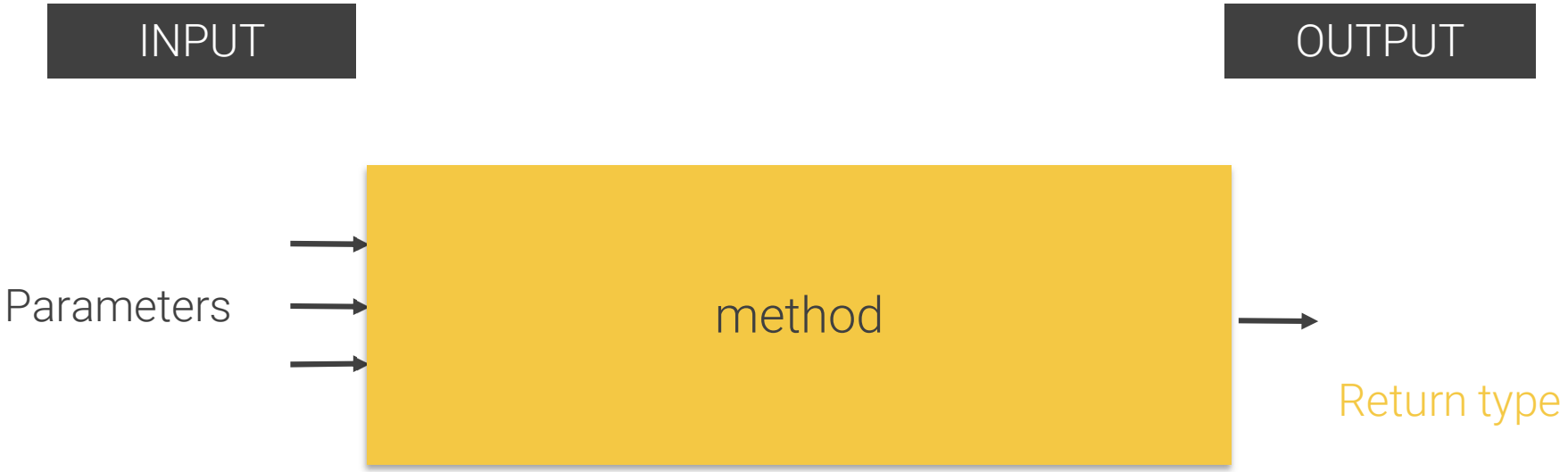
Method return value

1. Return value

Return value

1. Return value



To return a value you need:

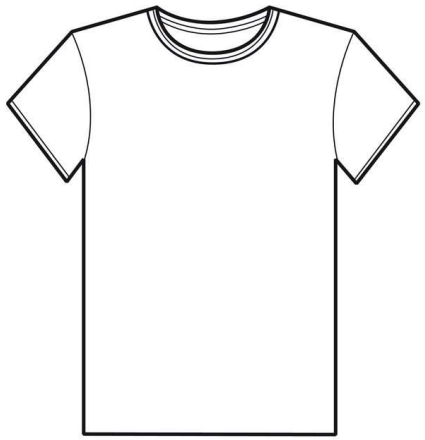
- A return type
- A return statement inside the block of code

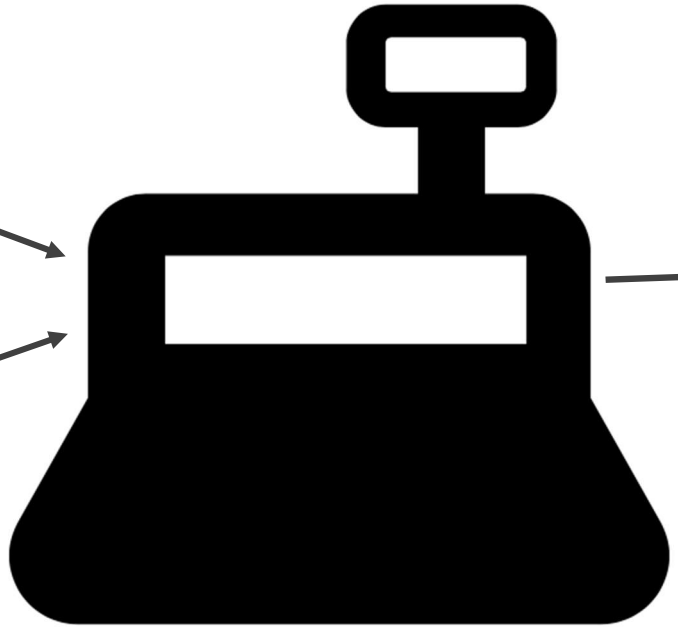
1. Return value

Calculate the correct change

eurosProvided 20€

price 9.90€

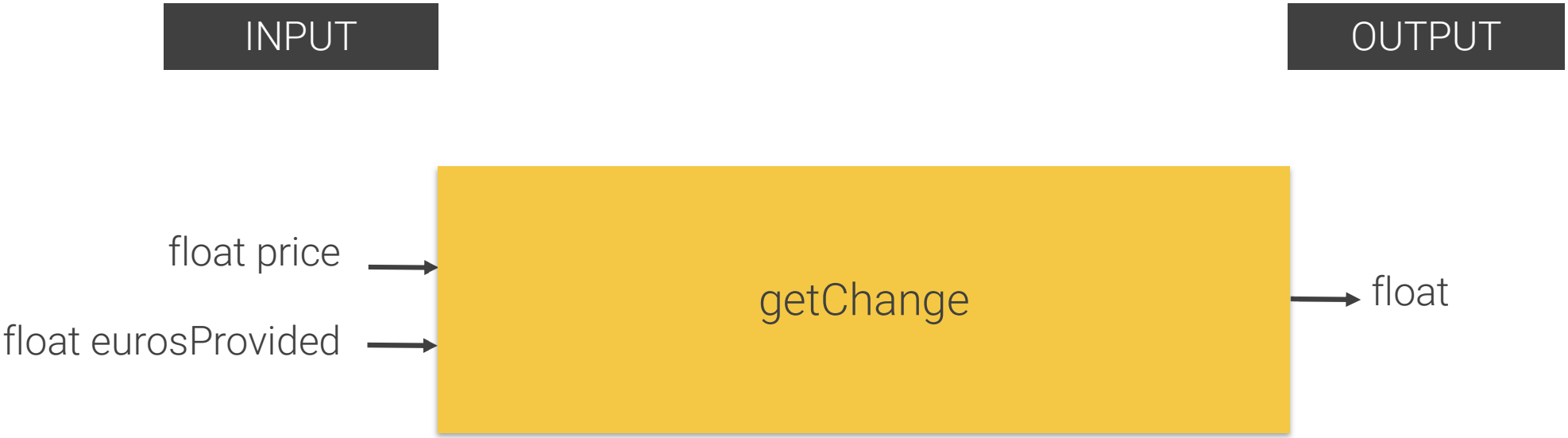




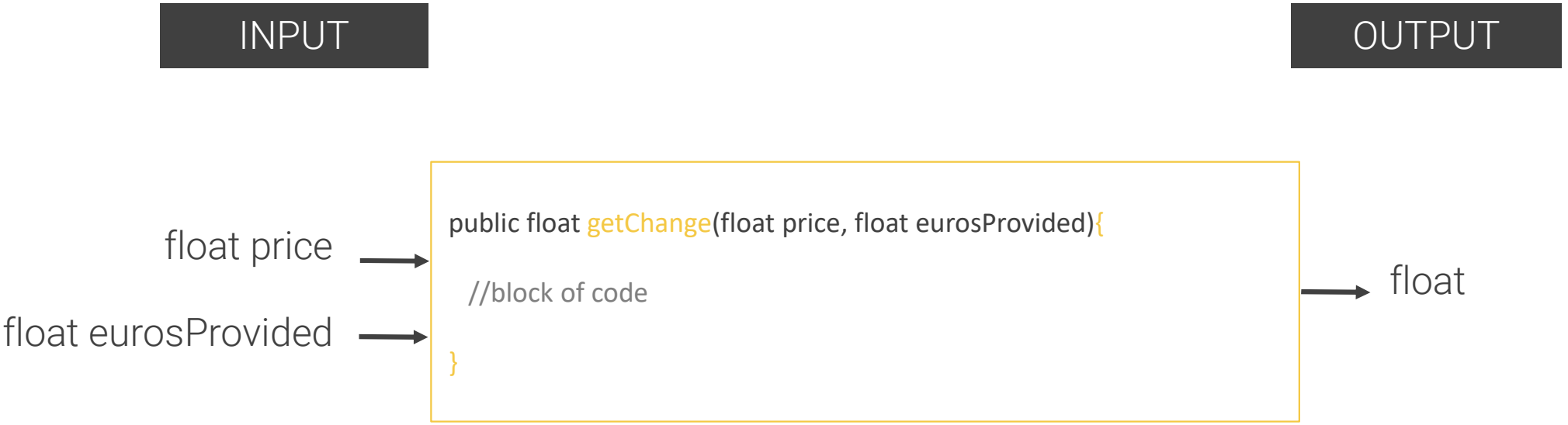
10.10€

eurosProvided-price

1. Return value



1. Return value



1. Return value

INPUT

OUTPUT

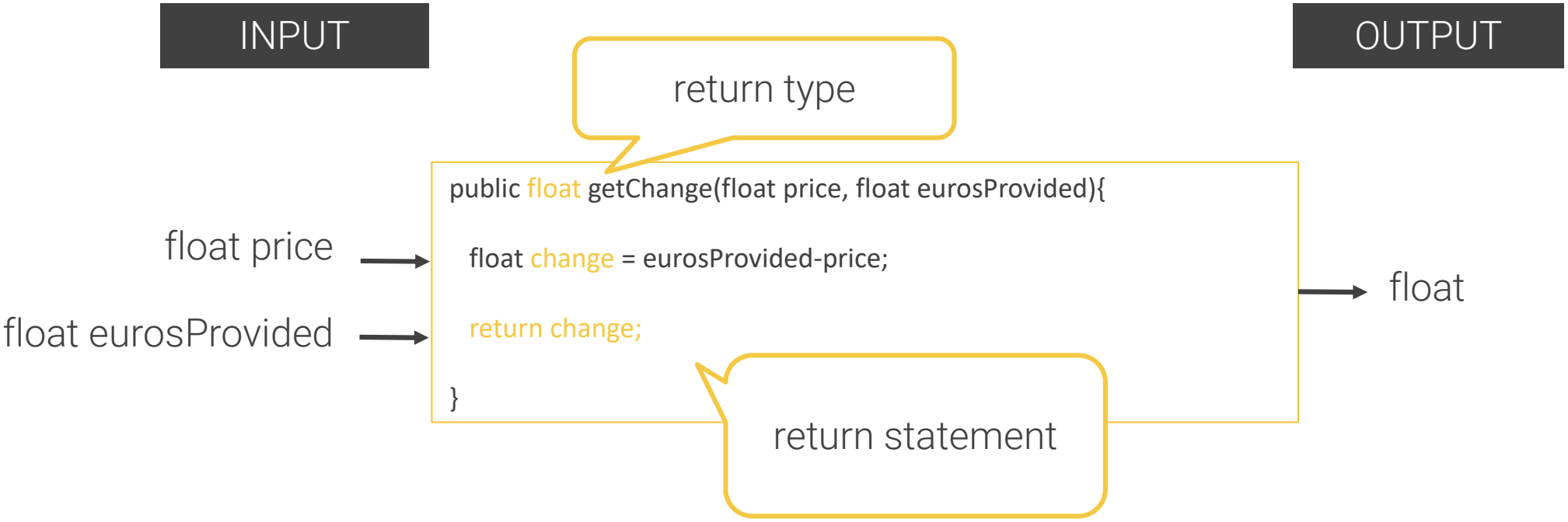
float price

float eurosProvided

```
public float getChange(float price, float eurosProvided){
    float change = eurosProvided-price;
    return change;
}
```

float

1. Return value



1. Return value

1. Return value

INPUT

OUTPUT

float price

float eurosProvided

```
public float getChange(float price, float eurosProvided){
   float change = eurosProvided-price;
   return change;
}
```

float

Method definition

Method calling

getChange(9.90, 20); → 10.10€

1. Return value

```
float returnedChange = getChange(9.90, 20); //10.10€
```

1. Return value

```
float returnedChange = getChange(9.90, 20); //10.10€
System.out.println("Your change: "+returnedChange);
```

1. Return value

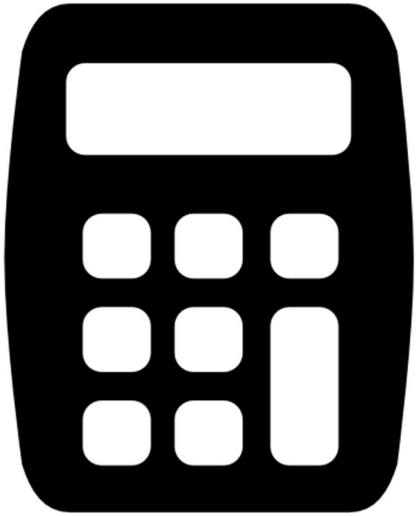
```
float returnedChange = getChange(9.90, 20); //10.10€
writeInDisplay(returnedChange);
```

1. Return value

Calculate the average temperature

4ºC, 3ºC, 6ºC, 8ºC, 10ºC

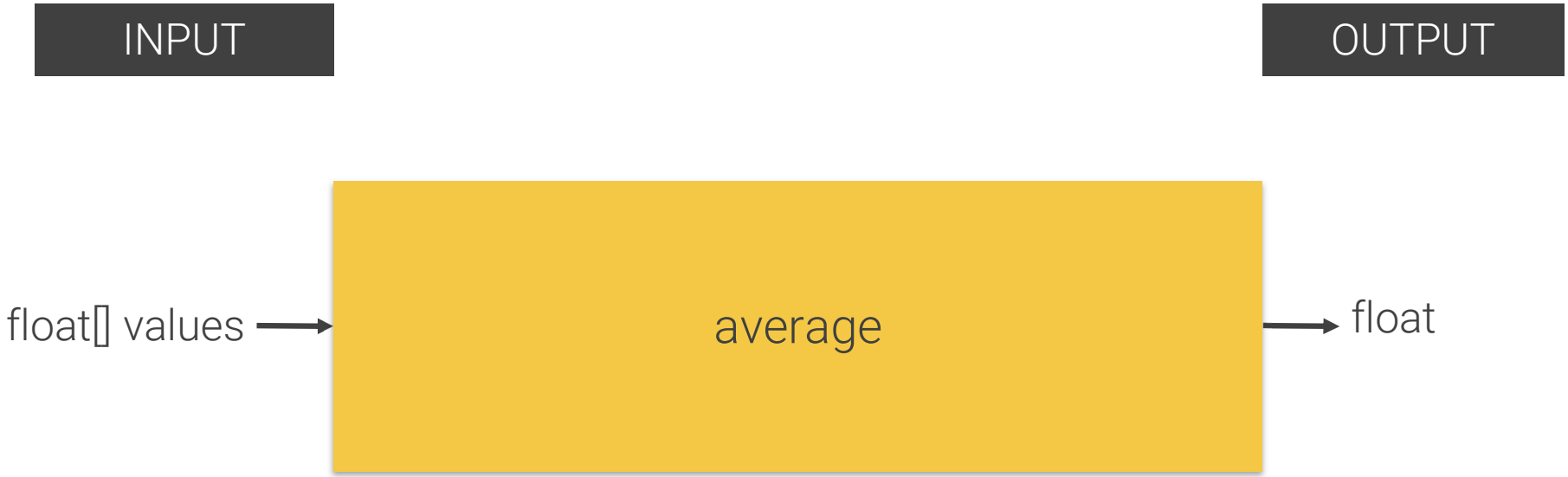
temperatures



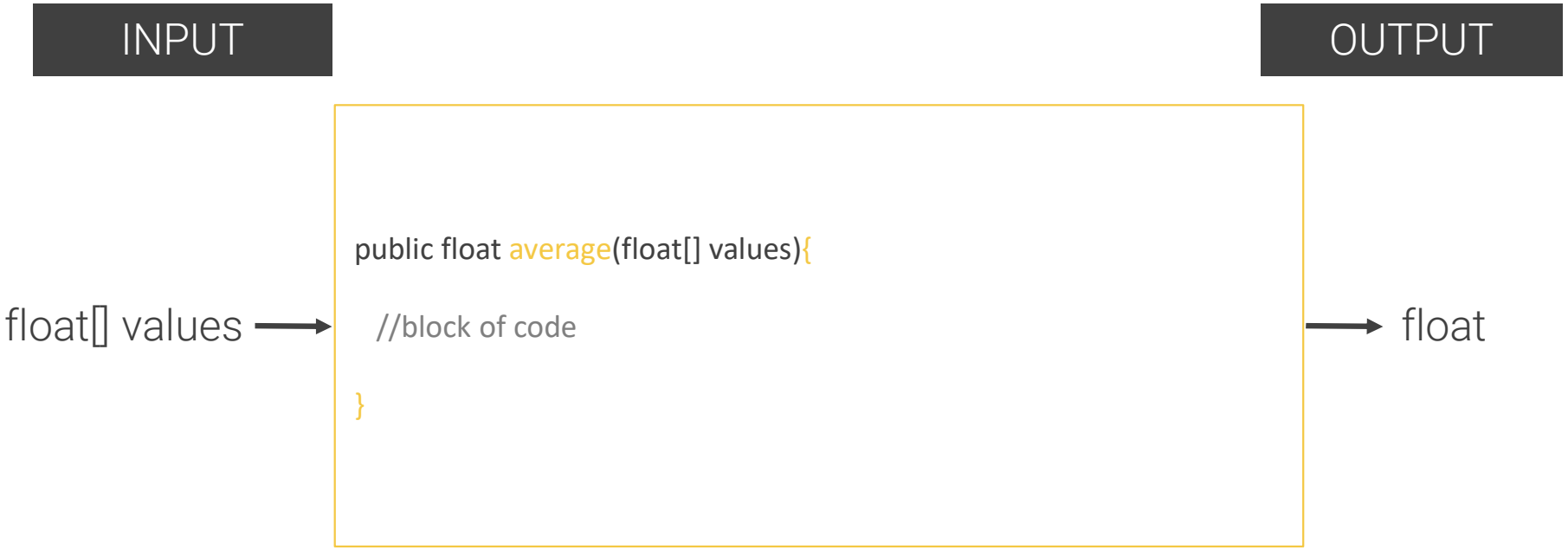
6.2ºC

sum(temperatures)/n

1. Return value



1. Return value



1. Return value

INPUT

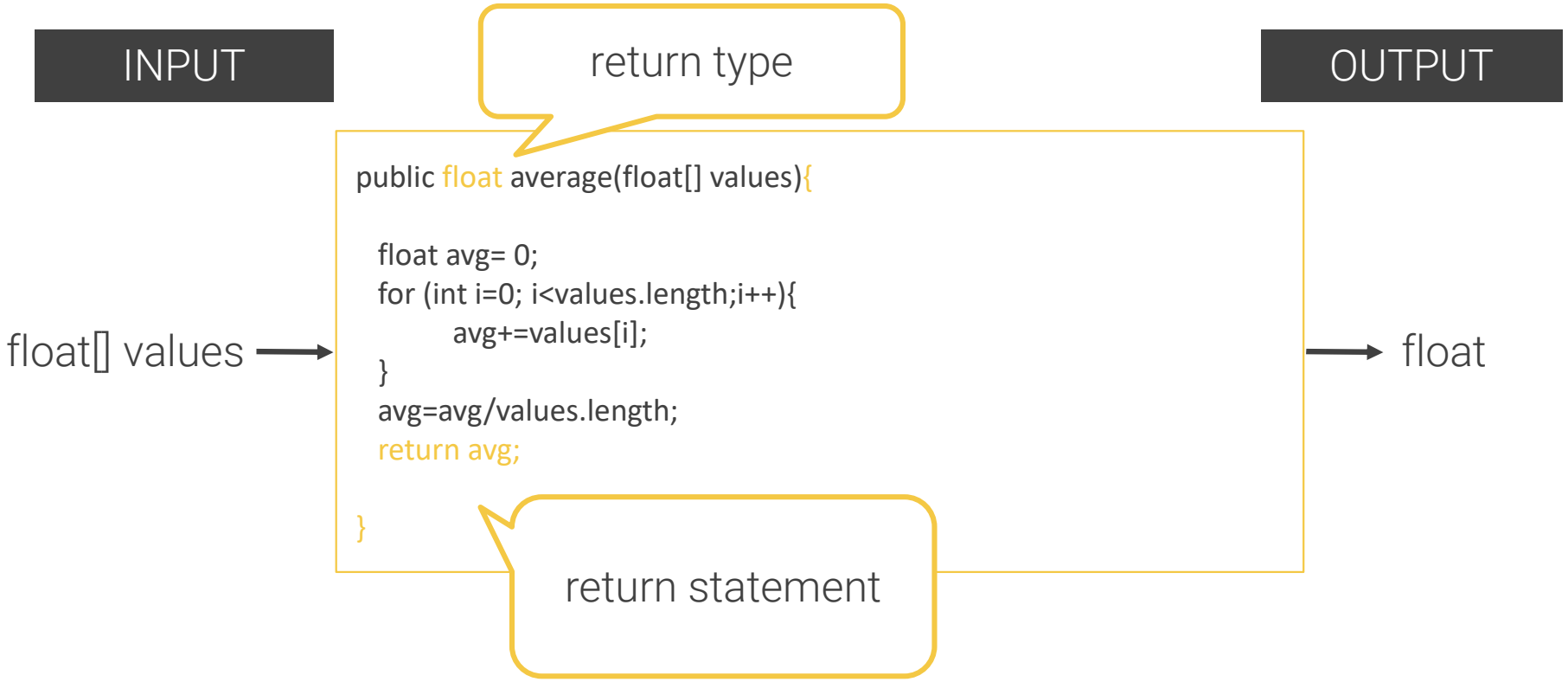
OUTPUT

float[] values →

```
public float average(float[] values){

   float avg= 0;
   for (int i=0; i<values.length;i++){
            avg+=values[i];
   }
   avg=avg/values.length;
   return avg;

}
```

→ float

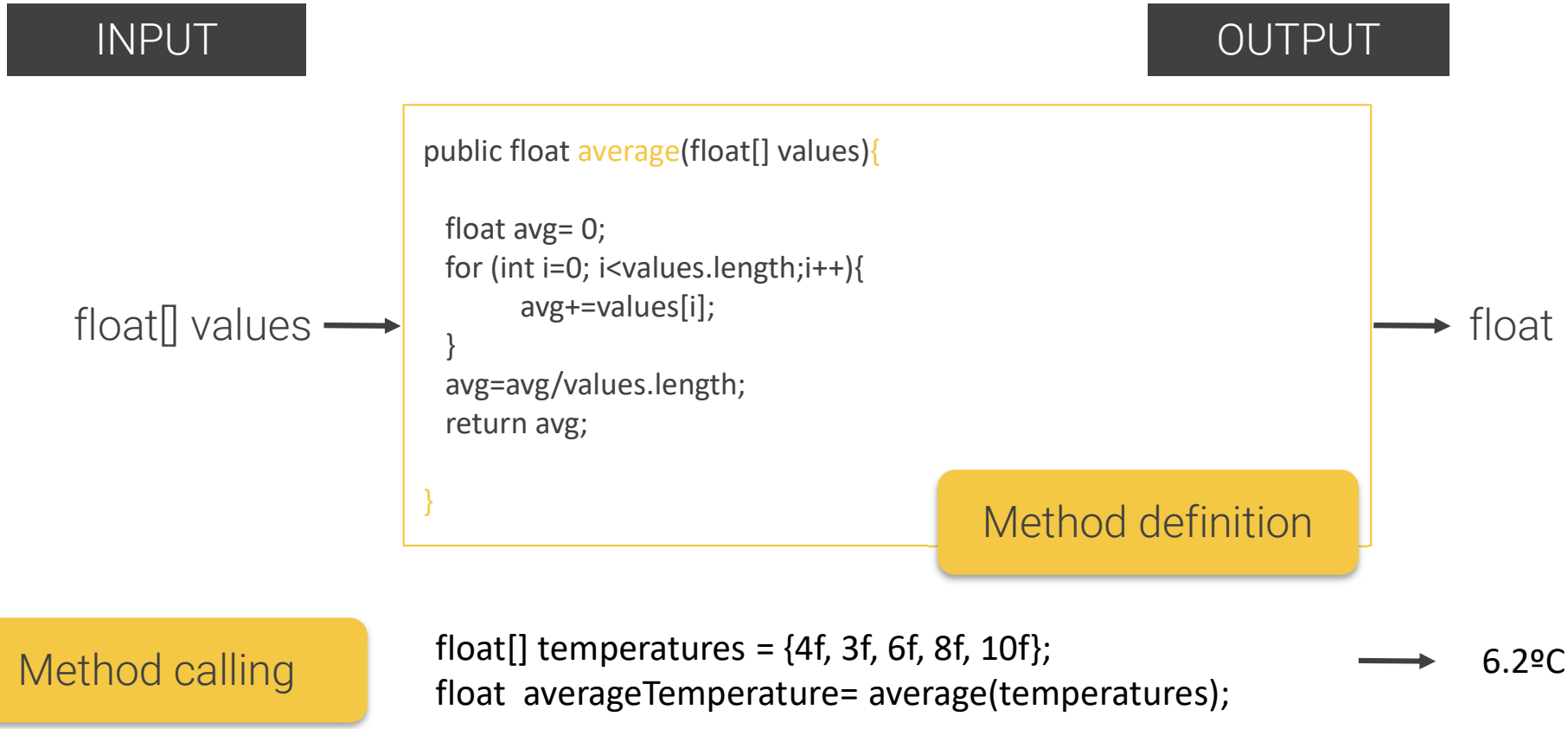
1. Return value



1. Return value

```
float[] temperatures = {4f, 3f, 6f, 8f, 10f};
float  averageTemperature= average(temperatures); //6.2ºC
```

1. Return value



1. Return value

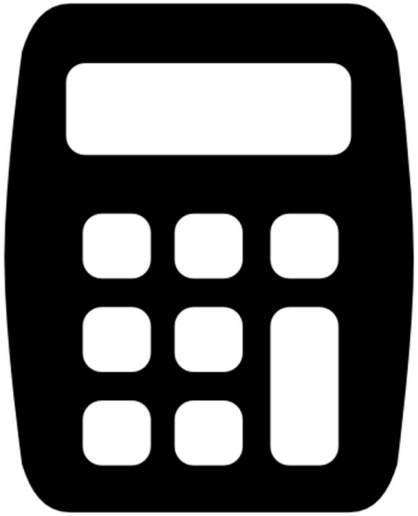
```
//Temperaturas en Burgos
float[] tempBurgos= {4f, 3f, 6f, 8f, 10f};
float  mediaTempBurgos= average(tempBurgos); //6.2ºC

//Temperaturas en Barcelona
float[] tempBarcelona= {10.5f, 11f, 11f, 11.5f};
float  mediaTempBarcelona= average(tempBarcelona); //11.0ºC
```

1. Return value

Calculate the average score

7, 7.5, 8, 8, 6, 6.5

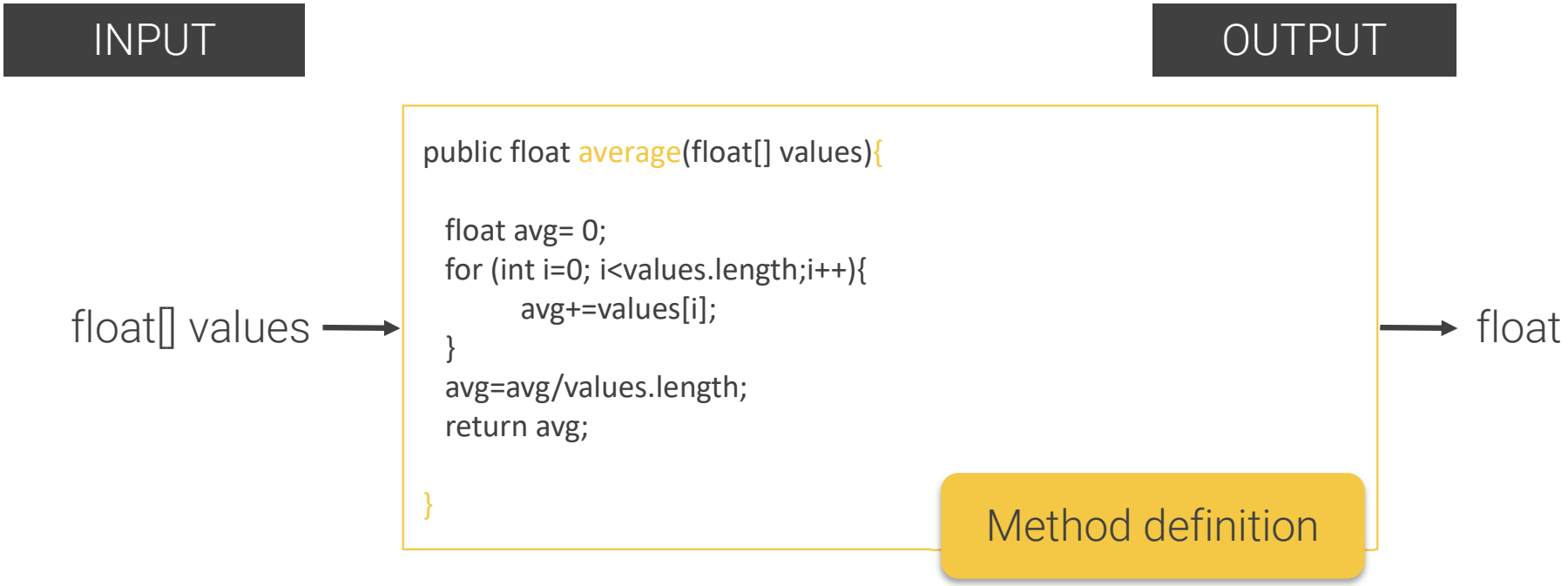
scores



7.2

sum(scores)/n

1. Return value



Method calling

```
float[] scores= {7f, 7.5f, 8.0f, 8.0f, 6.0f, 6.5f};
float  averageScore= average(scores);
```

1. Return value

Escribe en Java un método llamado valorAbsoluto, que reciba un número decimal.

- Si el número de entrada es positivo, lo devuelve tal cual
- Si el número de entrada es negativo, devuelve el número cambiado de signo

```
public ...
```

1. Return value

Escribe en Java un método llamado valorAbsoluto, que reciba un número decimal.

- Si el número de entrada es positivo, lo devuelve tal cual
- Si el número de entrada es negativo, devuelve el número cambiado de signo

```
public double valorAbsoluto(double x){
  //block of code
}
```

1. Return value

Escribe en Java un método llamado valorAbsoluto, que reciba un número decimal.

- Si el número de entrada es positivo, lo devuelve tal cual
- Si el número de entrada es negativo, devuelve el número cambiado de signo

```
public double valorAbsoluto(double x){
  if (x<0){
      return –x;
  }else{
      return x;
  }
}
```

“Never stop learning because life never stops teaching.“

Cambrige University Press