1.1计算机程序

1.2计算机语言，高级语言

1.3 c语言特点：1）语言简洁，紧凑，使用方便，灵活。

1. 运算符丰富
2. 数据类型丰富
3. 具有结构化的控制语言
4. 语法限制不严格，程序设计自由度大
5. 允许直接访问物理地址（可直接对硬件操作，具有高级语言和低级语言功能的双重性）
6. 可移植性好
7. 生成代码质量高，程序执行效率高

1.4简单的c语言程序

#include<stdio.h>

Int mian()

{

printf((“hello world”);

Return 0;

}

比较两者中大的

#include<stdio.h>

Int main()

{

Int a,b,c;

Scanf(“%d,%d”&a,&b);

C=max(a,b);

Printf(“max=%d”,c);

Int max(int x,int y)

{

Int z;

If(x>y)

Z=x;

Else

Z=y;

Return (z);

}

}

预处理命令

全局声明

函数定义

函数首部 函数类型 函数名 参数类型 参数名

函数体 声明部分 执行部分（可为空）

总是从main函数开始

数据声明加分号

本身不提供输入输出语句

应包含注释

1.5编译

运行

二 算法

2.1什么是算法

为解决一个问题而采用的方法和步骤

算法+数据结构=程序

2.2简单算法举例

1\*2\*3\*4\*5

是否为闰年

1-1/2+1/3-1/4+1/5...-1/100

判断是否为素数

2.3算法特性1）有穷性

1. 确定性
2. 有零个或多个输入
3. 有一个或多个输出
4. 有效性

2.4流程图表示

三种结构：顺序，选择，循环（当型，直到型）

三 顺序程序设计

3.1举例

计算温度

计算利润

3.2常量和变量

1. 常量

整型常量

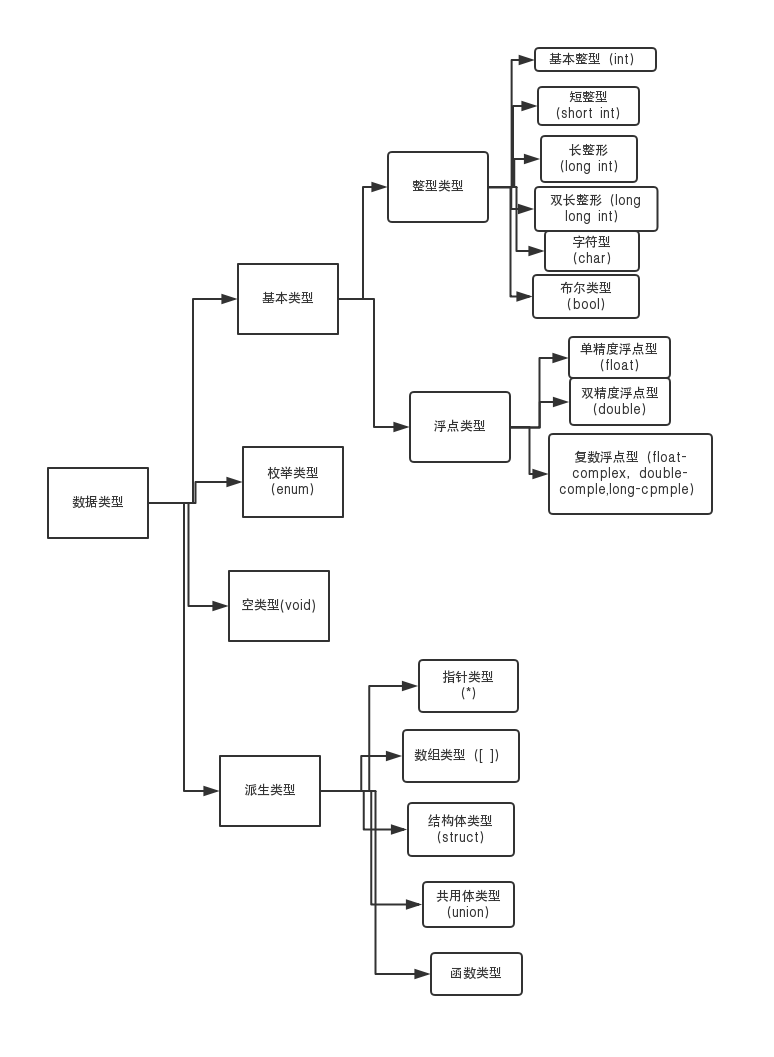
实型常量

字符常量（普通字符，转义字符）

字符串常量

符号常量

1. 变量
2. 常变量
3. 标识符（以字母数字下划线组成，第一个字符必须是字母或下划线）



正数的原码，补码，反码都一样。

负数补码为原码出符号位取反，补码在反码最后一位加一。

Int 4个字节

Short int 2个字节

Long int 4个字节

Long long int 8个字节

Char 一个字节

Float 4个字节

Double 8个字节

不同类型的运算转化为doouble后在运算

强制转化前面加括号，括号里加类型

运算符

1. 算术运算符 + - \* / % ++ --
2. 关系运算符 > < == >= <= ! =
3. 逻辑运算符 ！ && ||
4. 位运算符 << >> ~ | ^ &
5. 赋值运算符 =
6. 条件运算符 ？：
7. 逗号运算符 ，
8. 指针运算符 \* &
9. 求字节运算符 sizeof
10. 强制转换运算符 （（类型））
11. 成员运算符（. ->）
12. 下标运算符 （[]）
13. 其他 （如函数调用（））

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 优先级 | 运算符 | 名称或含义 | 使用形式 | 结合方向 | 说明 |
| 1 | [] | 数组下标 | 数组名[整型表达式] | 左到右 |  |
| () | 圆括号 | （表达式）/函数名(形参表) |  |
| . | 成员选择（对象） | 对象.成员名 |  |
| -> | 成员选择（指针） | 对象指针->成员名 |  |
| 2 | - | 负号运算符 | -算术类型表达式 | 右到左 | 单目运算符 |
| (type) | 强制类型转换 | (纯量数据类型)纯量表达式 |  |
| ++ | 自增运算符 | ++纯量类型可修改左值表达式 | 单目运算符 |
| -- | 自减运算符 | --纯量类型可修改左值表达式 | 单目运算符 |
| \* | 取值运算符 | \*指针类型表达式 | 单目运算符 |
| & | 取地址运算符 | &表达式 | 单目运算符 |
| ! | 逻辑非运算符 | !纯量类型表达式 | 单目运算符 |
| ~ | 按位取反运算符 | ~整型表达式 | 单目运算符 |
| sizeof | 长度运算符 | sizeof 表达式  sizeof(类型) |  |
| 3 | / | 除 | 表达式/表达式 | 左到右 | 双目运算符 |
| \* | 乘 | 表达式\*表达式 | 双目运算符 |
| % | 余数（取模） | 整型表达式%整型表达式 | 双目运算符 |
| 4 | + | 加 | 表达式+表达式 | 左到右 | 双目运算符 |
| - | 减 | 表达式-表达式 | 双目运算符 |
| 5 | << | 左移 | 整型表达式<<整型表达式 | 左到右 | 双目运算符 |
| >> | 右移 | 整型表达式>>整型表达式 | 双目运算符 |
| 6 | > | 大于 | 表达式>表达式 | 左到右 | 双目运算符 |
| >= | 大于等于 | 表达式>=表达式 | 双目运算符 |
| < | 小于 | 表达式<表达式 | 双目运算符 |
| <= | 小于等于 | 表达式<=表达式 | 双目运算符 |
| 7 | == | 等于 | 表达式==表达式 | 左到右 | 双目运算符 |
| != | 不等于 | 表达式!= 表达式 | 双目运算符 |
| 8 | & | 按位与 | 整型表达式&整型表达式 | 左到右 | 双目运算符 |
| 9 | ^ | 按位异或 | 整型表达式^整型表达式 | 左到右 | 双目运算符 |
| 10 | | | 按位或 | 整型表达式|整型表达式 | 左到右 | 双目运算符 |
| 11 | && | 逻辑与 | 表达式&&表达式 | 左到右 | 双目运算符 |
| 12 | || | 逻辑或 | 表达式||表达式 | 左到右 | 双目运算符 |
| 13 | ?: | 条件运算符 | 表达式1? 表达式2: 表达式3 | 右到左 | 三目运算符 |
| 14 | = | 赋值运算符 | 可修改左值表达式=表达式 | 右到左 |  |
| /= | 除后赋值 | 可修改左值表达式/=表达式 |  |
| \*= | 乘后赋值 | 可修改左值表达式\*=表达式 |  |
| %= | 取模后赋值 | 可修改左值表达式%=表达式 |  |
| += | 加后赋值 | 可修改左值表达式+=表达式 |  |
| -= | 减后赋值 | 可修改左值表达式-=表达式 |  |
| <<= | 左移后赋值 | 可修改左值表达式<<=表达式 |  |
| >>= | 右移后赋值 | 可修改左值表达式>>=表达式 |  |
| &= | 按位与后赋值 | 可修改左值表达式&=表达式 |  |
| ^= | 按位异或后赋值 | 可修改左值表达式^=表达式 |  |
| |= | 按位或后赋值 | 可修改左值表达式|=表达式 |  |
| 15 | , | 逗号运算符 | 表达式,表达式,… | 左到右 | 从左向右顺序结合 |

3.3 c语句

1.控制语句

1）if（） else 条件语句

2）for（） 循环语句

3）while 循环语句

4） do while（） 循环语句

5） continue 结束本次循环语句

6）break 中止执行switch或循环语句

7）switch 多分支选择语句

8） return 从函数返回语句

9）goto 转向语句

2 函数调用语句 函数调用加分号组成

3 表达式语句 表达式加分号组成

4 空语句

5 复合语句

3.4 数据的输入与输出

c语言本身不提供输入输出语句 是通过函数实现的

Printf（格式控制，输出列表）

格式字符

可改变域宽

整数输出 d

字符输出 c

字符串输出 s

基本型 f

指数输出 e

八进制输出 o

十六进制输出 x

Scanf（格式控制，地址表列）

单个字符 putchar getchar

四 选择结构程序设计

If else if 以及嵌套

逻辑运算符：与 或 非

条件运算符： ？ ：

Switch 多分支选择

五 循环结构程序设计

While

Do ... while

For(循环变量赋初值;循环条件;循环变量增值)

Break

Continue

六数组

1. 定义和引用一维数组

定义：类型符 数组名 [常量表达符]

引用：数组名[下标]

一维数组初始化：要么给下标，要么初始化数组元素

1. 定义和引用二维数组

二维数组定义 ：类型说明符 数组名[常量表达式][常量表达式4]

二维数组引用：数组名[下标][下标]

二维数组初始化:给定下标，初始化数组元素可以不定义一维长度，但要定义二维长度

1. 字符数组

定义 初始化 引用 储存最后\0

用字符串处理函数

1. Puts
2. Gets
3. Strcat strcat（字符数组一，字符数组二）字符串连接
4. Strcyp（字符串一，字符串二） 字符串复制，将二复制到一（只能用于输出，不能用新数组承接）
5. Strncpy（字符串一，字符串二，数字），将字符串一的前几个值替换成字符二的（只能用于输出，不能用新数组承接）
6. Strlen 测量字符串长度，实际长度。
7. Strlwr 转化为小写
8. Struper 转化为大写

七 用函数实现模块化程序设计

1. 定义函数

类型名 函数名（参数）

{

函数体

return 返回值；

}

如：int max（int a）

{

Int

Printf

Scanf

（能写主函数里面的其他函数都能写）

返回值由函数类型决定，void型的可以没有返回值。  
}

程序从main函数开始到main函数结束。

函数如果写在main函数后要先在main函数中声明才能使用。

1. 调用函数 函数括号中是形参，调用时输入实参

形参和实参的区别（形参被调用前不占内存，函数调用时为形参分配内存；调用结束，内存释放）

从库调入函数 加include<xxx.h>

函数不能嵌套定义但可以嵌套调用（可以在调用一个函数是调用另一个如：max（jia（））//是先调用jia函数在调用max函数）

函数递归调用

数组不能作为形参

形参与实参类型一致，个数相同顺序相同

函数递归，在本函数中调用自己叫递归。

数组元素作为实参，向形参传递的是值，用数组名作为实参时，传递的是元素首地址。

值传递是单向的，不改变实参的值。而地值传递改变原来的值。

多维数组作为函数参数

1. 局部变量和全局变量

局部变量（只能在本函数中使用）：

1. 在函数开头定义
2. 在函数内的复合语句内定义

全局变量：在函数外部定义

1. 变量储存方式

程序区，静态储存区，动态储存区

自动变量auto

静态局部变量 static 不可改变的值

八 指针

8.1什么是指针：一个变量的储存地址即是该变量的指针

8.2 指针变量

使用 将变量的地址赋值给指针变量，int \*p; p=&a;

定义 类型名 \*指针名，定义时可以初始化

引用 int \*p=&a; printf(“%d”,\*p);输出a的值，printf（“%d”，p）；输出的是a的地址；

作为参数：

8.3 指针引用数组

P=&a[0]或int \*p=a

\*p++ 输出\*p后p自增一

\*(p++) 输出\*p后p自增一

(\*p)++ 输出\*p后地址所对应的值加一

++(\*p) 地址所对应的值加一后输出\*p

++\*p 地址所对应的值加一后输出\*p

\*++p p自增后输出值

\*(++p) p自增后输出值

数组名作为函数的形参，形参改变实参随之改变。

形参和实参可以为数组名或指针变量

指针调用多维函数

8.4指针引用字符串

8.5指向函数的指针

定义：类型名 （\*指针变量名）（函数参数列表）

8.6返回指针的函数

类型名 \*函数名（参数列表）

8.7指针数组

存放指针的数组

定义：类型名 \*数组名[数组长度]

8.8动态内存分配与指向他的指针

Void \*malloc（unsigned int size）

在动态储存区分配一个长度为size的空间

Void \* calloc （unsigned n，unsigned size）

分配n个长度为size的连续空间 足够保存一个数组

Void free（void \*p）

释放p所指的动态空间

Void \*realloc（void \*p，unsigned int size）

改变通过malloc或calloc分配空间的大小

8.9小结

指针是地址

指针变量的值是地址

指针运算

9用户建立数据类型

9.1定义和使用结构体变量

定义结构体

Struct 结构体名

{

成员列表

} ；

定义变量

Struct 结构体名

{

成员列表

}变量名列表；

或

结构体类型名 结构体变量名；

应用成员的值：

结构体变量名.成员名

9.2结构体数组

{成员列表}数组名 [数组长度]；

结构体类型名 数组名[数组长度]；

9.3结构体指针

指向结构体数组的指针

9.4指针处理链表

什么是链表

动态进行储存分配的结构

构造静态链表 结点是定义好的用完也不能释放的

动态链表的创建

9.5共用体

将不同类型的变量储存到同一地址

Union 共用体名

{

成员列表；

}变量列表；

引用共同体 a.i;

9.6枚举类型

如果变量只有几种可能的值，可以定义为枚举类型。

Enum [枚举名]{枚举元素列表}；

9.7 用typedef

定义新的类型名代替原有变量名

1.简单的用一个新的类型名代替原有类型名

如：typedef int integer 指定integer类型与int相同

2.命名一个简单类型名代替复杂类型名

Typedef sruct

{

Int month;

}Date;

声明了一个Date的结构体

十 文件的输入输出

1.程序文件：源程序文件后缀为.c,目标文件 后缀为.obj，可执行文件 后缀为.exe

2.数据文件：供程序读写的文件

输入输出流

文件名:文件有唯一标识由文件路径+文件名主干+文件后缀组成

分为ascll文件（文本文件）和二进制文件

打开和关闭文件

打开fopen（文件名，使用文件方式）r只读 w只写 a 追加

关闭fclose(文件指针)

FILE \*fp

Fp=fopen(“al”,”r”);

Fclose（fp）；

10.3 顺序读写文件

Fgetc（fp）从fp指向的文件读入一个字符

Fputc（ch，fp）把字符ch写入文件指针变量为fp的指向文件中

向文件中读写一个字符串

Fgets（str，n，fp）从fp指向的文件读入一个长度为（n-1）的字符串，存放在数组str中

Fputs（str，fp）把str所指向的字符串写道文件指针变量fp中