

C语言课程学习报告

制作报告人

20049200530

王赫雨

目录

一：笔记部分 3

二：易错点 13

三：积累部分 14

四：学习感悟 15

1. 笔记部分

（一）

1.计算机的工作原理：储存程序+程序控制。

2.字节byte：是最小的储存单位。8位二进制数字为1个字节。

即1B=8bit,1KB=103B,1MB=106B,1GB=109B,1TB=1012B.

3.计算机中的数值：

原码：正数符号为0，负数符号为1，其它位为绝对值。

反码: 正数的反码与原码相同；负数的反码是符号位不变，其他位按位取反。

补码: 正数的补码与其原码相同；负数的补码为其反码最末位加 1。(负数在计算机中以补码形式存在。)

//在unsigned型的变量中，最高位不代表符号，而代表数值。

1. 符号char类型的变量在计算机中以ASCII编码型储存。且数字=编码值之差（如9=‘9’-‘0’）。小写字母与对应的大写字母之差为32。
2. C语言是结构化的语言。有结构化的控制语句（顺序，判断，循环）。（二）
3. C语言程序基本结构

库中已经接触的

#include<stdio.h>

#include<math.h> 数学函数

#include<string.h> 处理字符串

#define

#include<stdio.h> //编译预处理

int main() //主函数

{ //开始

printf(“”); //输入

return 0; //函数执行完毕返回函数值

} //结束

1. 数据类型 对应格式转换符

int 整型变量 %d

float 单精度浮点数 %f

double 双精度浮点数 %lf

char 字符类型 %c

long int 长整型变量 %ld

unsigned 无符号型

%x为十六进制输出

\n为换行输出

\t为向前8个空格位置

保留位数方法：如%.1f为保留一位（不适用于scanf中）

1. 基本语句

printf() 输出

scanf（） 输入

(三)

1. 算术运算符

+，-，\*，/，%（求余）,++,--

1. ++,--,+=,-=,\*=,/=

a+=b 与a=a+b 等价

1. =b 与a=a-b 等价

a\*=b 与a=a\*b 等价

a/=b 与a=a/b 等价

a++ 与a+=1 与a=a+1 等价

1. - 与a-=1 与a=a-1 等价
2. 关系运算符

< <= > >= == !=

1. 优先级

先计算算术运算符，再计算关系运算符，最后计算赋值运算符

且 算术运算符与关系运算符均为从左向右计算，赋值运算符为从右向左计算。

1. 选择结构
2. if条件基本结构

if（） 括号内表达式非0时即执行

{

}

1. switch多分支选择结构

switch（） 括号内可以为计算式，但必须为整数，

{ case 0：…………

case 1：…………

Default：………… （全未执行时，执行此句）

}

1. 两个判断语句的比较

If中可以是关系比较运算，而switch中必须是整数

If中是判断条件逐句判断，为真时执行{}内语句，而switch是直接从某个整数开始，依次向后执行！

1. 循环结构
2. 三种循环

while 循环 do while 循环 for 循环

1. 各种循环基本结构

while（循环条件）

do{}while（判断条件）；注意此处有分号！！！

for（变量初始化；循环条件表达式；循环变量的调整）

1. continue 与break 对循环的调控作用

continue执行时直接跳出本次循环。

break执行时终止循环。

1. 三种循环的比较

for循环可以与while循环相互转换。

do while与while的循环之前要对变量初始化

在已知循环次数是用for循环更方便，在必须先进行一次循环再判断时用do while循环更方便，其它情况下可以用while。

1. 数组
2. 用途：用于存放数据
3. 定义数组：

元素类型 数组名 [常量表达式 (表示元素个数—数组的长度)]

float s[50]; // 50名学生一门课的成绩

int a[10]; // 10个元素的整型数组

double b[10]; // 常量10个元素的double数组

char c[50]; // 50个元素的char型数组

1. 数组从[0]开始到[n-1]结束，所以注意不要越界。
2. 数组的赋值初始化举例

①a[5]={1，3，5，8，61}

②给数组中某一个元素赋值a[2]=5

③用循环给数组赋值

for(i=0;i<=4,i++){scanf(“%d”,&a[i]);}

1. 冒泡排序！

冒泡排序的核心程序

int a[10]={9,8,7,6,5,4,3,2,1,0},i,j,t;

for(j=1;j<=N-1;j++) //进行N-1次循环，实现N-1趟比较

{

for(i=0;i<=N-1-j;i++) //在每一趟中进行N-j次比较相邻元素两两比较

{ if(a[i]>a[i+1]) //相邻两个数比较, 注意检查数组不要越界

{ t=a[i]; a[i]=a[i+1]; a[i+1]=t; } // 交换 } }

for(i=0;i<N;i++)

printf(”%d ”,a[i]);

每次循环，使最大的数沉底。

1. 冒泡排序的优化

在第一层的循环内设置一个标志变量flag=0，初始化

在第二层循环中，若执行了数字交换，则改变flag的值

当flag量在循环中没有改变时，执行break，直接终止循环，可以减少计算时间，提高效率。

7.二维数组的定义：

元素类型 数组名[行数][列数]

int a[3][3]

则为

|  |  |  |  |
| --- | --- | --- | --- |
| a[3][3] | 0 | 1 | 2 |
| 0 | a[0][0] | a[0][1] | a[0][2] |
| 1 | a[1][0] | a[1][1] | a[1][2] |
| 2 | a[2][0] | a[2][1] | a[2][2] |

1. 字符串 字符数组
2. 字符串的初始化

Char s[81]={‘a’，‘b’‘c’ }； //三个有效字符

或

char s[]=”abcd” //五个有效字符

因为在第二种定义中，会在末尾自动追加‘\0’,表示字符串的结束。

定义时应定义大的数组，因为字符串比较大。

1. 字符串的输入输出

输入：

scanf(“%s”,s)，注意没有&

或gets（s） （可以接受在空格的字符串）

输出：

Printf（“%s”，s）；

或

For（i=0；s[i]!=’\0’,i++）

Printf(“%c”,s[i])

或

puts(s)

逐字输出，遇到\0结束输出，换行。

1. 字符串处理函数#include<string.h>

strcat(s1,s2)

连接两个字符串，并储存到s1中

Strcpy（s1,s2）

把s2复制给s1

strcmp（s1,s2）

比较连个字符串的大小

strlen（s）

计算字符串的长度，不算\0

1. 函数与递归
2. 定义函数

函数名

形式参数

返回函数值：

函数计算的值返回调用者

int fun(int a,int b)

{

int ret ;

ret=a+b;

return ret;

int r=fun(a,b)

对定义函数的调用

2.函数的类型

有返回值函数与无返回值函数

int /float /double /char fun（）都是有返回值的函数

void fun（）是无返回类型的函数 不返回任何数值

3.递归函数

在定义函数中，自己调用自己的函数为递归函数

void output（int a）

{if（a!=0)

Printf(“%d”,a/10);

else return;

}

可以用来逆序输出整数。

栈：先进后出的数据结构。

在递归函数中，先入栈，后出栈，先进后出。

出

栈

1

2

3

4

5

output（0）

output（1）

output（12）

output（123）

output（1234）

output（12345）

入

栈

二、易错点积累

（一）格式等基本错误

1.注意在输入时，不能输入中文的字符，中文的符号和分号。

2.变量数据类型与对应的格式转换符不相符合：如int 未与%d对应，double 未与%lf对应。

3.在scanf printf中，引号的位置引错。

4.在scanf中，变量前面没有加&取地址符号，或在字符串前加了&符号。

（二）运算时的错误

1.整数除以整数时自动舍去小数点，导致结果错误。

2.使两个字符串相等时，不能用s1=s2，应该使用函数strcpy（s1,s2）。

（三）在选择，循环内的错误

1.对于

if（a=b）式子，是先把b的值赋给a，若b不为0，则执行下面语句。

错把if（a==b）写成了if（a=b）

2.对于if（） switch（），while（）等，括号后面不要加分号。

3.在for（；；）循环中！

for内定义的变量只能在for循环内使用，不能在{}以外再使用

如果需要使用，应该在{}外面定义变量。

1. 数组

1.注意在使用数组时，不要数组越界

2.数组的最后一位，是数组的元素数减一

三、积累部分

1.对于三个循环while do for

在已知循环次数时，使用for循环更好

在必须执行一次循环时，使用do更好

在其它情况下，使用for 或 while 都比较好

2.提取后几位字节时，使用a%16，因为16进制的两个数为一个字节

3.调试的内容可以用{}花括号括起来，因为对于C语言编译器，可以不管{}，顺序执行其中的内容，而且方便调试完成后删除其中的内容

4.递归函数的积累：

void output（int a）

{

If(a!=0){

Output(a/10);printf(“%d”,a%10)

}

Else return；

}

1. C语言学习感想

学习了C语言，让我对计算机的了解更加深刻，而且同时掌握了一门编程语言，更锻炼了逻辑思维能力。

通过学习C语言这半个学期，在解决一个问题时，让我学会了先理清楚思路，明确每一步应该怎么做，下一步应该做什么，极大的提高了解决问题的效率。并为以后的课程学习打下了基础。

在剩下的半个学期里，我一定会更加努力的学习C语言，提高我的成绩，同时提高解决问题的能力！