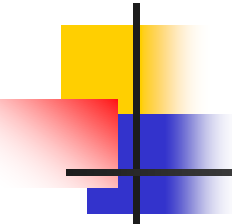


値渡しと参照渡しについて



引数における値渡しと参照渡し

```
int main()
{
    int a = 2, b = 3, c;
    c = add(a, b);
}
```

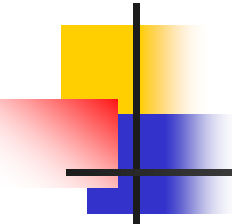
値渡し: a,bのコピーを渡す
参照渡し: a,b自体を渡す
値渡しも参照渡しも
a=2, b=3, c=5となる

引数が値渡しの場合

```
int add(int x, int y)
{
    return x + y;
}
```

引数が参照渡しの場合

```
int add(int& x, int& y)
{
    return x + y;
}
```



引数における値渡しと参照渡し

```
int main()
{
    int a = 2, b = 3, c;
    c = add(a, b);
}
```

add関数内で引数の値を変更した場合、参照渡しでは元の実引数に変更される。

値渡し: **a=2**, b=3, c=5

参照渡し: **a=5**, b=3, c=5

引数が値渡しの場合

```
int add(int x, int y)
{
    x += y;
    return x;
}
```

引数が参照渡しの場合

```
int add(int& x, int& y)
{
    x += y;
    return x;
}
```

値を返す関数と参照を返す関数

```
int main()
{
    int a;
    Hoge h;

    h.set(5);
    a = h.get();
}
```

値渡し: xのコピーを返す
参照渡し: x自体を返す
どちらもa=5

get()の戻り値が値渡し

```
class Hoge {
    int x;
public:
    void set(int a) {x = a;}
    int get() {return x;}
};
```

get()の戻り値が参照渡し

```
class Hoge {
    int x;
public:
    void set(int a) {x = a;}
    int& get() {return x;}
};
```

値を返す関数と参照を返す関数

```
int main()
{
    Hoge h;

    h.set(5);
    h.get() = 3;
}
```

値渡し: エラー (5 = 3;)
参照渡し: OK (h.x = 3;)
set関数と同じ使い方が
できる

```
get()の戻り値が値渡し
class Hoge {
    int x;
public:
    void set(int a) {x = a;}
    int get() {return x;}
};
```

```
get()の戻り値が参照渡し
class Hoge {
    int x;
public:
    void set(int a) {x = a;}
    int& get() {return x;}
};
```