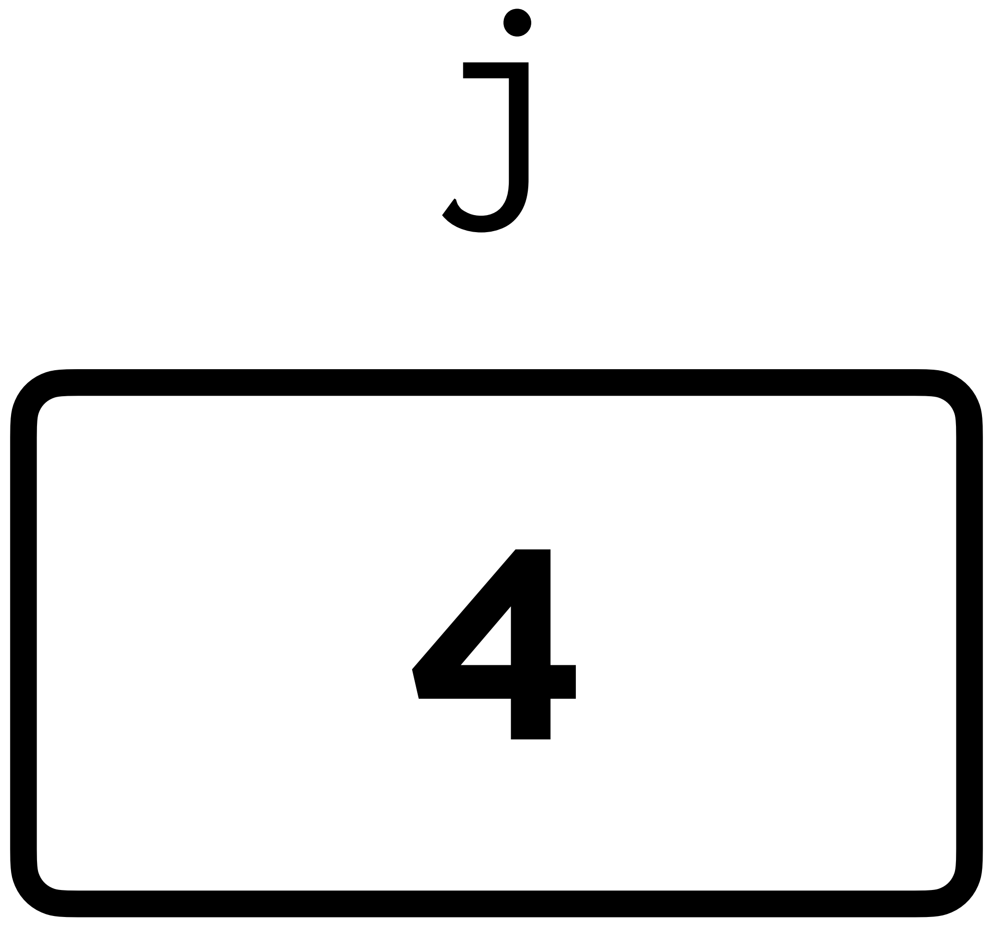
```
####
```

```c
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



```
####
```

```
int j = 0;
while (j < 4)
{
    printf("#");
    j++;
}
printf("\n");
```



####\n

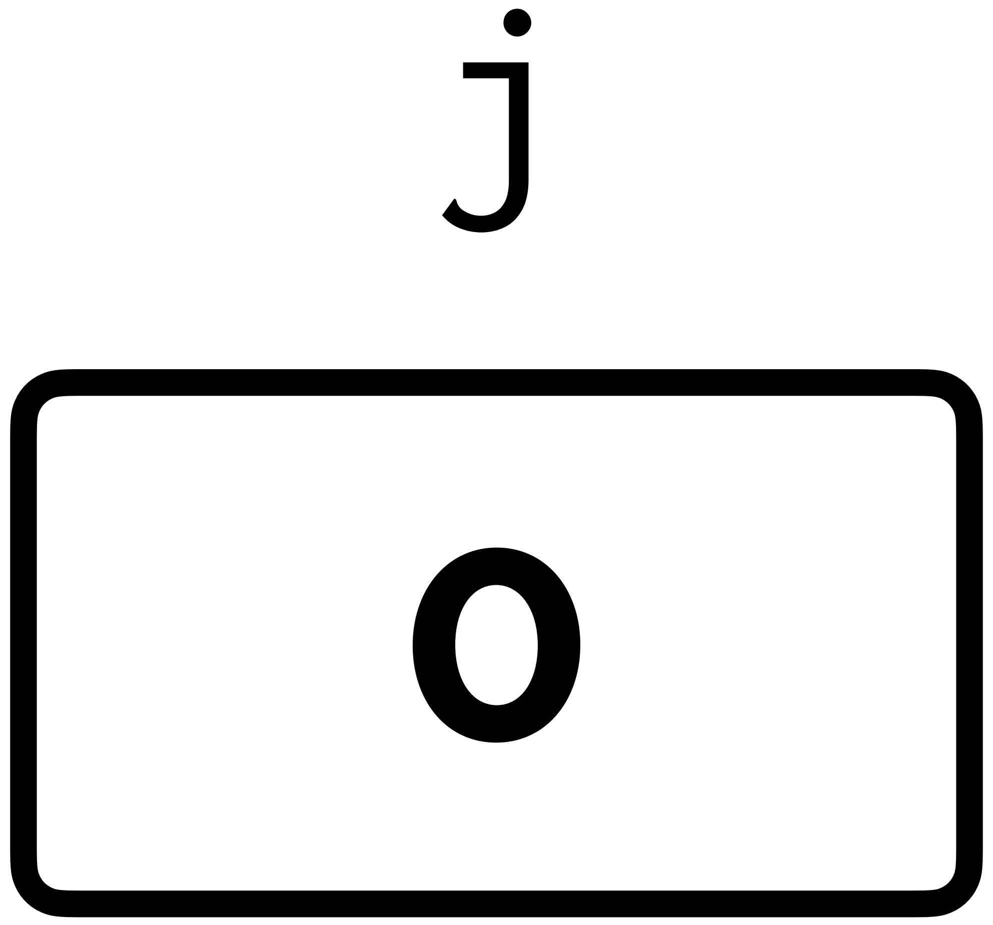
For Loops

```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```

```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```

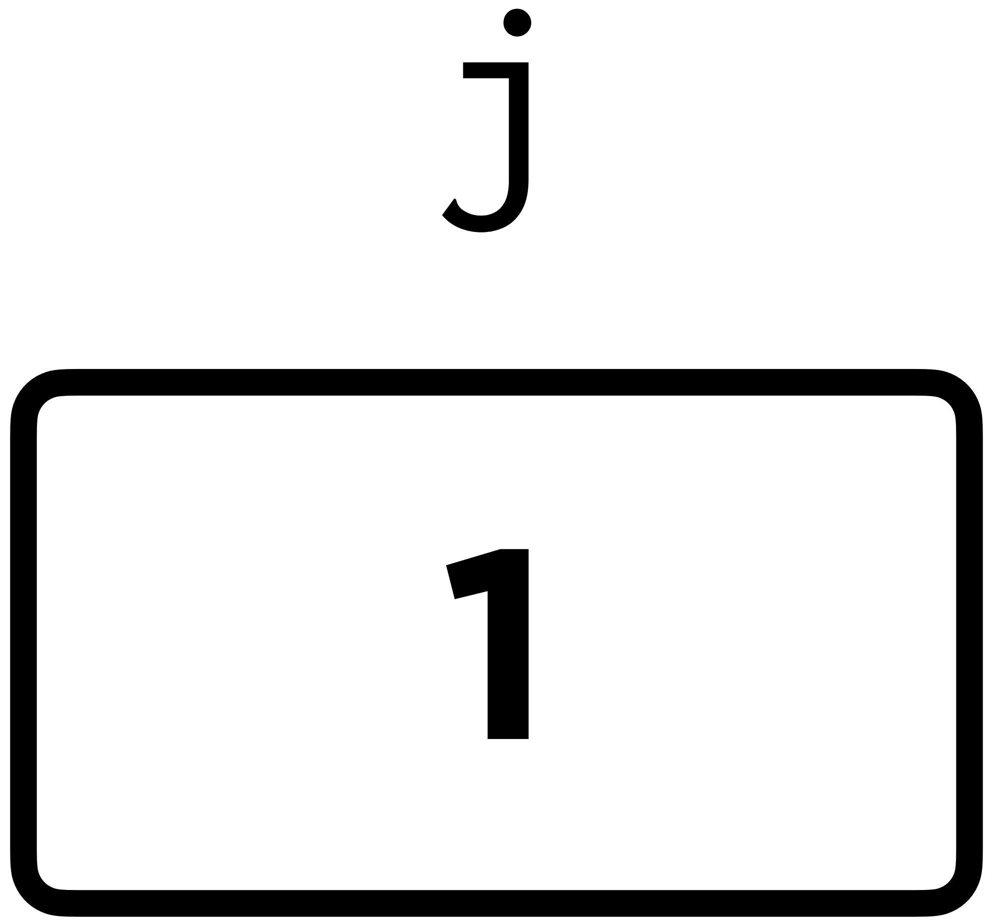
```
####
```

```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```



#

```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```



```
##
```

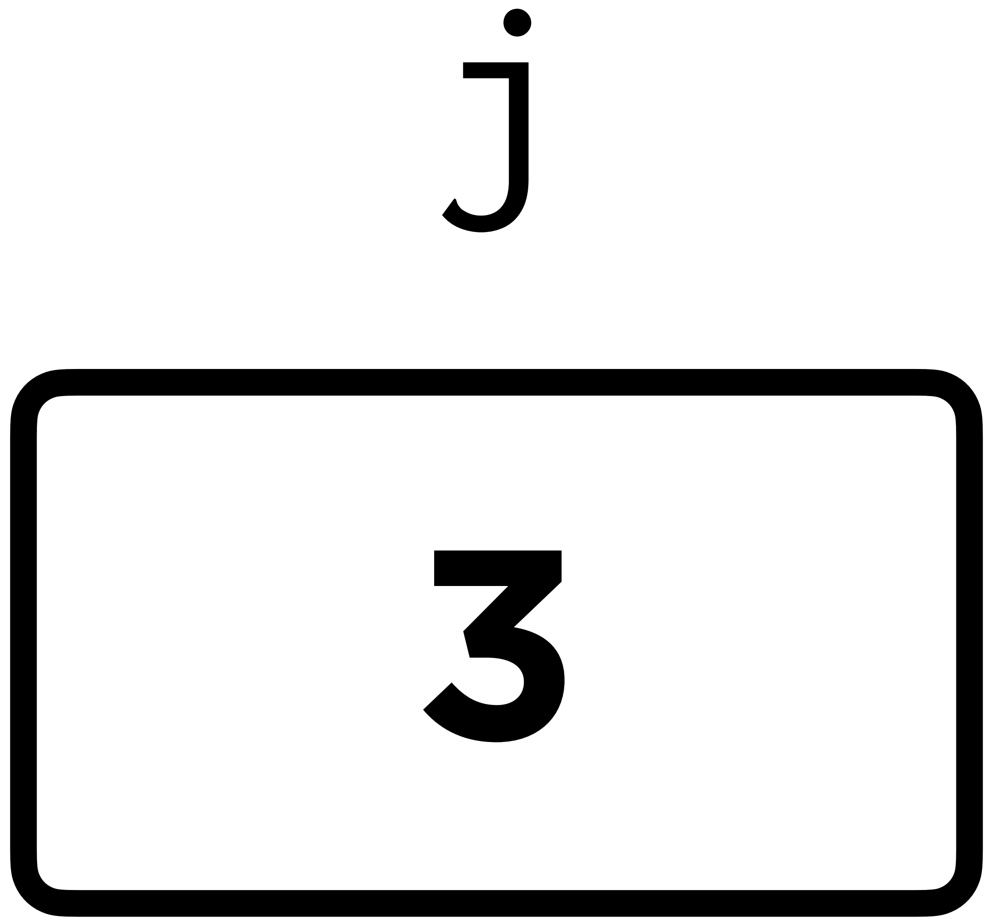
```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```

j

2

###

```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```

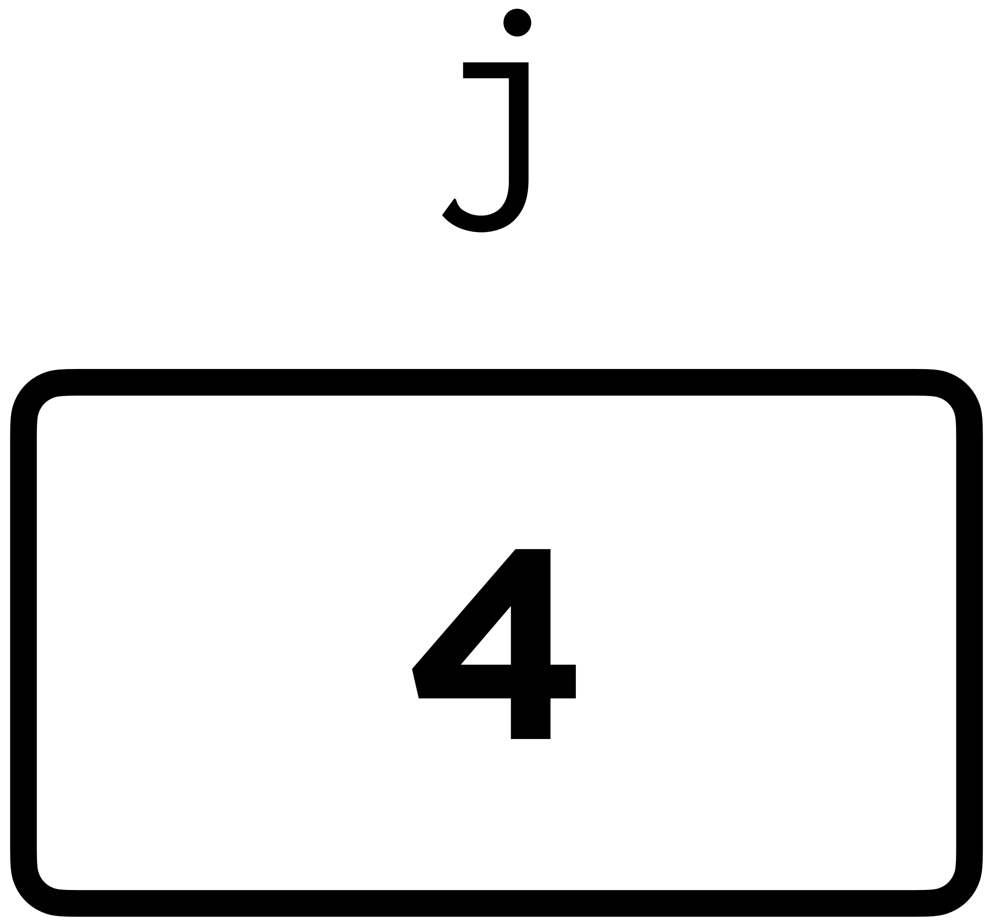


j

3

```
####
```

```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```



j

4

####\n

```
for (int j = 0; j < 4; j++)
{
    printf("#");
}
printf("\n");
```

```
for (int i = 0; i < 4; i++)
{
    for (int j = 0; j < 4; j++)
    {
        printf("#");
    }
    printf("\n");
}
```

```c
for (int i = 0; i < 4; i++)
{
    for (int j = 0; j < 4; j++)
    {
        printf("#");
    }
    printf("\n");
}
```

```
####
####
####
####
```

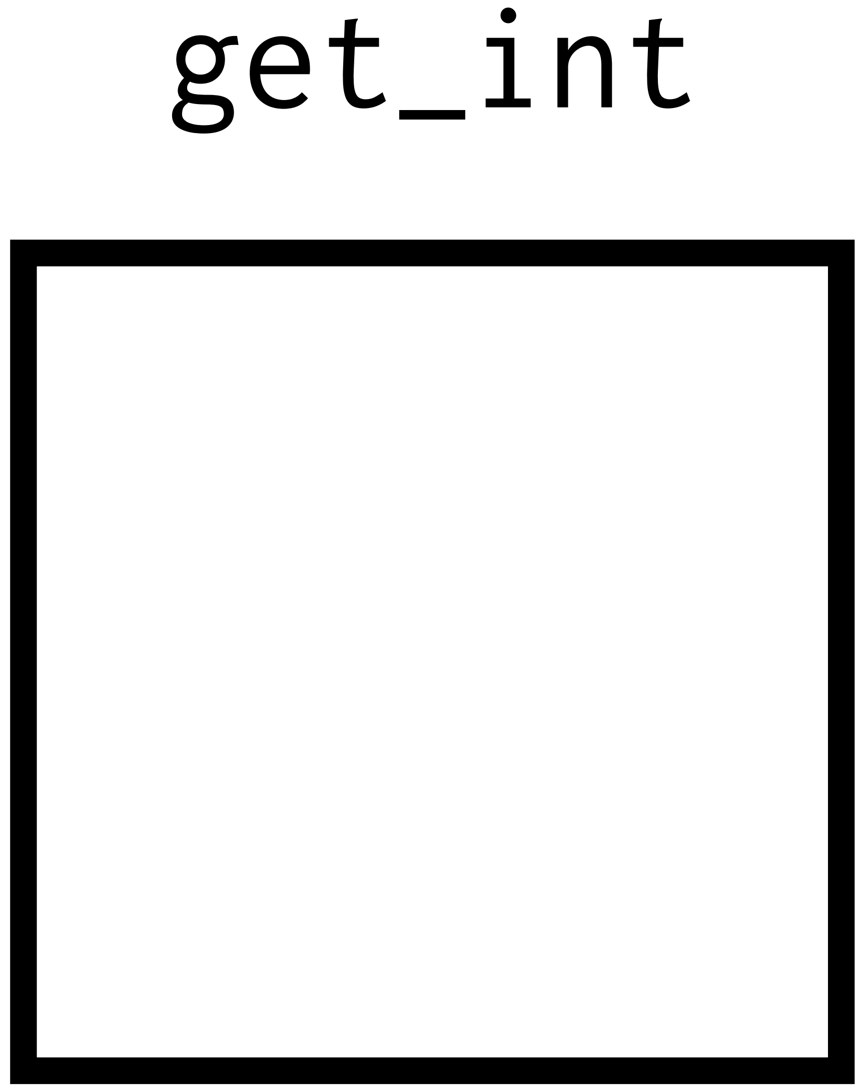
Mario

- Let’s write a program to print a right-aligned pyramid.

Functions

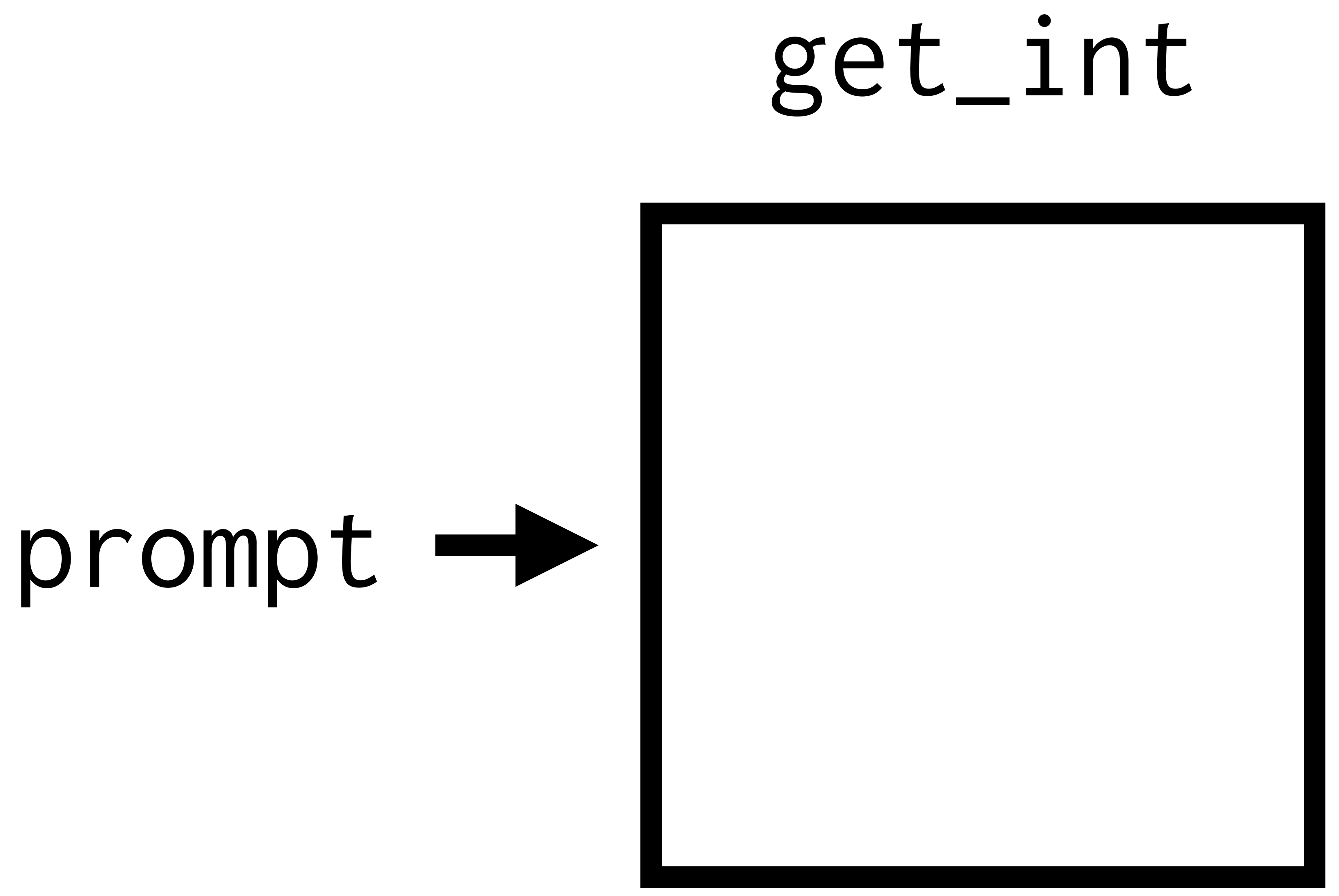
```
int get_int(string prompt)
{
    // Get int from user
}
```

```
int get_int(string prompt)
{
    // Get int from user
}
```

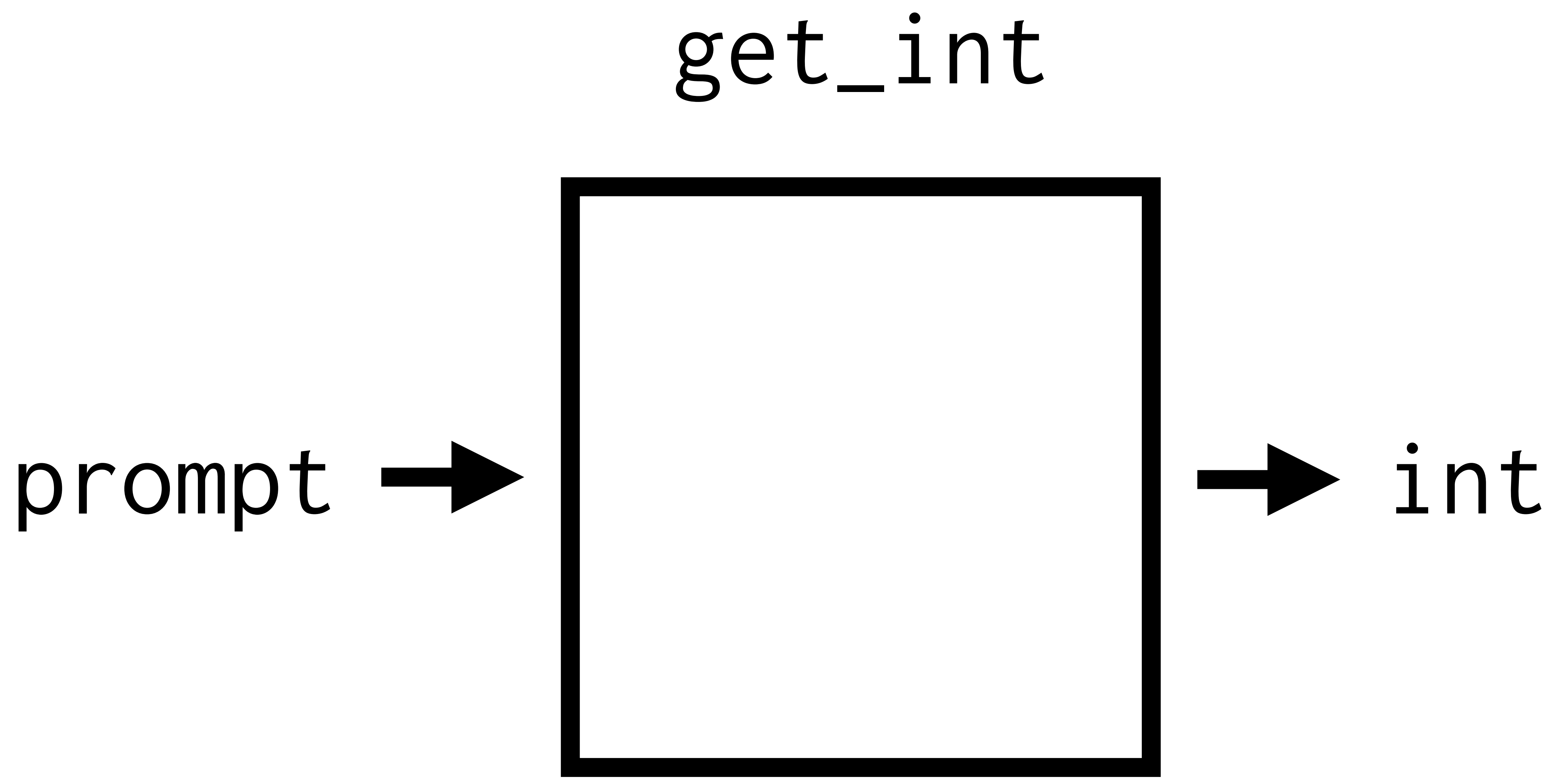


get_int

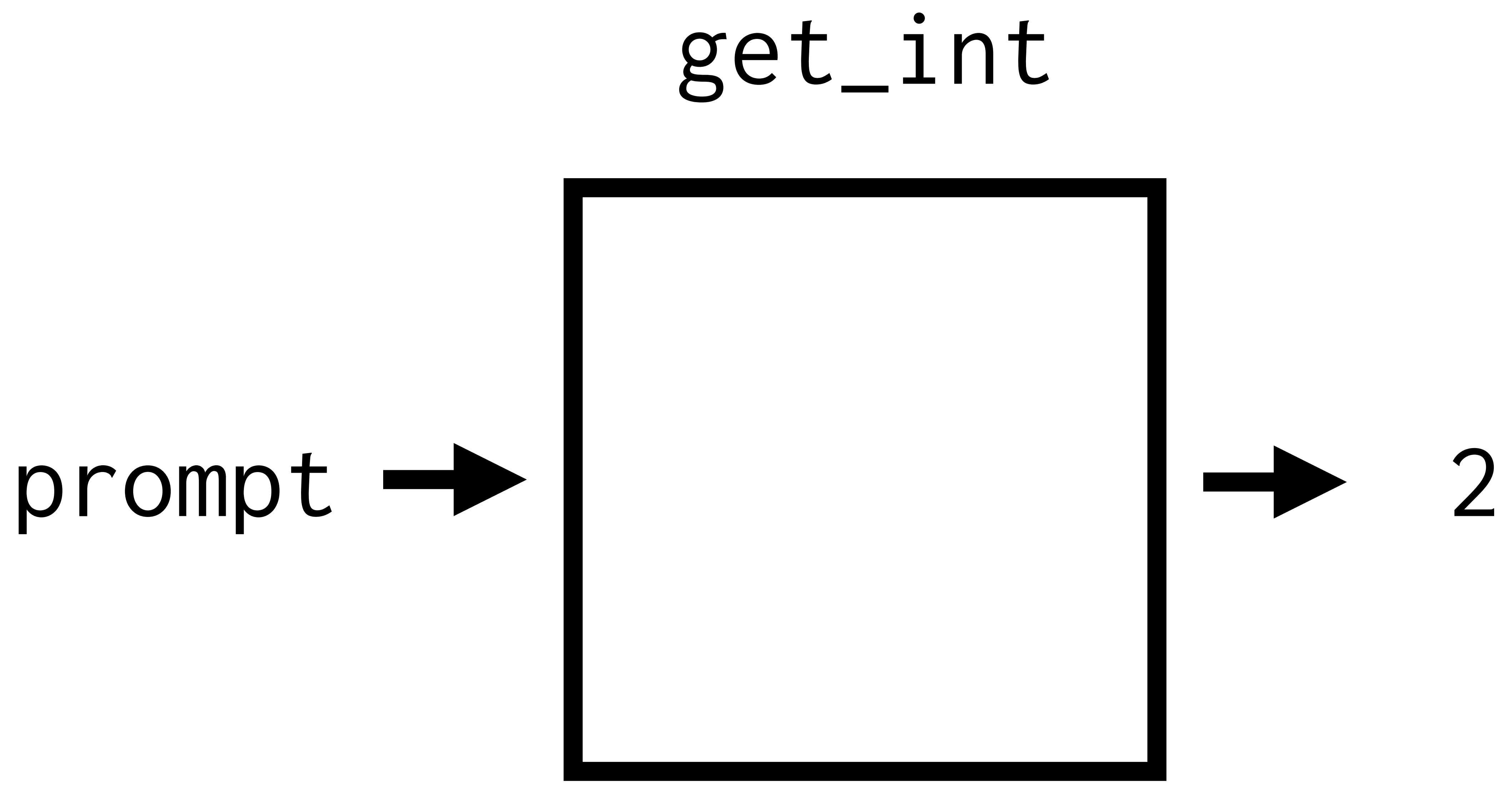
```
int get_int(string prompt)
{
    // Get int from user
}
```



```
int get_int(string prompt)
{
    // Get int from user
}
```



```
int get_int(string prompt)
{
    // Get int from user
}
```

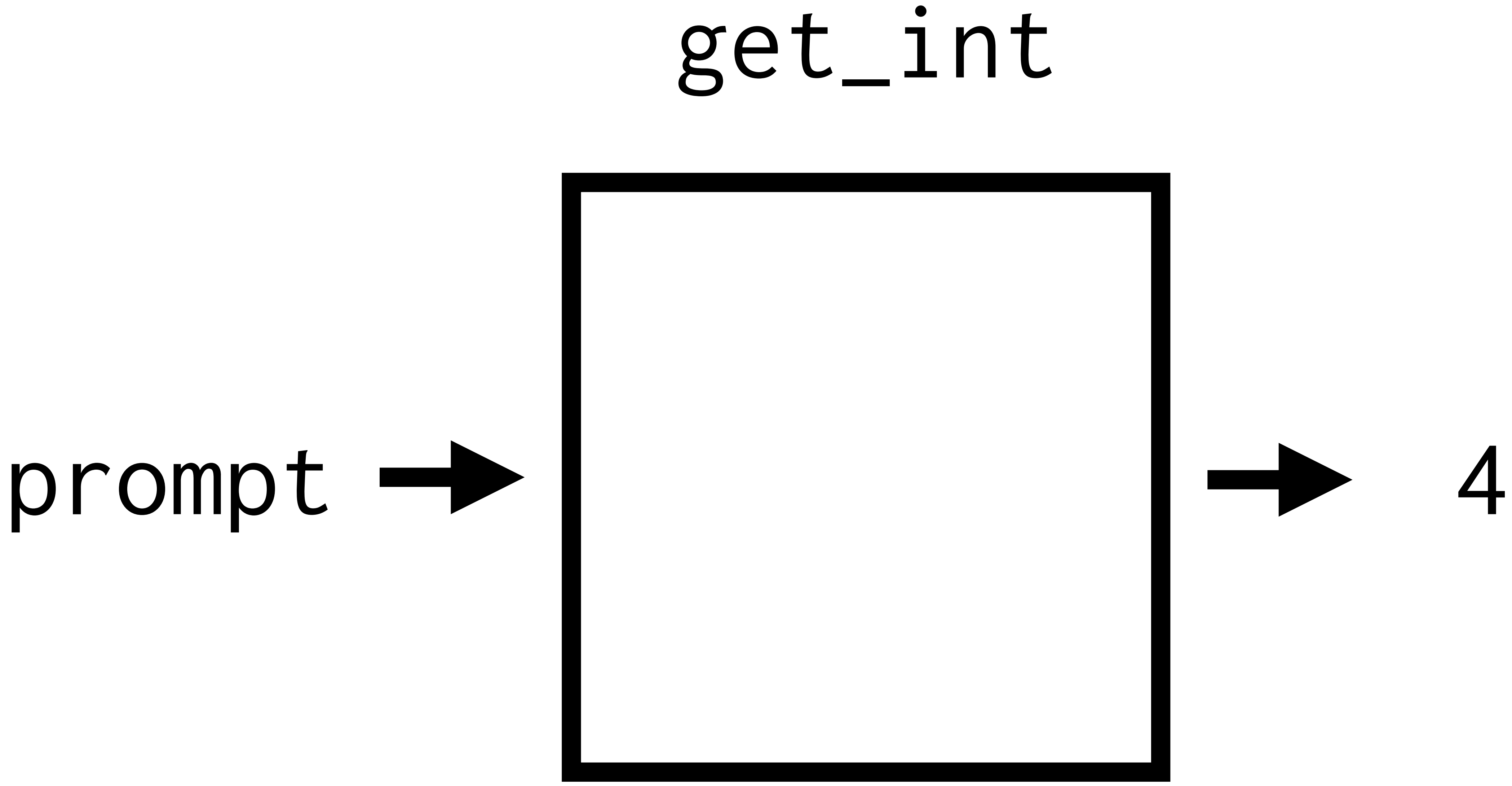


get_int

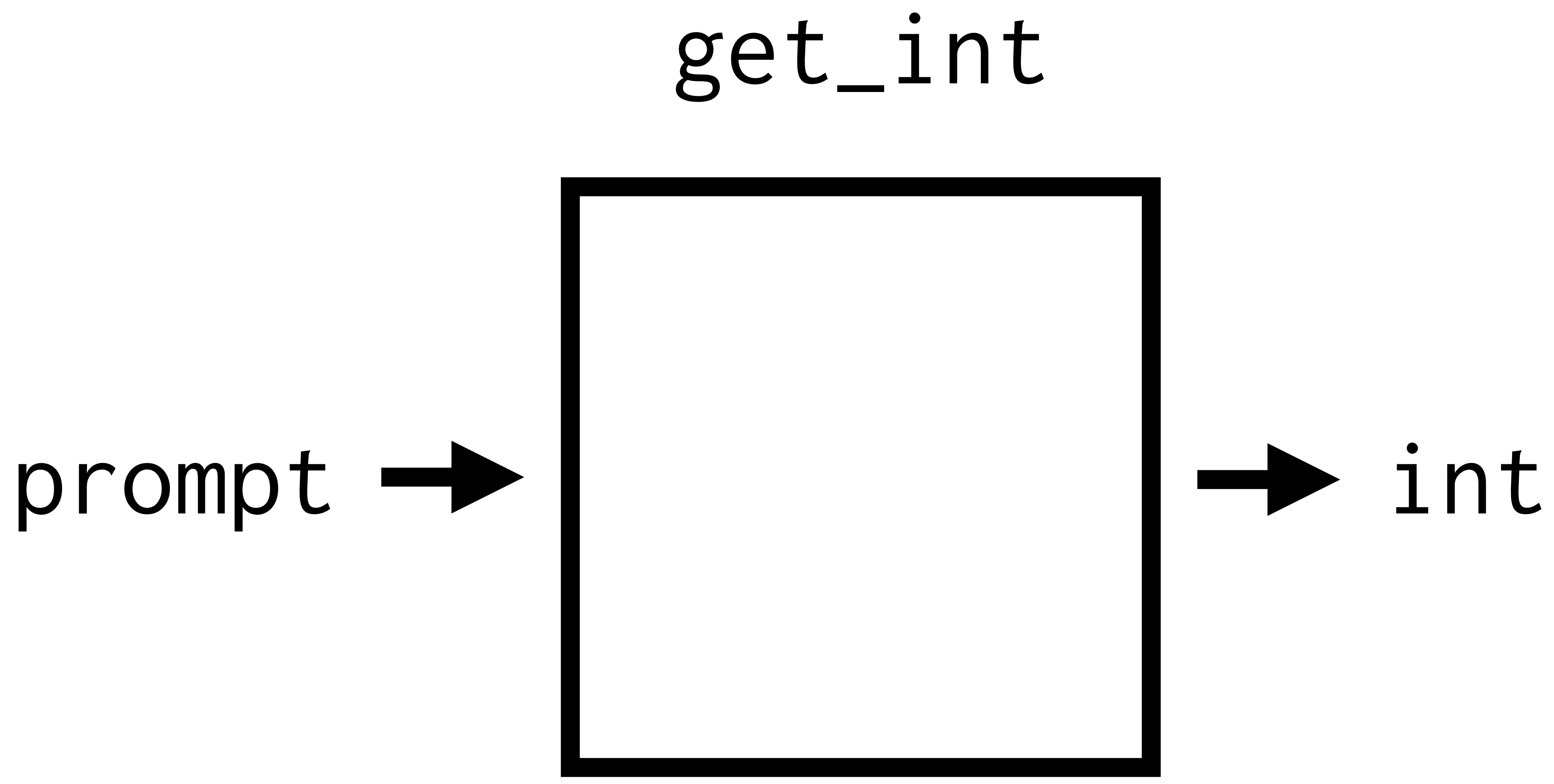
prompt

2

```
int get_int(string prompt)
{
    // Get int from user
}
```



```
int get_int(string prompt)
{
    // Get int from user
}
```



get_int

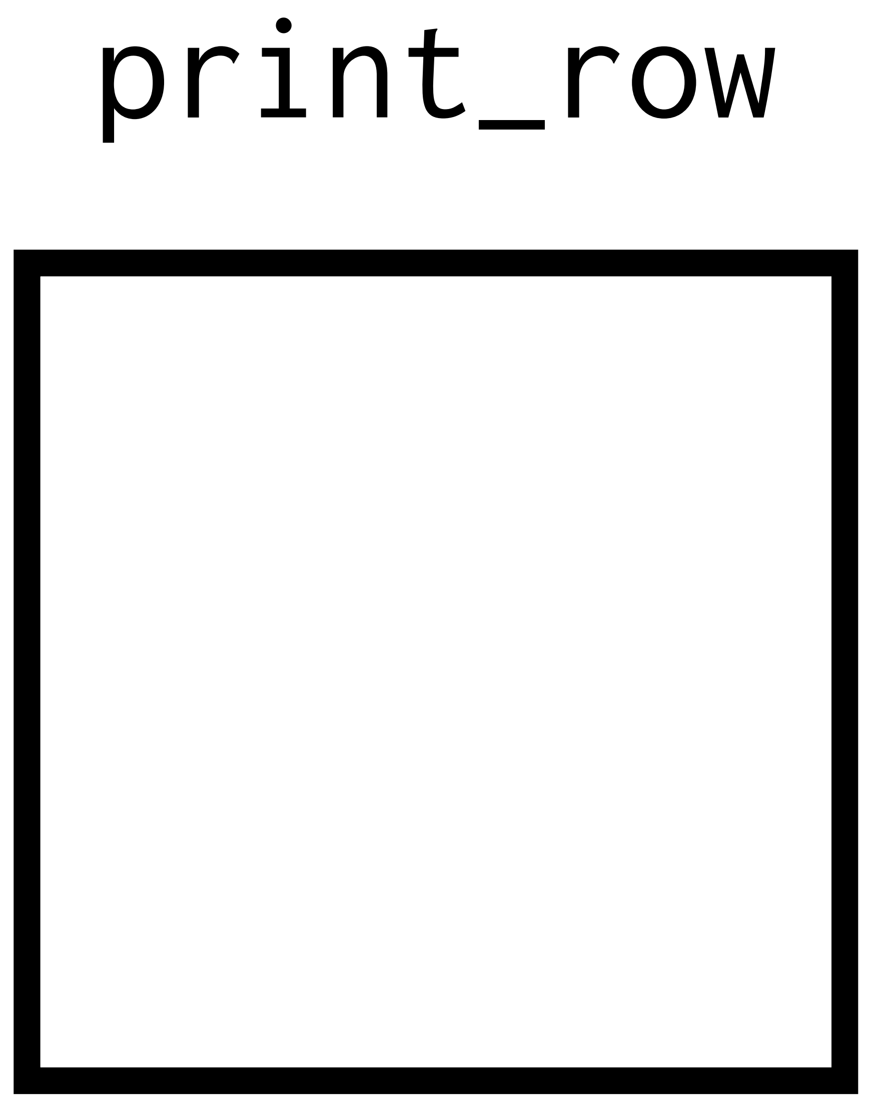
prompt

int

```
int calls = get_int("Calls: ");
```

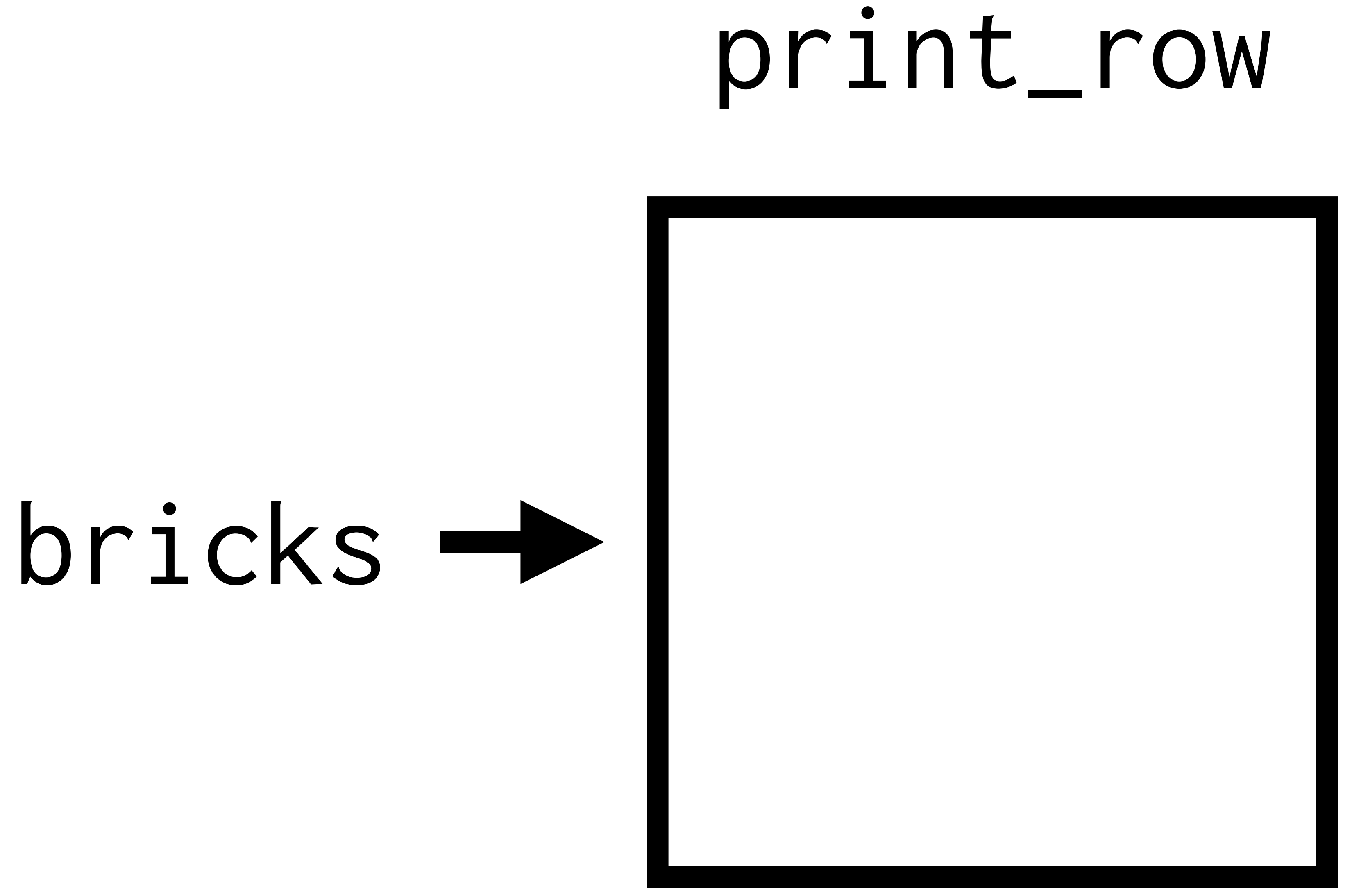
```
void print_row(int bricks)
{
    // Print row of bricks
}
```

```
void print_row(int bricks)
{
    // Print row of bricks
}
```

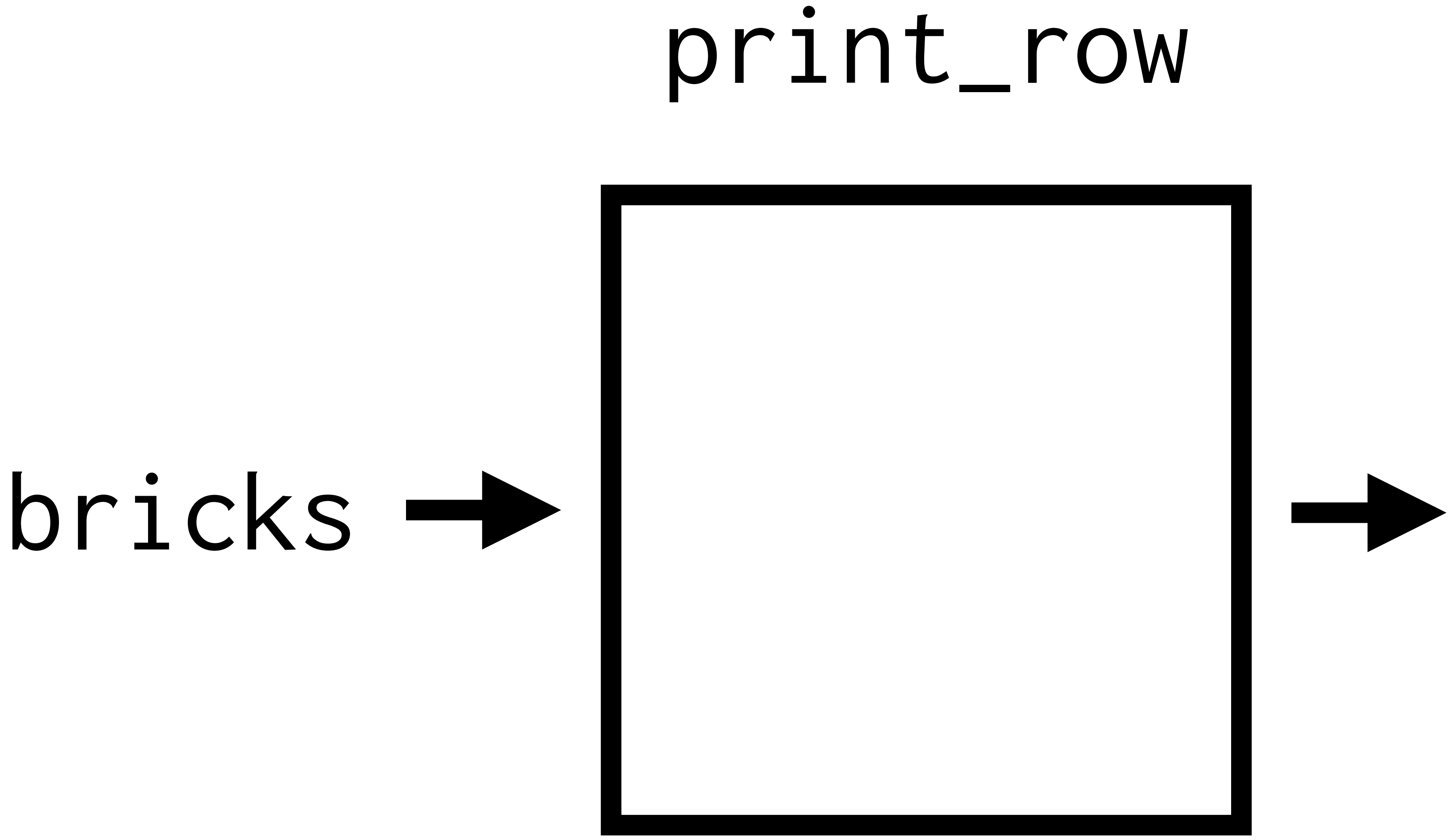


print_row

```
void print_row(int bricks)
{
    // Print row of bricks
}
```



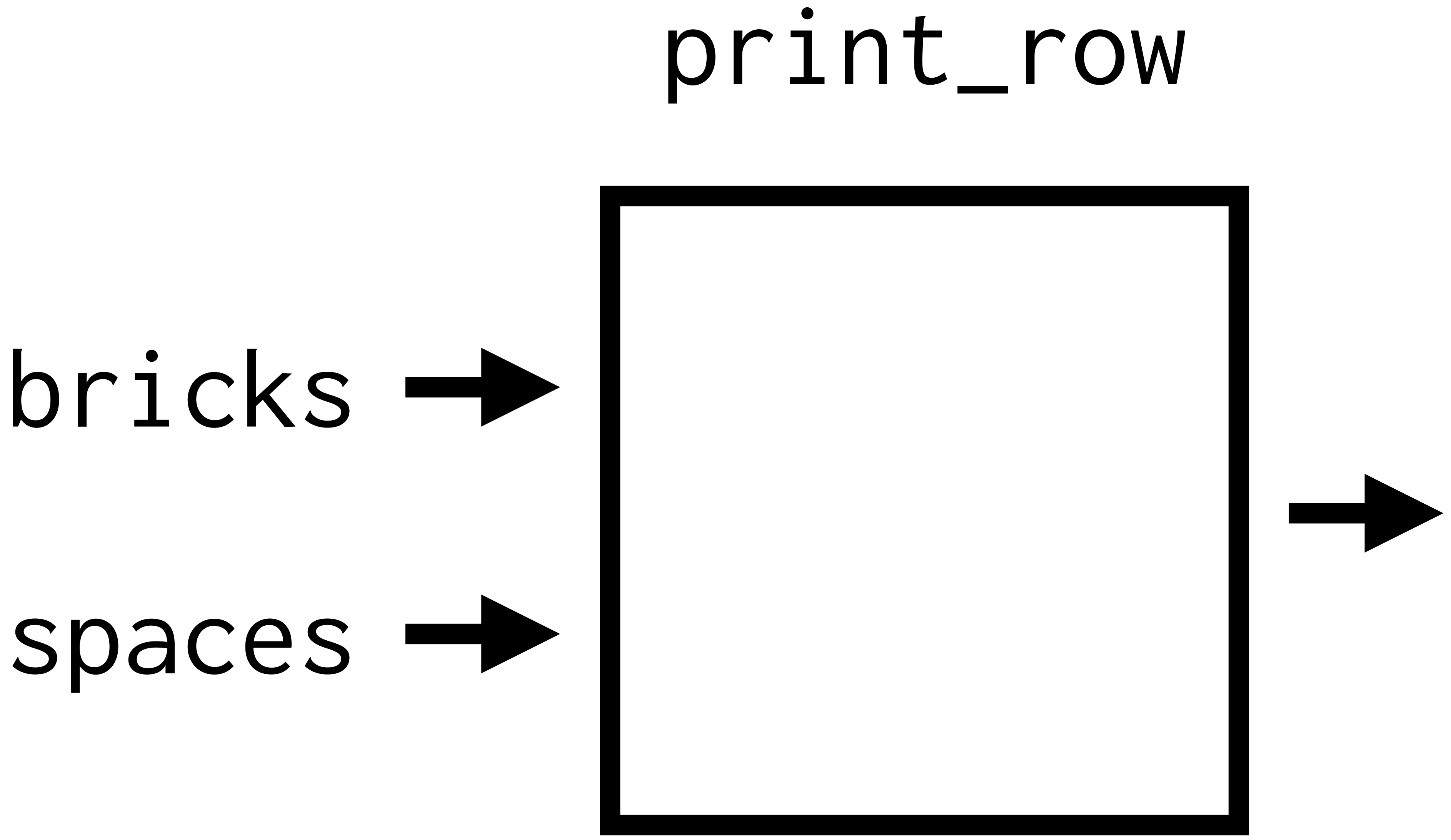
```
void print_row(int bricks)
{
    // Print row of bricks
}
```



```
print_row(4);
```

```
void print_row(int spaces, int bricks)
{
    // Print row of bricks
}
```

```
void print_row(int spaces, int bricks)
{
    // Print row of bricks
}
```



The week ahead

- **Submit Problem Set 1** by Sunday, February 5, 11:59 PM.
- Attend **office hours**.
- Complete `https://cs50.ly/studybuddy` to be paired with a classmate if you'd like!