

Chapter 8 Evaluation strategy

错误看法

错误理解一：值传递和引用传递，区分的条件是传递的内容，如果是个值，就是值传递。如果是个引用，就是引用传递。

错误理解二：JavaScript是引用传递。

错误理解三：传递的参数如果是普通类型，那就是值传递，如果是对象，那就是引用传递。

值传递 (pass by value) 是指在调用函数时将实际参数复制一份传递到函数中，这样在函数中如果对参数进行修改，将**不会影响到实际参数**。

引用传递 (pass by reference) 是指在调用函数时将实际参数的地址直接传递到函数中，那么在函数中对参数所进行的修改，将**影响到实际参数**。

所以，值传递和引用传递的区别并不是传递的内容。而是实参到底有没有被复制一份给形参。

所以说，JavaScript中其实还是**值传递**的，只不过对于对象参数，值的内容是对象的引用。

介绍

一般理论

按值传递

参数的值是调用者传递的对象**值的拷贝**(copy of value)，函数内部改变参数的值不会影响到外面的对象（该参数在外面的值）

按引用传递

另外一个众所周知的按引用传递接收的**不是值拷贝**，而是对象的**隐式引用**，如该对象在外部的直接引用地址。

该策略可以**更有效地传递复杂对象**，例如带有大批量属性的大结构对象。

按共享传递

该策略的要点是：函数接收的是**对象对于的拷贝**（副本），该引用拷贝和形参以及其值相关联。

地址拷贝？？？

按共享传递是按值传递的特例

按共享与指针

ECMAScript实现

现在我们知道了ECMAScript中将对象作为参数传递的策略了——**按共享传递**：**修改参数的属性将会影响到外部，而重新赋值将不会影响到外部对象**。但是，正如我们上面提到的，其中的ECMAScript开发人员一般都称之为是：按值传递，只不过该值是引用地址的拷贝。

总结

无论是值传递还是引用传递，其实都是一种求值策略(**Evaluation strategy**)。在求值策略中，还有一种叫做**按共享传递**(call by sharing)。其实JavaScript中的参数传递严格意义上说应该是**按共享传递**。

按共享传递，是指在调用函数时，传递给函数的是实参的地址的拷贝（如果实参在栈中，则直接拷贝该值）。在函数内部对参数进行操作时，需要先拷贝的地址找到具体的值，再进行操作。如果该值在栈中，那么因为是直接拷贝的值，所以函数内部对参数进行操作不会对外部变量产生影响。如果原来拷贝的是原值在堆中的地址，那么需要先根据该地址找到堆中对应的位置，再进行操作。因为传递的是地址的拷贝所以函数内对值的操作对外部变量是可见的。

简单点说，JavaScript中的传递是值传递，而这个**值**实际上是对对象的引用。

而按共享传递其实只是按值传递的一个特例罢了。所以我们可以说JavaScript的传递是按共享传递，或者说JavaScript中的传递是值传递。

