Shortcut Operators

```
i += 1;     /*   i = i+1;   */
a /= 2;     /*   a = a/2;   */
x *= 5;     /*   x = x*5;   */
++i;        /*   i = i+1;   */
i++;        /*   i = i+1;   */
--i;        /*   i = i-1;   */
i--;        /*   i = i-1;   */
```

Shortcut Operators

```
i += 1;      /*   i = i+1;   */
a /= 2;      /*   a = a/2;   */
x *= 5;      /*   x = x*5;   */
++i;         /*   i = i+1;   */
i++;         /*   i = i+1;   */
--i;         /*   i = i-1;   */
i--;         /*   i = i-1;   */
```

Shortcut Operators

```
i += 1;     /*   i = i+1;   */
a /= 2;     /*   a = a/2;   */
x *= 5;     /*   x = x*5;   */
++i;        /*   i = i+1;   */
i++;        /*   i = i+1;   */
--i;        /*   i = i-1;   */
i--;        /*   i = i-1;   */
```

Shortcut Operators

```
i += 1;     /*   i = i+1;   */

a /= 2;     /*   a = a/2;   */

x *= 5;     /*   x = x*5;   */

++i;        /*   i = i+1;   */

i++;        /*   i = i+1;   */

--i;        /*   i = i-1;   */

i--;        /*   i = i-1;   */
```

Shortcut Operators

```
i += 1;     /*   i = i+1;   */

a /= 2;     /*   a = a/2;   */

x *= 5;     /*   x = x*5;   */

++i;        /*   i = i+1;   */

i++;        /*   i = i+1;   */

--i;        /*   i = i-1;   */

i--;        /*   i = i-1;   */
```

WARNING

```
b = ++i;
```

$$\neq$$

```
b = i++;
```

Pre-increment

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = ++y + 1;

printf("x

printf("y = %d\n", y);
```

update then evaluate

Output

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 4
```

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 4
```

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 4
y = 3
```

Pre-increment

```
int x, y = 2;

x = ++y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 4
y = 3
```

Pre-increment

```
int x, y = 2;

/* x = ++y + 1; */
y = y + 1;
x = y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Pre-increment

```
int x, y = 2;

/* x = ++y + 1; */
y = y + 1;
x = y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

update then evaluate

Post-increment

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = 1 + y++;

printf("x

printf("y = %d\n", y);
```

evaluate then update

Output

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 3
```

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 3
```

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 3
y = 3
```

Post-increment

```
int x, y = 2;

x = 1 + y++;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Output

```
x = 3
y = 3
```

Post-increment

```
int x, y = 2;

/* x = 1 + y++; */
x = 1 + y;
y = y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

Post-increment

```
int x, y = 2;

/* x = 1 + y++; */
x = 1 + y;
y = y + 1;

printf("x = %d\n", x);

printf("y = %d\n", y);
```

evaluate then update