

本章知识点小结

内容	描述	备注
关系运算符	> < >= <= == !=	< <= > >= 的优先级高于 == != 关系运算符的优先级低于算术运算符
逻辑运算符	&& !	除逻辑非为一元运算符之外, &&和 均为二元运算符, 并且!高于&&, &&高于
条件运算符	? :	三元运算符
if 形式的条件语句	if (表达式) 语句 A	用于单分支选择控制
if-else 形式的条件语句	if (表达式) 语句 1 else 语句 2	用于双分支选择控制
else-if 形式的条件语句	if (表达式 1) 语句 1 else if (表达式 2) 语句 2 ... else if (表达式 m) 语句 m else 语句 m+1	用于多分支选择控制
switch 语句	switch (表达式) { case 常量 1: 语句序列 1 case 常量 2: 语句序列 2 ... case 常量 n: 语句序列 n default: 语句序列 n+1 }	用于多分支选择控制

本章常见错误小结

常见错误实例	常见错误描述	错误类型
<pre>if (a > b); max = a;</pre>	在紧跟着 if 单分支选择语句的条件表达式的圆括号外之后写了一个分号。	运行时错误
<pre>if (a > b); max = a; else max = b;</pre>	在紧跟着 if-else 双分支选择语句的条件表达式的圆括号外之后写了一个分号。	编译错误
<pre>if (a > b) max = a; printf("max=%d\n",a);</pre>	在界定 if 语句后的复合语句时, 忘记了花括号。	运行时错误
<pre>if (a > b) max = a; printf("max=%d\n",a); else max = b; printf("max=%d\n",b);</pre>	在界定 if-else 语句后的复合语句时, 忘记了花括号。 由于 if 或 else 子句中只允许有一条语句, 因此需要多条语句时必须用复合语句, 即把需要执行的多条语句用一对花括号括起来。	编译错误
<pre>if (a = b) printf("a=b\n");</pre>	if 语句的条件表达式中, 表示相等条件时, 将关系运算符 == 误用为赋值运算符=。	运行时错误
	将关系运算符与相应的数学运算符混淆, 写成了≠, ≤, ≥	无法输入
<pre>if (a = = b) printf("a=b\n");</pre>	在关系运算符<=, >=, ==和!=的中间加入空格。	编译错误
<pre>if (a <= b) printf("max=%d\n",b);</pre>	将关系运算符!=, <=, >=的两个符号写反, 写成了=!, =<, =>。	编译错误
<pre>if (x == 1.1)</pre>	用==或者!=测试两个浮点数是否相等, 或者判断一个浮点数是否等于 0。	运行时错误
<pre>if ('A' <= ch <= 'Z')</pre>	误以为语法上合法的关系表达式在逻辑上一定是正确的。	运行时错误

<pre> switch (mark) { case 10: case 9: printf("A\n"); case 8: printf("B\n"); ... } </pre>	switch 语句中，需要每个 case 分支单独处理时，缺少 break 语句。	运行时错误
<pre> switch (mark) { case10: case9: printf("A\n"); break; case8: printf("B\n"); break; ... } </pre>	switch 语句中，case 和其后的数值常量中间缺少空格。	运行时错误
<pre> switch (mark) { case 100: case 90~100: printf("A\n"); break; case mark<90:printf("B\n"); break; ... } </pre>	switch 语句中，case 后的常量表达式用一个区间表示，或者出现了运算符（如关系运算符等）。	编译错误