**c语言网课记录（前为知识点，后附思维图）**

**一、初识c语言**

1.visual studio 2022

2. .c为源文件，.h为头文件

3.c语言中一定要有主函数：main函数。main是程序的入口，有且只能有一个，写法为int main()，int-整型类型，一般不用void main()。

4.printf为库函数，专门用来打印数据（访问需要#include<stdio.h>，std是标准，i和o分别是input和output）

5.运行快捷键：ctrl+f5或fn+ctrl+f5

6.return 0是指正常运行后返回一个约定俗成的“0”，如果异常运行则返回非零

7.计算机里的单位：

最小单位bit—比特位

1 byte—字节= 8 bit；1kb=1024 byte；1mb=1024kb；1gb=1024mb；1tb=1024gb；1pb=1024tb.

8.sizeof(long)≥sizeof(int)=4 byte

9.变量和常量：

变量：局部变量({}内部的变量)；全局变量({}外部的变量)，名称冲突时，局部优先

10.变量初始化int a = 0;

11.scanf函数报错，可在首行添加：

#define \_CRT\_SECURE\_NO\_WARNINGS

12.变量的作用域（变量在哪生效，哪就是他的作用域）

局部变量作用域：变量所在局部范围

全局变量作用域：整个工程

13.变量的生命周期：

局部变量的生命周期：进入作用域生命周期开始，出作用域生命周期结束

全局变量生命周期：整个程序的生命周期

14.常量：

字面常量：

const修饰的常变量：本质上是变量，但是不能被修改，有常量的属性

#define定义的标识符常量：

枚举常量：

15.c语言字符串和字符表示方法：”abc”; ’a’

16.%d打印有符号整型（包括正负），%s打印字符串，%c打印字符，%f打印浮点数，%lf打印double类型的数据，%zu打印sizeof的返回值，\n是换行的意思，%p是打印地址的格式，%u是打印无符号的整型（无负）

17.字符串：由双引号引起来的一串字符，其结束标志是一个\0的转义字符，但不算做字符串内容

18.转义字符：

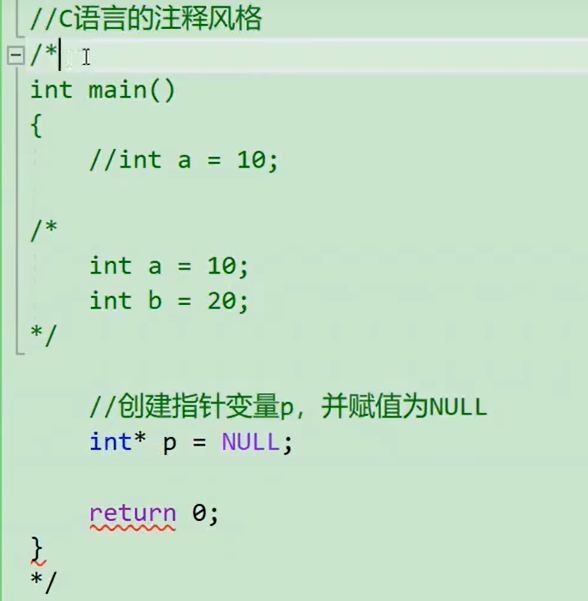


19.ASCII编码表：



20.空格也是字符

21.c语言中的注释方式：/\*要注释的内容\*/ 不支持嵌套注释，遇到一个\*/就会结束注释



C++的注释方式：//要注释的内容

22.语句

选择语句：if else，switch

循环语句：while，for，do...while

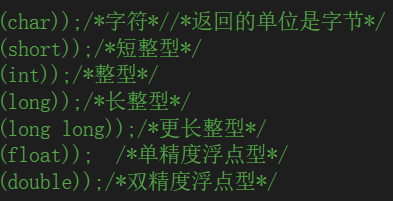
23.c语言是结构化的程序设计语言：顺序结构，选择结构，循环结构

24.函数



25.数组：一组相同类型元素的集合，从0开始给元素编号，称为数组的下标

26.c语言中的内置数据类型：



27：

0—数字0；

’0’—字符0，ASCII值为48；

‘\0’—字符，ASCII值为0；

EOF—end of file，为整型常量，值为-1；

28.操作符

算数操作符：+ - \* /（除，取商） %（取模，取余数）

移位操作符：>> <<

位操作符：& ^ |

赋值操作符：+= -= \*= /= &= %= |= ^= >>= <<=

单目操作符：只有一个操作数的操作符，a+b，所以其中的+是双目操作符



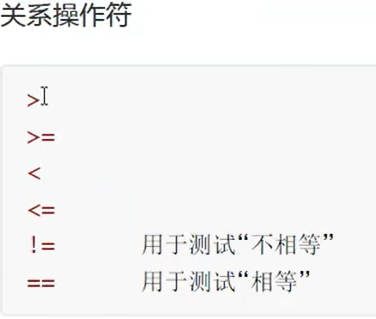
后置++：



前置++：先++，后使用

强制类型转换：（int），（char）

关系操作符：



逻辑操作符：

&&：逻辑与，并且

||：逻辑或，或者

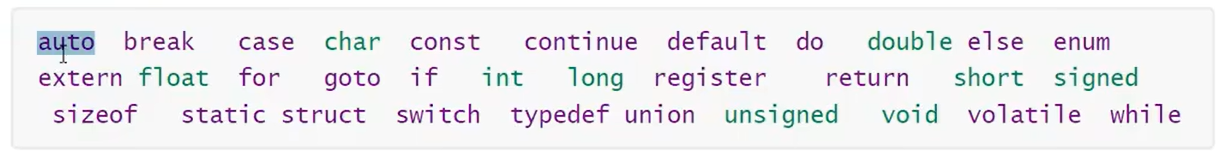
条件操作符：exp1？exp2：exp3 三目操作符。如果表达式一exp1为真，则执行表达式二exp2；如果表达式一exp1为假，则执行表达式三exp3.

逗号表达式：exp1，exp2，exp3，……，expN就是用逗号隔开的一串表达式，其特点是从左至右依次计算各表达式，整个表达式的结果是最后一个表达式的结果

下标引用操作符，函数调用操作符：



29.常见的关键字：



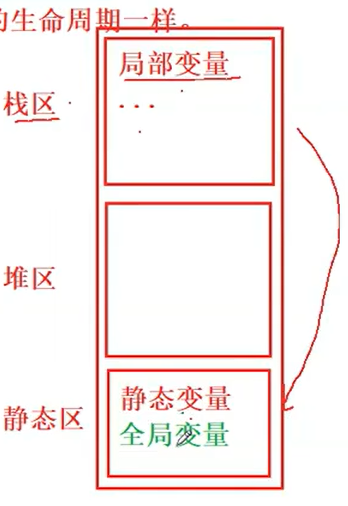
define不是关键字，它是预处理指令.

30.变量的命名：不能和上述关键字一样，必须是字母，数字或下划线组成，不能有特殊字符，不能以数字开头

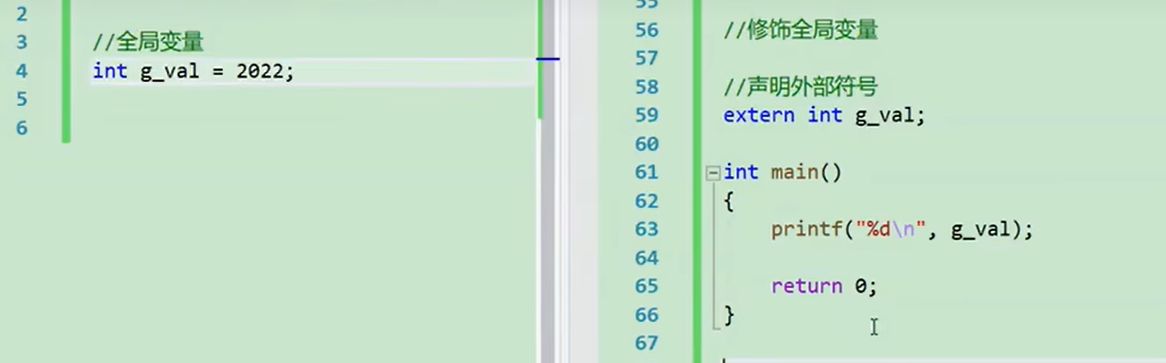
31.typedef：类型重定义

32.static:修饰变量和函数

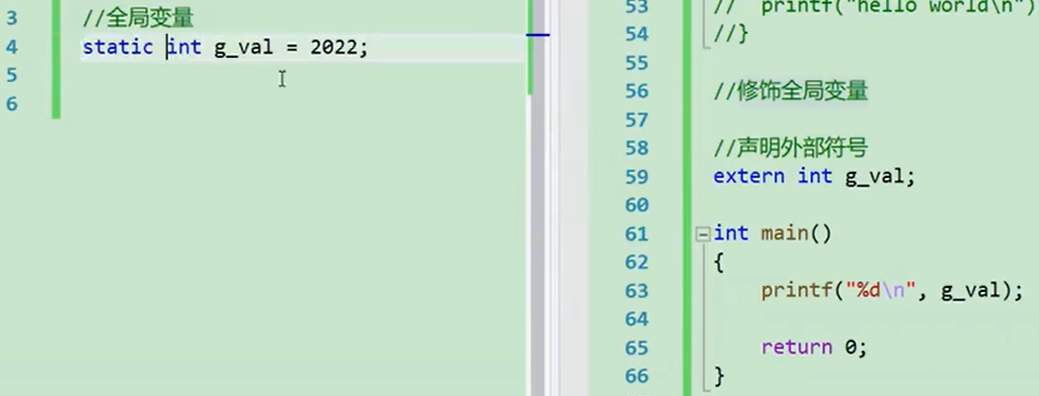
static修饰局部变量：static修饰局部变量的时候，局部变量出了作用域，变量不销毁，本质上改变了变量的存储位置（栈区，堆区，静态区），从而影响了变量的生命周期，使其生命周期变长，和整个程序的生命周期一样。



static修饰全局变量：



可正常运行

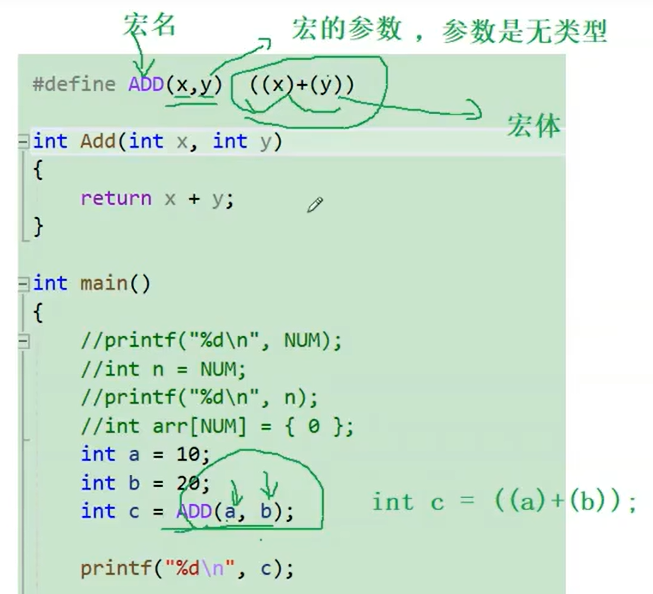


不能正常运行

原因是：全局变量具有外部链接属性，所以第一种情况可以不同源文件声明使用，但是static修饰过后，全局变量的外部链接属性就被改变为内部链接属性了，故不能跨文件使用。

static修饰函数：基本逻辑和全局变量一致。

33.define定义宏：本质上是完成替换

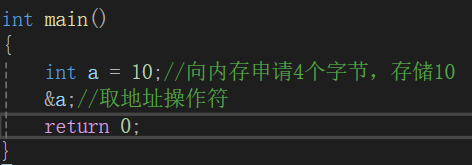


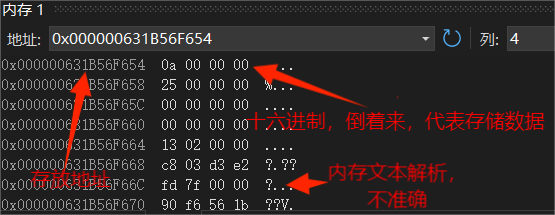
34.指针：

内存：为了有效的使用内存，就把内存划分成一个个小的内存单元，每个内存单元的大小是1个字节，每个内存单元都有它的编号。它的编号就是它的地址，它的地址也叫做指针。所以存放地址或者指针的变量就叫做指针变量。



a的地址是第一个字节的地址，即首地址





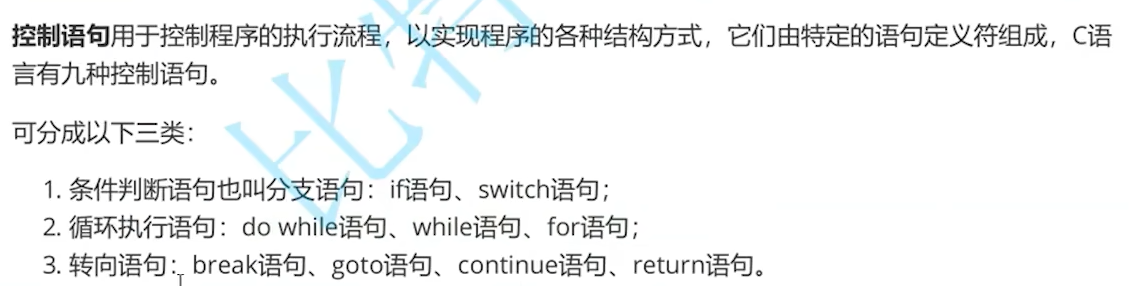
图中十六进制为1，2，3，4，5，6，7，8，9，a，b，c，d，e，f

指针变量p：int\* p =&a;中int的意思是指针变量p指向的对象是int类型的；\*说明了p是指针变量

35.结构体：对于复杂的对象，c语言给了自定义类型的能力，自定义类型中有一种叫做结构体struct，结构体就是把一些单一类型组合在一起的做法。

**二、c语言进阶**

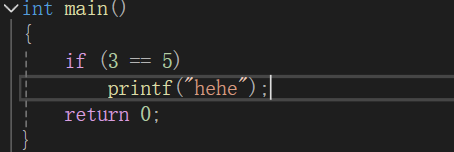
1.语句：表达式语句、函数调用语句、控制语句、复合语句和空语句。分支语句和循环语句都属于控制语句。



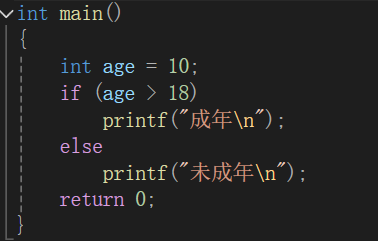
2.分支语句和循环语句：

分支语句：

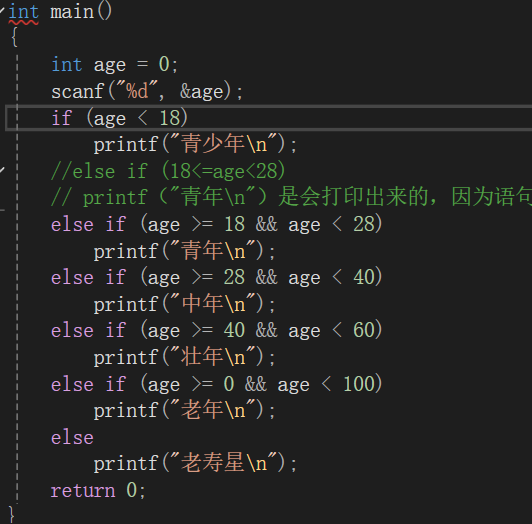
if语句



语法结构1

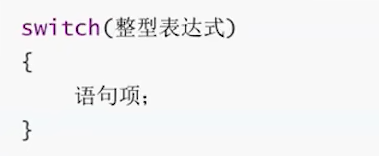


语法结构2

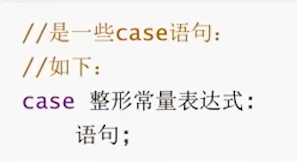


语法结构3

switch：常用于多分支情况



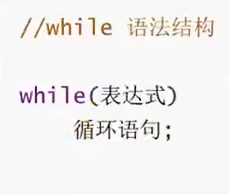
switch语句结构



语句项

循环语句：

while



for：for循环默认后边跟一个循环语句，多了要加代码块

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

do while：



goto语句：C语言中提供了可以随意滥用的 goto语句和标记跳转的标号。从理论上 goto语句是没有必要的，实践中没有goto语句也可以很容易的写出代码。但是某些场合下goto语句还是用得着的，最常见的用法就是终止程序在某些深度嵌套的结构的处理过程。

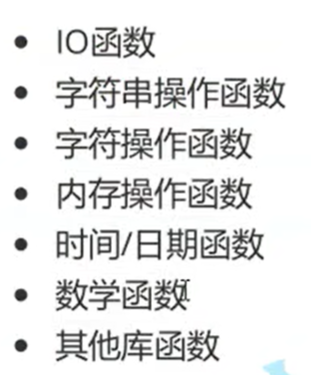
例如：一次跳出两层或多层循环。

多层循环这种情况使用break是达不到目的的。它只能从最内层循环退出到上一层的循环。

3.指针变量初始化：int\* p=NULL；

4.函数：有库函数和自定义函数

库函数：查阅msdn或者[cplusplus.com/reference/](https://cplusplus.com/reference/)

****

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

自定义函数：参数可以有0，1，多个



5.函数参数

实际参数：真实传给函数的参数，叫实参。实参可以是：常量、变量、表达式、函数等。无论实参是何种类型的量，在进行函数调用时，它们都必须有确定的值，以便把这些值传送给形参。

形式参数：形式参数是指函数名后括号中的变量，因为形式参数只有在函数被调用的过程中才实例化(分配内存单元)，所以叫形式参数。形式参数当函数调用完成之后就自动销毁了。因此形式参数只在函数中有效。形式参数和实际参数符号可以相同，可以不同。

6.函数的调用：

传值调用：函数的形参和实参分别占有不同内存块，对形参的修改不会影响实参。

传址调用：传址调用是把函数外部创建变量的内存地址传递给函数参数的一种调用函数的方式。这种传参方式可以让函数和函数外边的变量建立起真正的联系，也就是函数内部可以直接操作函数外部的变量。

要想把函数里的值可以在主函数里使用，可以采取传址，传数组，创建全局变量等方法。

7.函数的嵌套调用：可以嵌套调用，不能嵌套定义，函数默认返回int类型，写函数一定要写明返回类型，如果要有返回值，那就一定要写明返回内容，不要模棱两可。

文本

描述已自动生成

8.函数的声明的定义：告诉编译器有一个函数叫什么，参数是什么，返回类型是什么。但是具体是不是存在，函数声明决定不了。函数的声明一般出现在函数的使用之前。要满足先声明后使用。函数的声明一般要放在头文件中的。

9.函数的递归：程序调用自身的编程技巧称为递归。

递归做为一种算法在程序设计语言中广泛应用。一个过程或函数在其定义或说明中有直接或间接调用自身的一种方法，它通常把一个大型复杂的问题层层转化为一个与原问题相似的规模较小的问题来求解递归策略，只需少量的程序就可描述出解题过程所需要的多次重复计算，大大地减少了程序的代码量。递归的主要思考方式在于：把大事化小

两个必要条件：

-存在限制条件，当满足这个限制条件的时候，递归便不再继续。

-每次递归调用之后越来越接近这个限制条件。

10.函数的迭代：循环就是迭代的一种

11.数组：数组在内存中是连续存放的

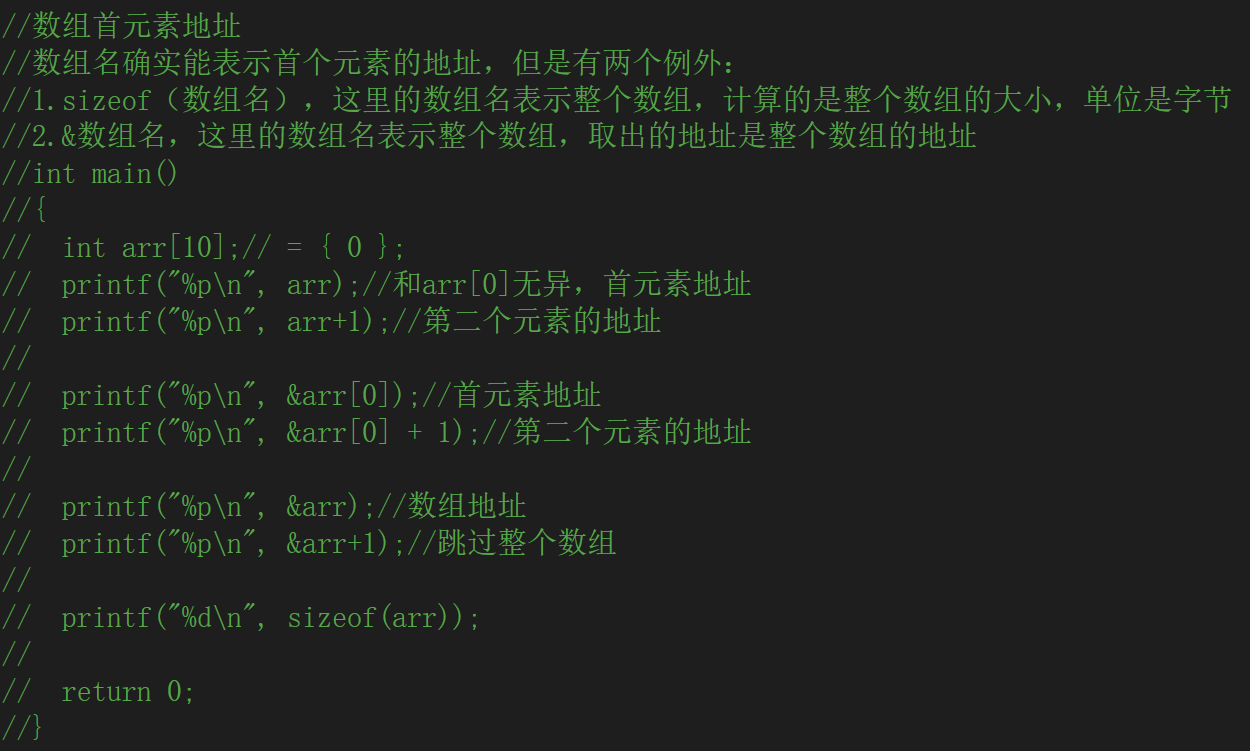


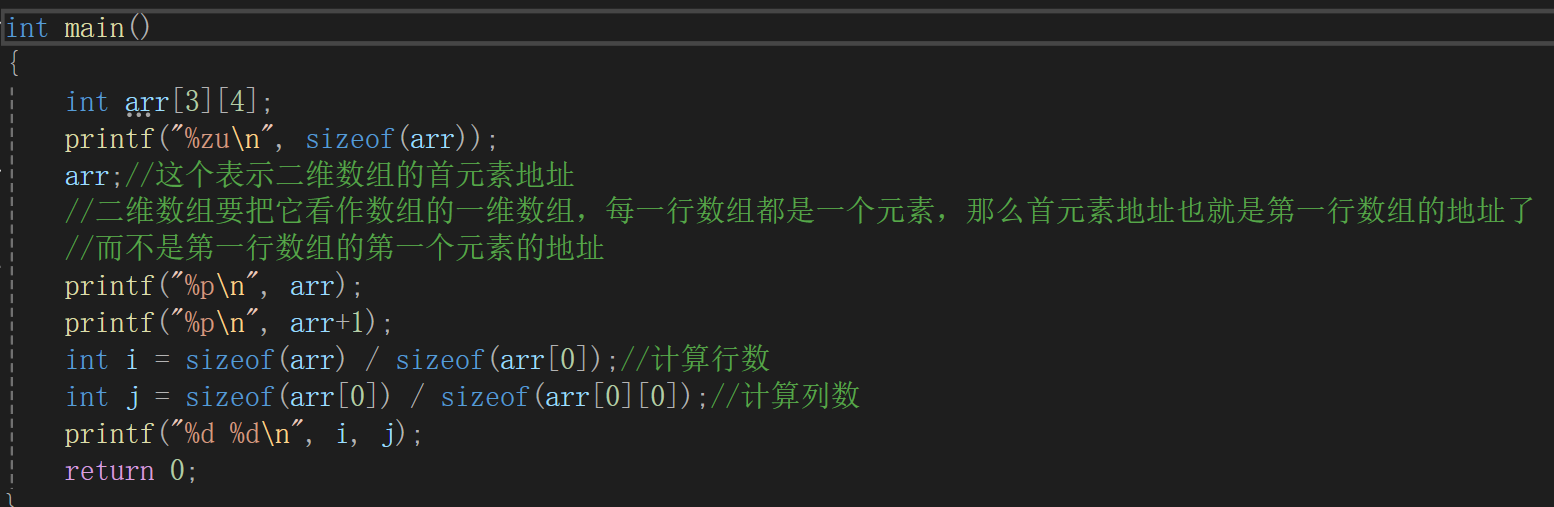
二维数组：



数组越界：往往编译器不报错，需要自己注意

数组名：





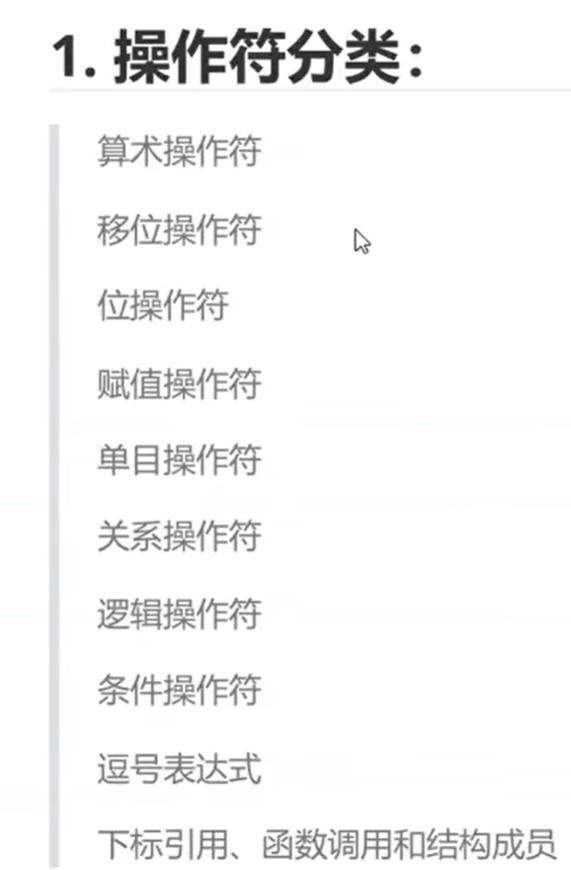
数组类型：

int arr [10]={0}中arr的类型为int [10]

12.sizeof和strlen的区别：

sizeof是计算通过某个类型所创造出来的变量占用的空间大小；strlen是一个库函数，专门用来求字符串长度的（\0之前的），仅限于字符串。

13.操作符：



算术操作符：+ - \* / %

/：整型和浮点型除法

%：符号两端必须是整数

移位操作符：左移<< 右移>>

移动的是二进制位的补码，其操作数只能是整数，不能为负数，浮点数

左移操作符：左边丢弃，右边补0；

右移操作符：

算术移位：右边丢弃，左边补原符号位

逻辑移位：右边丢弃，左边补0

位操作符：& | ^

其操作数只能是整数，不能为浮点数

按位与&：相同取同，异则取0

按位或|：有1取1，双0为0

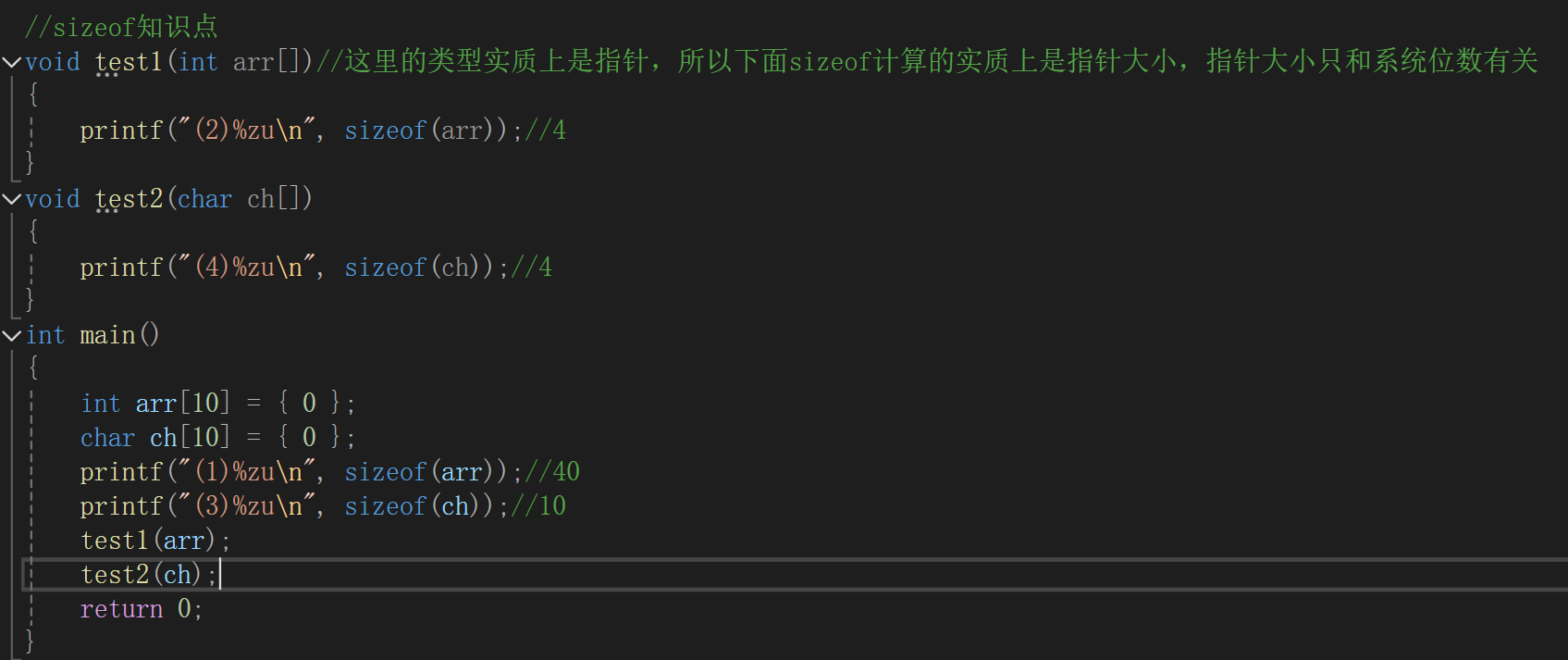
按位异或：相同为0，相异为1

赋值操作符：= += -= \*= /= %= >>= <<= &= |= ^=

int a =0;初始化

a = 20;赋值

单目操作符：



关系操作符

逻辑操作符：&& ||

int a = 3;

int b = 5;

int c = a && b;//c=1

//a && b只有两个结果0和1

条件操作符：三目操作符

逗号表达式

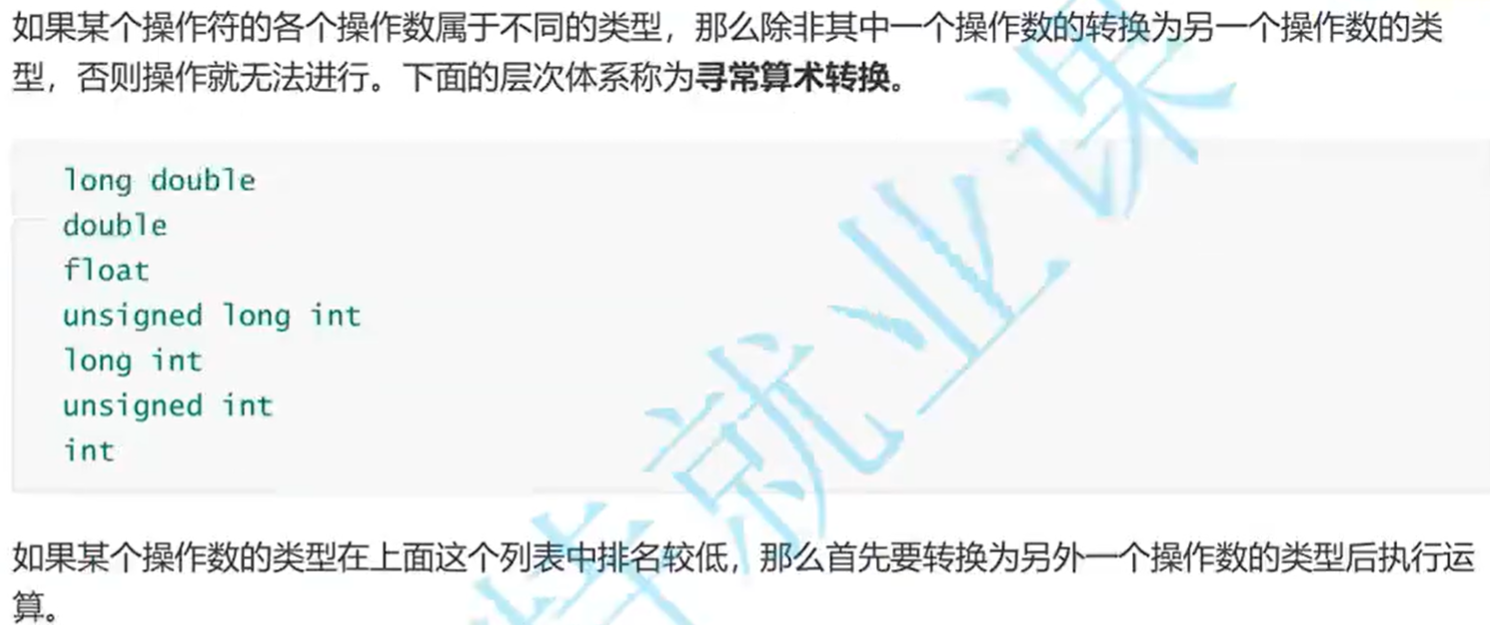
下标引用，函数调用和结构成员

14.表达式求值：由三个因素影响1.操作符优先级2.操作符结合性3.是否控制求值顺序

隐式类型转换：C的整型算术运算总是至少以缺省（默认）整型类型的精度来进行的。

为了获得这个精度，表达式中的字符和短整型操作数在使用之前被转换为普通整型，这种转换称为整型提升。整型提升是按照变量的数据类型的符号位来提升的。

算术转换：



操作符优先级：

&& || ?: ,可以控制求值顺序







15.指针：指针是内存中一个最小单位的编号，也称地址，指针变量是存放地址的变量

在32位的机器上，地址是32个0或者1组成二进制序列，那地址就得用4个字节的空间来存储，所以一个指针变量的大小就应该是4个字节。

那如果在64位机器上，如果有64个地址线，那一个指针变量的大小是8个字节，才能存放一个地址。

16.指针类型：

决定解引用一次性可以访问的字节数。int\*类型指针解引用可以访问4个字节，char\*类型指针解引用可以访问1个字节，其他类型以此类推。

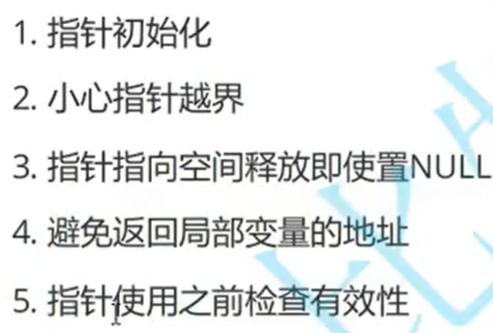
决定指针变量+-1跳过的字节数。因为char\*类型的一般情况下存储字符的地址，读取一个字符每次需要跳过一个字节；而int\*类型的存放的是占四个字节的整型的地址，读取一个整型每次需要跳过4个字节。指针的类型所具备的访问特点由它的适用对象所决定。

不同类型指针变量，即便解引用访问和跳过字节数一致，也不能通用。

指针变量的大小和指向对象的空间大小没有任何关系。

17.野指针：指针指向的位置是不可知的，随机的，不正确的，没有明确限制的。产生的原因有：指针未初始化，指针越界访问和指针指向的空间释放。

18.如何避免产生野指针?



19.指针和指针相减得到的绝对值是两个指针之间的元素的个数，只有指向同一块内存空间的两个指针可以相减，指针+指针无意义

20.指针的关系运算：就是大小比较

标准规定:

允许指向数组元素的指针与指向数组最后一个元素后面的那个内存位置的指针比较，但是不允许与指向第一个元素之前的那个内存位置的指针进行比较。

21.二级指针：用来存放一级指针变量的地址

22.指针数组：存放指针的数组

23.结构体：结构是一些值的集合，这些值称为成员变量。结构的每个成员可以是不同类型的变量，结构的成员可以是标量、数组、指针，甚至是其他结构体。

24.结构体声明：



25.结构体传参：首选传结构体的地址，因为如果直接传实参，且结构体本身很大，形参的创建需要占很多空间，导致性能降低。