

25300180035 池裕涵 程序设计第二周周末作业

吴老师&助教骆老师好，以下是我的作业。

-教材

P.39 题 2, P.40 题 10、11、12：均为理论题，请简明扼要一一对应作答。通过直接输入或手写拍照后插入 word/markdown 电子文档的形式作答，要求提交 pdf 文档。

P40 题 16：编程题，需要提供.c 源代码和成功运行的截图，截图插入理论题电子文档。

编程题：

输入两个整数，输出大于/小于/等于/不等于/整除关系等。

程序文件名 quantitative_relationship.c

Github 网址

[Homework-in-CS10004-Programming-by-yhchi/quantitative_relationship.c at main · Chi-](#)

The screenshot shows the Visual Studio Code interface. The left sidebar displays a file tree with several files, including 'quantitative_relationship.c' which is currently selected. The main area shows the C code for the program. The terminal at the bottom shows the output of running the program with input '3 9'.

```
#include<stdio.h>
int n,m;int main()
{
    n=m=-1; scanf("%d %d",&n,&m);
/*
匹配失败: %d 转换说明符要求输入的是十进制数字。
当 scanf 在输入流中遇到非数字字符（如字母 'a'）时，它会立即停止读取，并认为这次转换失败了。
故变量 a 未被赋值 -> 变量 a 将保持它原来的值。
*/
    if(n<0||m<0){printf("请重新输入2个自然数\n");return 0;}
    if(n<m)printf("%d 小于 %d\n",n,m);
    if(n>m)printf("%d 大于 %d\n",n,m);
    if(n==m)
    {
        printf("%d 等于 %d\n",n,m);
        if(n!=0)printf("%d 与 %d 互相整除\n",m,m);
        return 0;
    }
    if(n!=m)printf("%d 不等于 %d\n",n,m);
    if(m!=0&&n!=0)printf("%d 被 %d 整除\n",n,m);
    if(n!=0&&m!=0)printf("%d 被 %d 整除\n",m,n);
    //会先进行零判断—若为0，则跳过&&后的语句，不会报错
    return 0;
}
```

```
64\bin\gdb.exe' '--interpreter=mi'
3 9
3 小于 9
3 不等于 9
9 被 3 整除
PS D:\FDU_1\CS10004_Programming>
```

[Shan0707/Homework-in-CS10004-Programming-by-yhchi](#)

(由于修改了账户名，所以 0917 提交的 pdf 文档内的网址现在点不开了)

截图如上

以下为理论题部分

[Homework-in-CS10004-Programming-by-yhchi/Weekend 2 theory.c at main · Chi-Shan0707/Homework-in-CS10004-Programming-by-yhchi](#)

第 2 题

- (1) 哪也不是
- (2) 哪也不是
- (3) 实型常量
- (4) 整型常量,
- (5) 实型常量
- (6) 哪也不是
- (7) 哪也不是
- (8) 整型常量
- (9) 哪也不是
- (10) 整型常量,
- (11) 哪也不是
- (12) 整型常量
- (13) 实型常量
- (14) 实型常量

The screenshot shows a code editor window titled 'Weekend 2 theory.c 9+' with the file path 'code_pack > C Weekend 2 theory.c > ...'. The code is a C program with comments explaining various types of constants. The code is as follows:

```
1  /*
2  第二周周末作业
3  -教材
4  P.39 题 2, P.40 题 10、11、12：均为理论题，请简明扼要一一对应
5  P40 题16：编程题，需要提供.c源代码和成功运行的截图，截图插入在
6
7 */
8 // 理论题
9
10 // 2. 判断类型
11 E-4; // (1) 哪也不是
12 A423; // (2) 哪也不是
13 -1E-31; // (3) 实型常量
14 // ±Ne±A, A 必须为整数, N 不可省, e=E
15 0xABCL; // (4) 整型常量, 0x十六进制, l Long 类型
16 .32E31; // (5) 实型常量
17 087; // (6) 哪也不是, 0 开头意味八进制, 何能有“8”?
18 0xL; // (7) 哪也不是 0x十六进制后可以补 L 表示 Long 类型, 但中间得有数
19 003; // (8) 整型常量
20 0x12.5; // (9) 哪也不是
21 077; // (10) 整型常量,
22 11E; // (11) 哪也不是, E 后得有数字
23 056L; // (12) 整型常量
24 0.; // (13) 实型常量
25 .0; // (14) 实型常量
26
27
```

```
30 /*  
31 quotation—Accelerated C++ writes:  
32 x++:对左值x加1, 生成x的初始值  
33 --x:递减左值x, 生成递减后的x  
34 左值: 特定的有地址的变量  
35 右值: 右值通常是“临时”的、没有名字的、即将被销毁的值, 我们不能对它取地址。  
36 */  
37 //10.省去了对i,j,k,x,y的变量类型声明和初始值的赋予  
38 k=(j+(i++));// (1) 把整型变量i,j的和赋值给变量k, 并同时让i的值增加1  
39 x<=1;// (2) 把整型变量x值扩大一倍  
40 j-=(-i);// (3) 把变量i减去1后, 将变量j减去变量i  
41 r=i%j;// (4) 计算变量i除以变量j的余数r  
42 y=(int)(x*1000+0.5)/1000.0;// (5) 让实型变量x精确到小数点后第三位四舍五入后的值赋给实型变量y  
43  
44  
45 //11.描述表达式  
46 float V=4.0/3.0*3.14*r*r*r; //(1)  
47 float R=1.0/(1.0/R1+1.0/R2); //(2)  
48 float y=x*x*x*x*x*x*x*x*x+6.0; //(3)  
49 float F=6.674 *M1*M2/(R*R)*(1e11); //(4)  
50 float y=sin(x)/x+fabs(cos(3.1415926/2.0*x));//(5)  
51 return 0<a&&a<10; //(6)  
52 return (x==1)^{y==2};//(7)
```

第 10 题

```
55 //12. 预测a,b,c值
56 //在a=4的情况下执行以下语句
57 b=a*a++;
58 //此时, b=20,a=5
59 //在a=5的情况下执行以下语句
60 c=++a+a;
61 //此时, c=12,a=6
62 /*
63 reason:
64 https://learn.microsoft.com/zh-cn/cpp/cpp/cpp-built-in-operators-precedence-and-associativity?view=msvc-170
65 参见↑
66 运算符优先级指定了包含多个运算符的表达式中的运算顺序。
67 运算符关联性指定了在包含多个具有相同优先级的运算符的表达式中，操作数是与其左侧还是右侧的操作数组合。
68
69 后缀递增与后缀递减属于第2组优先级，从左到右关联
70 前缀递增与前缀递减属于第3组优先级，从右到左关联
71
72 * / % 属于第5组优先级；+ - 属于第6组优先级，皆为从左到右关联
73
74
75 a*a++ 先执行 (a++) 语句返回4，但将a左值递增至5；再执行*，得到 5*4=20
76 c=++a+a 先执行 (++a) 语句，a左值递增至6，并返回6，再执行+，得到6+6=12
77 */
```

第 12 题

reason:

<https://learn.microsoft.com/zh-cn/cpp/cpp/cpp-built-in-operators-precedence-and-associativity?view=msvc-170>

参见↑

运算符优先级指定了包含多个运算符的表达式中的运算顺序。

运算符关联性指定了在包含多个具有相同优先级的运算符的表达式中，操作数是与其左侧还是右侧的操作数组合。

后缀递增与后缀递减属于第 2 组优先级，从左到右关联

前缀递增与前缀递减属于第 3 组优先级，从右到左关联

* / % 属于第 5 组优先级；+ - 属于第 6 组优先级，皆为从左到右关联

a*a++ 先执行 (a++) 语句返回 4，但将 a 左值递增至 5；再执行*，得到 $5*4=20$
c=++a+a 先执行 (++a) 语句，a 左值递增至 6，并返回 6，再执行+，得到 $6+6=12$

下图为程序验证。

```
code_pack > C verify_week2_theory.c U ✘
C verify_week2_theory.c > main()
1 #include<stdio.h>
2 int a,b,c;
3
4 int main()
5 {
6     printf("Verify the result of Problem 12 on Page 40.\n");
7     a=4;
8     b=a*a++;
9     printf("a=%d,b=%d\n",a,b);
10    c=++a+a;
11    printf("a=%d,c=%d\n",a,c);
12    return 0;
13 }
```

问题 68 输出 调试控制台 终端 端口

```
+ < cppdbg: verify_week2_theory.exe < ... | [ ] x
h' '--stderr=Microsoft-MIEngine-Error-ct0elfbi.i50' '--pid=Microsoft-MIEngine-Pid-1b2mmqjm.vzk' '--dbgExe=D:\C_C++\mingw64\bin\gdb.exe' '--interpreter=mi'
Verify the result of Problem 12 on Page 40.
a=5,b=20
a=6,c=12
PS D:\FDU_1\CS10004_Programming>
```