

STEPS  
01/12/20

```
class X {
private: int a;
        char b;
        float c;
public: X (int s, char t, float w) {
    a = s;
    b = t;
    c = w;
}
```

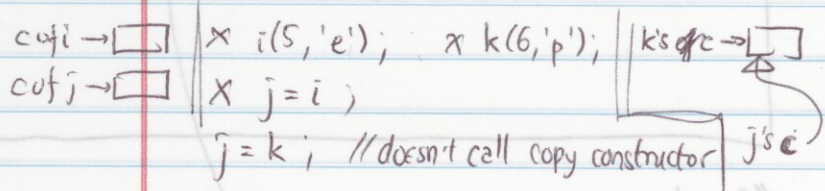
```
X i(5, 'c', 2.9);
X j = i; // create object by copying
// attributes from i;
X j(i);
```

**C++**: if copy constructor's not provided  
the compiler creates one for you  
→ do shallow copy;

prevent change, dereference name

```
X::X (const X &orig) {
    a = orig.a;
    b = orig.b;
    c = new float [a];
}
```

class name



class name

```
X & operator = (const X &orig) {
    if (this == orig) { return *this; } //check
    delete [] c;
    a = orig.a;
    b = orig.b;
    c = new float [a];
    for (int i = 0; i < a; i++) {
        c[i] = orig.c[i];
    }
    return *this;
}
```