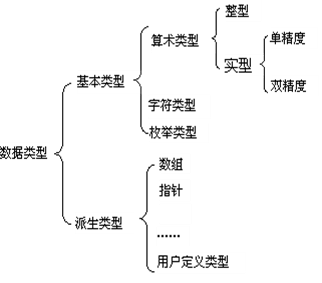
**C语言基本分析：**

一、基本成分：

1、数据成分：

C语言的数据构造方式可分为基本类型和派生类型，其中基本类型分为算数类型、字符类型、枚举类型三类，算数类型又包括整型和实型（单精度、双精度）；派生类型分为数组、指针、结构体、共用体和空类型等。



2、运算成分：

C语言具有强大的计算能力，一共有34种运算符，可分为8类：

算术运算符包含+、-、\*、/、%；赋值运算符分为简单赋值运算符和复合赋值运算符；自增运算符和自减运算符；sizeof运算符；逗号运算符，其中整个逗号表达式的值是最后一个表达式的值；关系运算符包含<、<=、>、>=、==、!=；逻辑运算符有&&（与）、||（或）、!（非）；C语言特有的三目运算符为“？：”；位运算符包括&、|、^（按位异或）、~、<<、>>等。

3、控制成分：

C语言允许顺序结构、条件选择结构和循环结构三种控制结构。其中条件选择结构使用if-else或switch-case语句实现；循环结构使用for循环或while循环或do-while循环完成。

4、传输成分：

C语言提供多种输入输出函数供程序员选择。

C语言提供scanf（从标准输入（键盘）读取并格式化）和printf（发送格式化输出到标准输出（屏幕））作为格式化输入输出函数；getchar和putchar为单个字符输入\输出函数；gets和puts是文本行输入\输出函数；以及对应的文件输入输出函数：fscanf、fgets、fgetc、fread；fprintf、fputs、fputc、fwrite。

二、语言特性：

心理特性中，C语言具有一致性，即采用的标记法总是协调一致的，同时不存在二义性，但是C语言存在一些易产生误解而需要程序员注意的表达，如if-if-else，有时程序员想当然的将第二个else对应于第一个if，然而C语言中else总是属于最近的没有else的if而导致错误；C语言具有32个关键字，9种控制语句并且支持结构化部件的使用，程序员记忆的与编码有关的信息总量较少，有较好的紧致性；C语言程序由函数模块组成，模块独立，局部性较高；C语言在整体上使用顺序结构，故具有较好的线性；C语言作为一种传统的高级语言，一般作为入门语言进行学习，具有传统性。

工程特性中，C语言支持结构化部件和各种复杂的数据结构，具备特殊的I\O处理和按位操作，但是不支持面向对象方法，故将设计翻译成代码较为便利；C语言的编译器发展已经较为成熟，能够生成效率较高的代码；C语言拥有ANSI C标准，已经被标准化，程序代码的可移植性较高；C语言配套的开发环境也十分丰富，有devC++等开发工具，可减少编码时间并提高代码质量；由于C语言是一种结构化开发语言，故各函数均可轻松复用，可复用性好；C语言已经拥有一些工具使得生成注释文档变得方便，提高语言的可维护性。

应用特性中，C语言作为一种通用语言，几乎可以应用于所有场景中，同时由于C语言具有指针等方便进行硬件操作的功能，C语言特别用在需要与硬件进行较多接触的情况中。

**C语言实例分析：**

1、程序名：slime-volleyball/final.c

2、来源：<https://github.com/availity9/slime-volleyball/blob/master/final.c>

3、代码行：668 lines（583 sloc）

4、程序语言：C语言

5、注释：

序言性注释：

例如：

typedef struct ball {

float x;

float y; //position

float vx; //velocity

float vy;

int r; //radius

} Ball\_type;

// this calls options menu and uses the return to decide what to do next

void actOnOptions(Person\_type \*pPlayer1, Person\_type \*pPlayer2, Ball\_type \*pBall, Person\_type \*head, bool \*quit) {

int opt = optionsMenu();

if (opt == 1)

resetGame(pPlayer1, pPlayer2, pBall);

else if (opt == 2) {

displayData(pPlayer1, pPlayer2, pBall, head, quit);

} else if (opt == 3) {

\*quit = true;

} else

newPlayers(pPlayer1, pPlayer2, pBall);

}

功能性注释：

例如：

int main() {

// initialize variables and structs

int c;

Person\_type \*head = NULL, \*p1, \*p2, \*p11, \*p22;

……

}

6、程序设计风格分析：

（1）源程序中的内部文档分析：

该程序中的标识符命名适当，选择了含义明确而又简短的名字，例如类型Person\_type和Ball\_type，以及变量player1、player2等，能够正确提示标识符所代表的实体，见名知意，而又不易混淆。

程序的注释书写得当，序言性注释很好的介绍了模块的功能及重要的局部变量等内容，功能性注释很好的说明了各部分的功能细节，同时在该程序中还有提示性注释，记录一些易于犯错的细节，如函数使用前须先声明等内容，方便查阅和浏览。

（2）程序视觉组织分析：

该程序书写不吝啬空格和空行等的使用，同时注重使用缩进表现嵌套层次结构，帮助人们从视觉上看清程序的结构；该程序也多次通过添加括号突出运算的优先级，例如使用“(wd / 2) – 100”算式进行计算，清楚明了。

（3）数据说明分析：

数据说明的次序较为规范，大部分以目的或数据类型作为次序分类标准，同时使用了大量的注释说明不那么见名知意的数据或自定义的复杂数据类型的结构和意义。

在同一类型的变量顺序安排上，该程序存在有待改进的地方，例如int vx = 0, vy = 0, lives = 3, colors1[3] = {0, 255, 0}, colors2[3] = {0, 0, 255}, wins = 0, radius = 50;这句，可以将这些变量按照字母顺序进行排列，以便于后期使用查找。

（4）语句构造分析：

该程序几乎没有将多条语句写在一行的情况，较为清晰；循环及条件嵌套最多为三层，同时辅以花括号和缩进标志结构，故逻辑较为清楚明白。

同时，该程序在这方面存在改进的空间，程序中存在将else写在前一个if的结束花括号之后的情况，在这种情形下，我们可以将else另起一行，并与相应if缩进一致，使结构更加清晰。

（5）输入输出分析：

该程序的输入输出方式为类似图形用户界面式的输入输出，故较为方便用户使用。