Attit 现代编程语言重要特性

强类型机制 ，但是只有系统类型，没有用户类型。。比如user等。。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 垃圾回收 | √ | × | × | √ | √ | √ | × | √ |
| 多重继承 |  |  |  |  |  |  |  |  |

"强类型机制"是指在声明变量的时候，语言的规范强制程序员编程的程度。在"强类型机制"类型的语言中(大部分现代语言都支持强类型机制)，每一个数据的取值和使用的对象都必须从属于对上下文来说合适的类型。静态类型语言在编译时使用类型检查强制类型转换。动态类型语言在运行时检查类型，

多重继承最小化代码复制的数量，进而减少可能引起维护问题的不一致性，所以很重要。在静态类型语言中，多重继承可能允许新的类主动位于所有超类之下，这减少了系统其他部分需要的特例编程的数量。

Go 语言最主要的特性：

* 自动垃圾回收
* 更丰富的内置类型
* 函数多返回值
* 错误处理
* 匿名函数和闭包
* 类型和接口
* 并发编程
* 反射
* 语言交互性