Parameter Passing Mechanisms

2023. Spring

국민대학교 소프트웨어학부
최 준수

Arguments, Parameters

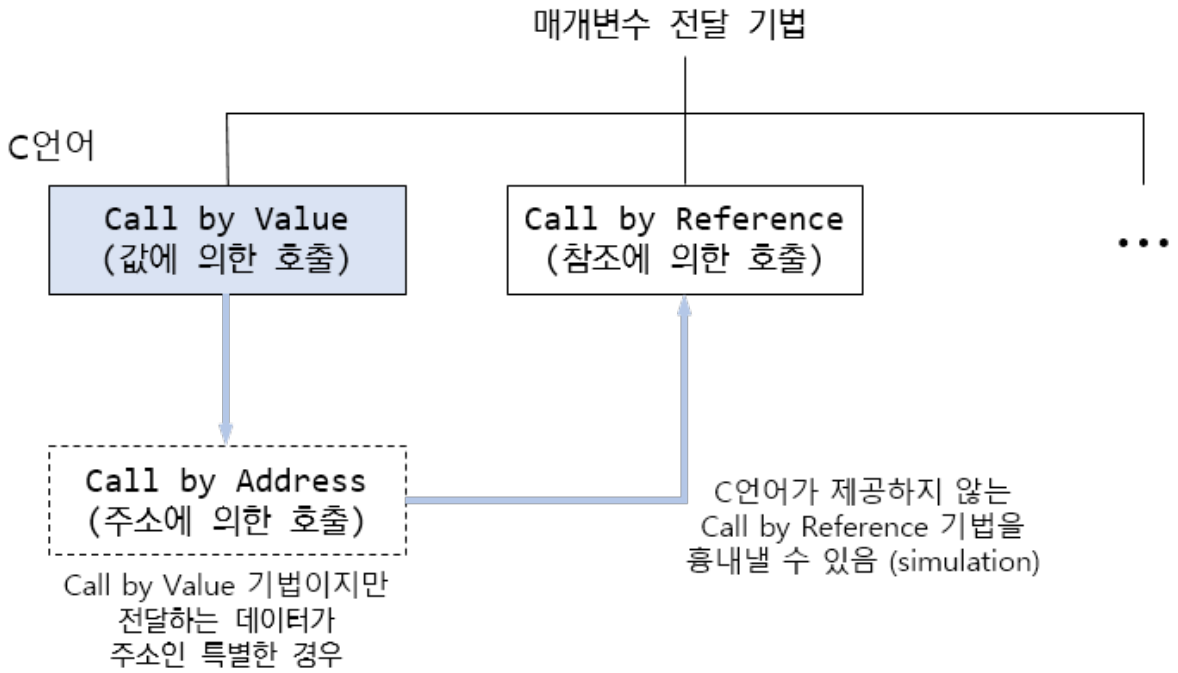
```
#include <iostream>
using namespace std;

int max(int x, int y)
{
    if( x >= y )
        return x;
    else
        return y;
}

int main()
{
    int a = 1, b = 2;

    cout << max(a, b);
    return 0;
}
```

x, y : parameter
(매개변수)

a, b : argument
(인자, 인수)

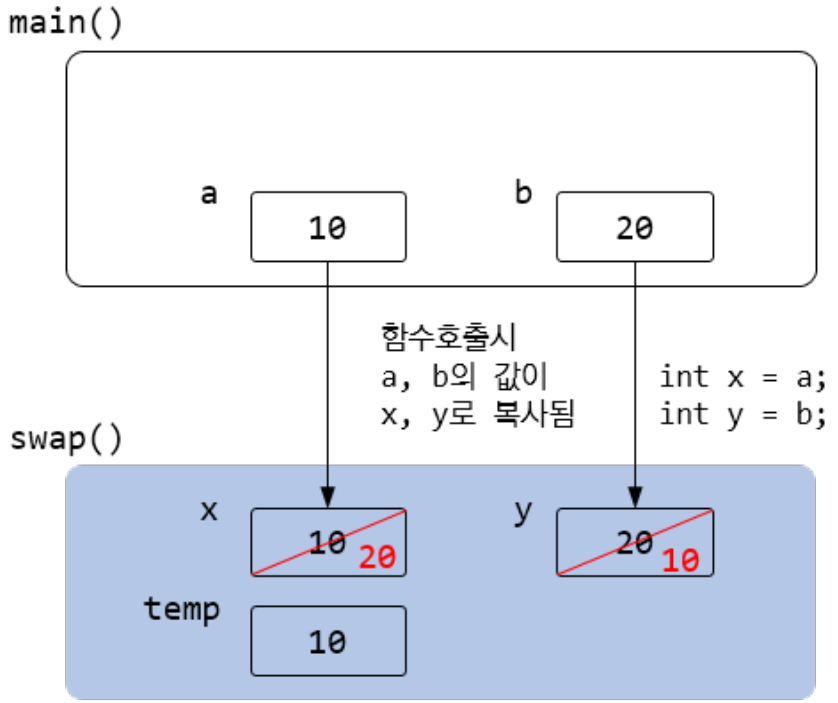
두 용어를 구분하여 사용할 때	parameter로 사용할 때	argument로 사용할 때
argument (인수, 인자, 전달인자)	actual parameter (실 매개변수)	actual argument (실 인자, 실 인수)
parameter (매개변수)	formal parameter (형식 매개변수)	formal argument (형식 인자, 형식 인수)

Parameter Passing Methods



Call By Value

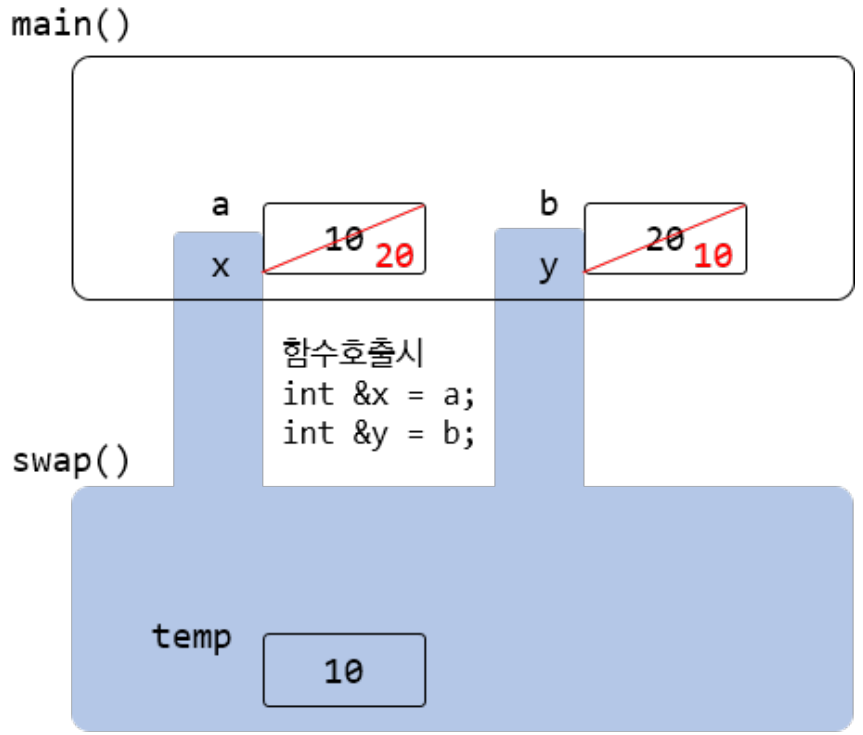
```
#include <iostream>
using namespace std;

void swap(int x, int y);  // call-by-value

int main()
{
    int a = 10;
    int b = 20;

    cout << "before swap: a = " << a << ", b = " << b << endl;
    swap(a, b);
    cout << "after  swap: a = " << a << ", b = " << b << endl;
    return 0;
}

void swap(int x, int y)
{
    int temp = x;
    x = y;
    y = temp;
    cout << "in     swap: x = " << x << ", y = " << y << endl;
}
```




```
before swap: a = 10, b = 20
in     swap: x = 20, y = 10
after  swap: a = 10, b = 20
```

Call By Reference

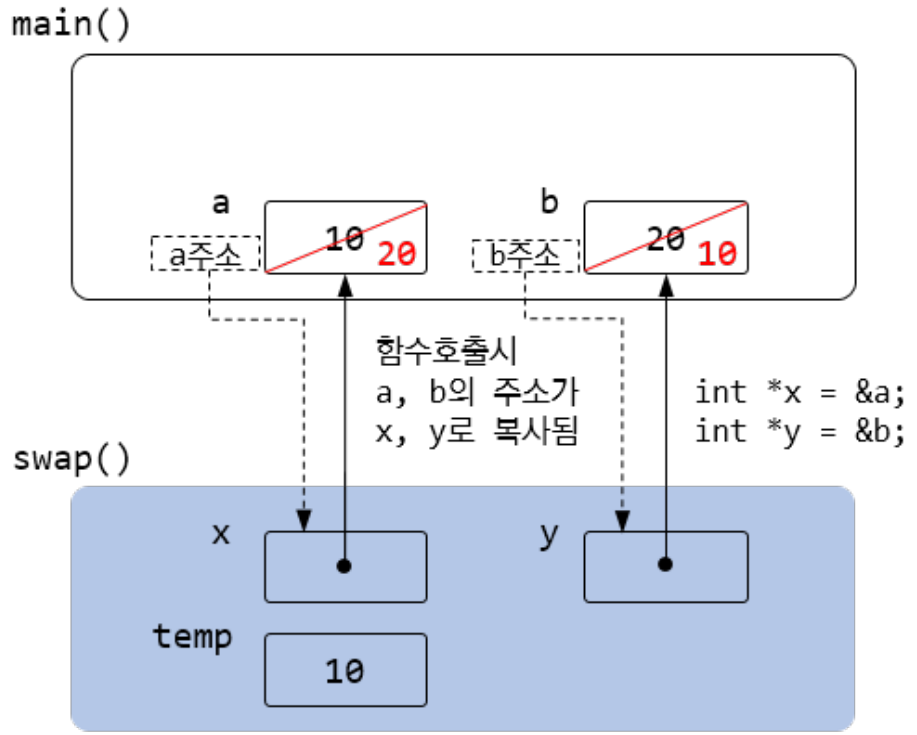
```
#include <iostream>
using namespace std;

void swap(int &x, int &y);  // call-by-reference

int main()
{
    int a = 10;
    int b = 20;

    cout << "before swap: a = " << a << ", b = " << b << endl;
    swap(a, b);
    cout << "after  swap: a = " << a << ", b = " << b << endl;
    return 0;
}

void swap(int &x, int &y)
{
    int temp = x;
    x = y;
    y = temp;
    cout << "in     swap: x = " << x << ", y = " << y << endl;
}
```




```
before swap: a = 10, b = 20
in     swap: x = 20, y = 10
after  swap: a = 20, b = 10
```

Call By Address

```
#include <iostream>
using namespace std;

void swap(int *x, int *y);  // call-by-address

int main()
{
    int a = 10;
    int b = 20;

    cout << "before swap: a = " << a << ", b = " << b << endl;
    swap(&a, &b);
    cout << "before swap: a = " << a << ", b = " << b << endl;
    return 0;
}

void swap(int *x, int *y)
{
    int temp = *x;
    *x = *y;
    *y = temp;
    cout << "in     swap:*x = " << a << ",*y = " << b << endl;
}
```




```
before swap: a = 10, b = 20
in     swap:*x = 20,*y = 10
after  swap: a = 20, b = 10
```