



www.enjoylinux.cn

第十四课 优先级和类型转换分析



版权声明：本课件及其印刷物、视频的版权归成都国嵌信息技术有限公司所有，并保留所有权力：任何单位或个人未经成都国嵌信息技术有限公司书面授权，不得使用该课件及其印刷物、视频从事商业、教学活动。已经取得书面授权的，应在授权范围内使用，并注明“来源：国嵌”。违反上述声明者，我们将追究其法律责任。

优先级和类型转换分析

❖ 警示：易错的优先级

优先级问题	表达式	经常误认为的结果	实际结果
. 的优先级高于* ->操作符用于消除这个问题	*p. f	p 所指对象的字段 f (*p). f	对 p 取 f 偏移，作为指针，然后进行解除引用操作。*(p. f)
[] 高于*	int *ap[]	ap 是个指向 int 数组的指针 int (*ap) []	ap 是个元素为 int 指针的数组 int *(ap[])
函数() 高于*	int *fp()	fp 是个函数指针，所指函数返回 int。 int (*fp) ()	fp 是个函数，返回 int * int *(fp())
== 和 != 高于位操作	(val & mask != 0)	(val & mask) != 0	val & (mask != 0)
== 和 != 高于赋值符	c = getchar() != EOF	(c = getchar()) != EOF	c = (getchar() != EOF)
算术运算符高于位移运算符	msb << 4 + lsb	(msb << 4) + lsb	msb << (4 + lsb)
逗号运算符在所有运算符中优先级最低	i = 1, 2	i = (1, 2)	(i = 1), 2

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116

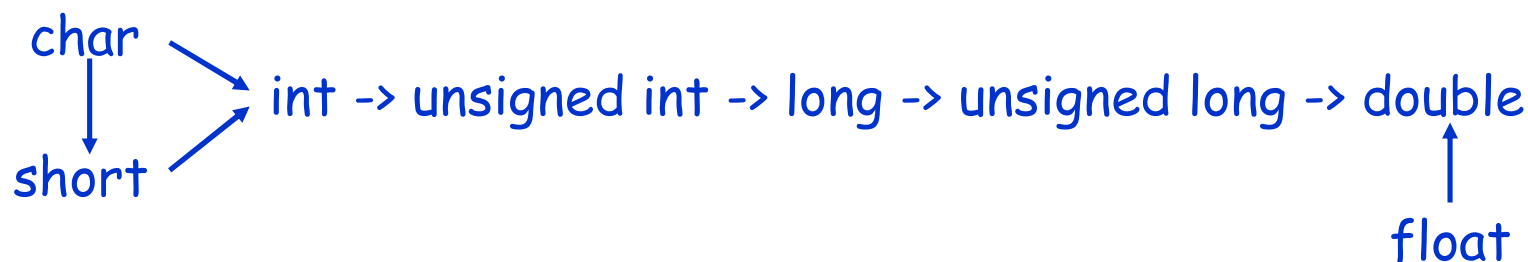


优先级和类型转换分析



❖ C语言隐式类型转换

- 算术运算式中，低类型转换为高类型
- 赋值表达式中，表达式的值转换为左边变量的类型
- 函数调用时，实参转换为形参的类型
- 函数返回值，**return**表达式转换为返回值类型



嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



实例分析7-1



www.enjoylinux.cn



编译的bug吗?

隐式类型转换带来困惑

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116

