# 数据

## 什么是数据

生活中时时刻刻都在跟数据打交道，比如体重数据、血压数据、股价数据等。在我们使用计算机的过程中，会接触到各种各样的数据，有文档数据、图片数据、视频数据，还有聊QQ时产生的文字数据、用迅雷下载的文件数据等。

## 数据的分类

计算机中存储的数据可以分为两种：静态数据和动态数据。

1. 静态数据

* 概念：静态数据是指一些永久性的数据，一般存储在硬盘中。硬盘的存储空间一般都比较大，现在普通计算机的硬盘都有500G左右，因此硬盘中可以存放一些比较大的文件。
* 存储的时长：计算机关闭之后再开启，这些数据依旧还在，只要你不主动删掉或者硬盘没坏，这些数据永远都在。
* 哪些是静态数据：静态数据一般是以文件的形式存储在硬盘上，比如文档、照片、视频等。

1. 动态数据（临时数据）

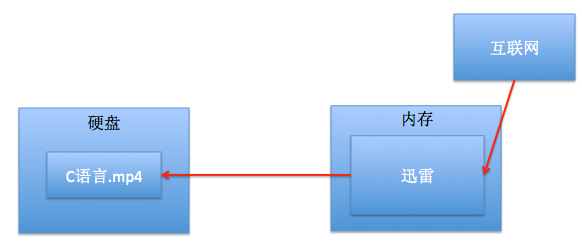
* 概念：动态数据指在程序运行过程中，动态产生的临时数据，一般存储在内存中。内存的存储空间一般都比较小，现在普通计算机的内存只有4G左右，因此要谨慎使用内存，不要占用太多的内存空间。
* 存储的时长：计算机关闭之后，这些临时数据就会被清除。
* 哪些是动态数据：当运行某个程序（软件）时，整个程序就会被加载到内存中，在程序运行过程中，会产生各种各样的临时数据，这些临时数据都是存储在内存中的。当程序停止运行或者计算机被强制关闭时，这个程序产生的所有临时数据都会被清除。

你可能会问：既然硬盘的存储空间这么大，为何不把所有的应用程序加载到硬盘中去执行呢？有个主要原因是内存的访问速度比硬盘快N倍。

程序员最关心的是什么数据呢？

1. 静态数据和动态数据的转换

静态 -> 动态 

动态 -> 静态 

## 数据的大小

1. 不管是静态还是动态数据，都是0和1组成的。0和1如何组成这么多的数据？
2. 数据都有大小，静态数据就会占用硬盘的空间，动态数据就占用内存的空间
3. 数据越大，包含的0和1就越多，比特位和字节
4. 1 KB = 1024 B，1 MB = 1024 KB，1 GB = 1024 MB，1 TB = 1024 GB

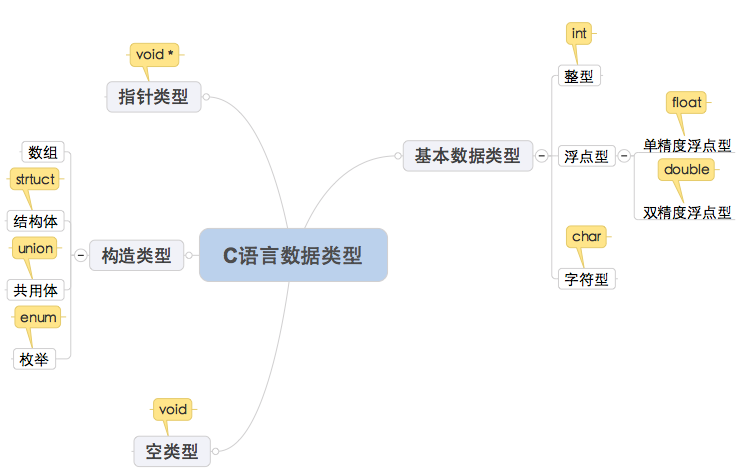
## app中的各种数据

 ****



## C语言中的数据类型

由于app中的数据种类繁多，为了方便操作数据，C语言数据进行了分类



# 常量

## 什么是常量

常量，表示一些固定的数据

## 常量的分类

1> 整型常量(int)

包括了所有的整数，比如6、27、109、256、-10、0、-289等

2> 浮点型常量(float\double)

浮点型常量分为double和float两种数据类型

* double：双精度浮点型，其实就是小数。比如5.43、-2.3、0.0等（注意，0.0也算是个小数）
* float：单精度浮点型，也是小数，比double的精确程度低，也就是说所能表示的小数位数比较少。为了跟double区分开来，float 型数据都是以f结尾的，比如5.43f、-2.3f、0.0f。需要注意的是，绝对不能有10f这样格式的，编译器会直接报错，只有小数才允许加上f。

3> 字符常量(char)

* 将一个数字（0~9）、英文字母（a~z、A~Z）或者 其他符号（+、-、!、?等）用单引号括起来，这样构成的就是字符常量。比如'6'、'a'、'F'、'+'、'$'等。

注意：单引号只能括住1个字符，而且不能是中文字符，下面的写法是错误的：'abc'、'123456'、'男'

4> 字符串常量

* 将一个或者多个字符用双引号（""）括起来，这样构成的就是字符串常量。比如"6"、"男"、"哇哈哈"、"abcd"、"my\_car4"，其实printf("Hello World");语句中的"Hello World"就是字符串常量。

那究竟6、'6'、"6"在用法上有什么区别呢？这个先不作讨论，以后会介绍。

## 习题

下面的都是什么类型的常量？

10.6 19.0f 0.0 0 ‘A’ “男” “mj” 294 ‘+’

# 变量

## 什么是变量

当一个数据的值需要经常改变或者不确定时，就应该用变量来表示。比如游戏积分。

## 定义变量

1. 目的

* 任何变量在使用之前，必须先进行定义。
* 定义变量的目的是：在内存中分配一块存储空间给变量，方便以后存储数据。
* 如果定义了多个变量，就会为这多个变量分别分配不同的存储空间。

1. 格式

变量类型 变量名;

比如int num;

* 变量名属于标识符
* 变量类型
* 不同类型的变量占用不同大小的存储空间。内存极其有限，分配适当的存储空间
* 约束变量所存放的数据类型（方便运算）

1. 实例

int main()

{

int i;

char c;

int a, b;

return 0;

}



## 变量的使用

1. 赋值

* 往变量里面存点东西，就是赋值。赋值语句后带个分号;

i = 10;

注意：这里的等号=，并不是数学中的“相等”，而是C语言中的赋值运算符，作用是将右边的常量10赋值给左边的变量i

* 第一次赋值，可以称为“初始化”
* 初始化的两种形式
* 先定义，后初始化：int a; a = 10;
* 定义的同时进行初始化：int a = 10;

1. 修改

* 可以修改变量的值，多次赋值。每次赋值都会覆盖原来的值

i = 10;

i = 20;

变量i最后的值是20

* 使用printf输出一个\多个变量的值

int a = 10, c = 11;

printf("a=%d, c=%d", a, c);

* double\float\char的输出，格式符的一些小技巧

double height = 1.55;

char blood = 'A';

printf("height=%.2f, 血型是%c", height, blood);

* 简单的加减操作

int a = 10 + 20;

* 没有初始化时不要拿来使用(下面的写法是不建议的)

int score;

printf("score=%d", score);

1. 变量之间值的传递

* 可以将一个变量的值赋值给另一个变量

int a = 10;

int b = a;

* 连续赋值

a = b = 10;

## 常见错误

1. 变量名相同 int a = 10; int a = 12;
2. 变量的作用域不对

* 变量的创建和释放过程
* 代码块作用域 { int a = 10; }

## 习题

1. 交换整型变量a、b的值。比如a=10、b=11；交换之后a的值是11，b的值是10。用两种方式实现：

* 使用第三方变量

int temp;

temp = a;

a = b;

b = temp;

* 不使用第三方变量

a = b - a;

b = b - a;

a = b + a;

1. 观察下图的游戏界面，思考需要定义多少个变量？

