# **拓展知识丨编译vs解释**

来源：谢老师

因为电子电路元件的状态只有开/关两种状态，只能对应0/1两种表现方式。

所以自从计算机诞生以来，它就与二进制结下了不解之缘。

我们所输入的一切指令、数据，最终也需要转换为计算机所能解释、处理的信号------****二进制机器代码****（0101111011…）来进行运算等操作。

因此，我们所熟悉的编程语言，以及我们通过编程语言所编写的代码，最终都将通过某种（些）方式变成二进制机器代码，再交由计算机执行。而这些方法，大体上可以分为两种思路：****编译和解释****。

****编译，是指程序员编写好源代码后，交由一个名为编译器的软件进行编译，最终使得源代码全部翻译成机器代码，再交由计算机一次性执行。****

****解释，是指通过一个名为解释器的软件一句一句地把编写好的代码解释为机器代码来执行。****

所以我们可以直观地看出编译型和解释型语言的第一点重要差别：

编译型语言程序的执行速度要快于解释型语言。因为它的翻译过程在编译阶段就全部完成了。

除此以外，编译与解释还有许多深层区别。

比如说，解释型语言的解释器对它进行解释方法是唯一的。在不同的机器上（机器代码不同的计算机），我们可以使用解释器运行相同的代码。因此解释型语言的可移植性良好。

而编译型则需要针对不同的计算机平台（如 arm架构 和 英特尔架构）进行不同的编译过程，生成不同的机器代码才可以不同的平台运行。因此可移植性较差。

再比如说，解释型语言的代码是一句一句执行的，所以对变量类型的检查会随着程序的执行而动态地进行，这点保证了较好的动态性。

还有一点非常重要的差别，解释型语言在运行时，解释器占了非常多的系统资源，其效率因此不如编译好之后可以独立运行的编译型语言高。

有一种很特殊的语言，既是编译型，又可以视作解释型，它叫Java，有兴趣的同学可以去了解一下Java的运行机制哦！