Pointer

A pointer in C++ is used to share a memory address among different contexts (primarily functions). They are used whenever a function needs to modify the content of a variable, but it does not have ownership.

In order to access the memory address of a variable, val, prepend it with $\&$ sign. For example, `&val` returns the memory address of val.

This memory address is assigned to a pointer and can be shared among functions. For example, $int^* p = \&val$ assigns the memory address of val to pointer p. To access the content of the memory pointed to, prepend the variable name with a `*`. For example, `*p` will return the value stored in val and any modification to it will be performed on val.

```
void increment(int *v) {
    (*v)++;
}

int main() {
    int a;
    scanf("%d", &a);
    increment(&a);
    printf("%d", a);
    return 0;
}
```

Function Description

Complete the *update* function in the editor below.

update has the following parameters:

- *int *a:* an integer
- *int *b:* an integer

Returns

- The function is declared with a `void` return type, so there is no value to return. Modify the values in memory so that a contains their sum and b contains their absoluted difference.
- $a' = a + b$
- $b' = |a - b|$

Input Format

Input will contain two integers, a and b, separated by a newline.

Sample Input

```
4
```

```
5
```

Sample Output

```
9
1
```

Explanation

- $a' = 4 + 5 = 9$
- $b' = |4 - 5| = 1$