第1章

习题解答

1.1 习题1解答

1. 填空题

(1) 面向对象的方法将现实世界中的客观事物描述成具有属性和行为的**对象**,抽象出共同属性和行为,形成**类**。

(2) C++程序开发通常要经过5个阶段,包括**编辑**、**编译**、**连接**、**运行**、**调试**。首先是**编辑**阶段,任务是**编辑源程序**,C++源程序文件通常带有**.cpp**扩展名。接着使用**编译器**对源程序进行**编译**,将源程序翻译为机器语言代码(目标代码),过程分为词法分析、语法分析、代码生成3个步骤。

在此之前,**预编译器**会自动执行源程序中的**预处理指令**,将其他源程序文件包括到要编译的文件中,以及执行各种文字替换等。

连接器的功能就是将目标码同缺失函数的代码连接起来,将这个"漏洞"补上,生成**可执行文件**。程序运行时,可执行文件由操作系统装入内存,然后CPU从内存中取出程序执行。若程序运行过程中出现了错误,还需要对程序进行**调试**。

(3) 对象与对象之间通过**消息**进行相互通信。

(4) **类**是对具有相同属性和行为的一组对象的抽象;任何一个对象都是某个类的一个实例。

(5) **多态性**是指在一般类中定义的属性或行为被特殊类继承之后可以具有不同的数据类型或表现出不同的行为。

(6) 面向对象的软件开发过程主要包括**面向对象的方法分析**、**面向对象的设计**、**面向对象的编程**、**面向对象的测试**和**面向对象的维护**。

(7) C++提供**名字空间(namespace)**将相同的名字放在不同空间中来防止命名冲突。

(8) # include < iostream >是一条**预处理**指令(语句),在**编译(或预处理)**时由**编译器(或预编译器)**执行,其功能是**将iostream文件包含(复制)到指令处**。

(9) C++中使用**cin**作为标准输入流对象,通常代表键盘,与提取操作符>>连用;使用**cout**作为标准输出流对象,通常代表显示设备,与<<连用。

2. 简答题

(1) 简述机器语言、汇编语言、高级语言的特点。

【答】 机器语言是计算机直接执行的语言,由二进制的0和1构成的一系列指令组成;汇编语言是机器语言的助记符;高级语言是接近人的自然语言习惯的编程语言,通过编译变成机器语言。

（2）结构化语言与面向对象的语言是截然分开的吗？

【答】 不是截然分开的，面向对象的程序设计中也包含过程，含有结构化的思想。

（3）C++语言是纯粹的面向对象的程序设计语言吗？

【答】 不是。C++语言是在C语言的基础上引入面向对象程序设计思想形成的，它保留了某些面向过程的程序设计特征。

（4）C语言编写的程序不加修改就可以在C++编译器中编译吗？

【答】 绝大多数C语言程序都可以在C++编译器中编译，因为C++语言兼容C语言。

（5）C++的源程序是什么类型的文件？如何在Word中进行编辑？

【答】 C++源程序是文书文件（文本文件）；在Word中编辑，存为扩展名为.cpp的文本文件。在Windows的写字板、记事本中都可以编辑。

（6）如何将一个C++源程序变成可执行程序？产生的各类文件的扩展名是什么？

【答】 通过编译变成带扩展名.obj的目标文件；再通过连接变成带扩展名.exe的可执行文件。

（7）如果要求不使用include包含头文件，有什么办法使程序正常编译运行？

【答】 在相应的文件夹（子目录）中找到需包含的头文件，将头文件复制到包含处。

（8）下列程序中如有错误与不妥当之处请指出。

```
/// ***************************************************
 *     程序文件名：p1_2.cpp                            *
 *************************************************** /
Using namespace std
#include <iostream>;
using std::endl;
int main()
    float num1, num2, num3;                    //定义3个数
    cin << num1 << num2 << num3;
    cout >> "The average is:" >> setw(30) >>(num1 + num2 + num3)/3 >> endl;
    return 0;
}
```

【答】 错误处标号如下：

```
①/// ***************************************************
 *     程序文件名：p1_2.cpp                            *
 *************************************************** /
②Using namespace std③
#include <iostream>;④
using std::endl;⑤
int main() ⑥
    float num1, num2, num3;                    //定义3个数
    cin ⑦ << num1 << num2 << num3;
    cout >> ⑧ "The average is:" ⑨ >> setw(30) ⑩>>(num1 + num2 + num3)/3 >> endl;
    return 0;
}
```

① 行注释符号//将块注释的头/ * 注释掉了，使得块注释的尾 * /没有相应的头与之匹配；
② 关键字 Using 中包含了大写字母，应改为 using；
③ using namespace std 不是预处理指令，要以分号结尾；
④ #include <iostream>是预处理指令，不能以分号结尾，而且要作为程序的开头行；
⑤ 使用了 using namespace std，就不必单独使用 std::endl；
⑥ int main()后少了{；
⑦ cin 应与提取操作符>>连用；
⑧ cout 应与插入操作符<<连用；
⑨ “The average is:”为全角引号，应改为英文；
⑩ 使用 setw(30)应包含头文件 iomanip。
修改后的正确程序如下：

```
/*************************************************
*     程序文件名: p1_2.cpp                        *
*************************************************/
#include <iostream>
#include <iomanip>
using namespace std;
int main(){
    float num1,num2, num3;                          //定义 3 个数
    cin >> num1 >> num2 >> num3;
    cout <<"The average is:"<< setw(30) <<(num1 + num2 + num3)/3 << endl;
    return 0;
}
```

3. 选择题

(1) C++语言属于(　　)。

A. 机器语言　　B. 低级语言　　C. 中级语言　　D. 高级语言

【答】 D

(2) C++语言程序能够在不同操作系统下编译、运行，说明 C++具有良好的(　　)。

A. 适应性　　B. 移植性　　C. 兼容性　　D. 操作性

【答】 B

【注解】 对于计算机硬件，一般使用“兼容”一词，对于程序使用“移植”。

(3) #include 语句(　　)。

A. 总是在程序运行时最先执行　　B. 按照在程序中的位置顺序执行

C. 在最后执行　　D. 在程序运行前就执行了

【答】 D

【注解】 #include 是预处理指令，在编译时就执行了，没有对应的机器指令。

(4) C++程序运行时，总是起始于(　　)。

A. 程序中的第一条语句　　B. 预处理命令后的第一条语句

C. main()　　D. 预处理指令

【答】 C

(5) 下列说法正确的是(　　)。

A. 用C++语言书写程序时,不区分大小写字母

B. 用C++语言书写程序时,每行必须有行号

C. 用C++语言书写程序时,一行只能写一个语句

D. 用C++语言书写程序时,一个语句可分几行写

【答】 D

(6) 在下面概念中,不属于面向对象编程方法的是(　　)。

A. 对象　　B. 继承　　C. 类　　D. 过程调用

【答】 D

(7) 下列程序的运行结果为(　　)。

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
    cout << setprecision(4)
         << setw(3)
         << hex
         << 100/3.0
         <<" , ";
    cout << 24 << endl;
    return 0;
}
```

A. 3.333e+001, 18　　B. 33.33, 18　　C. 21,18　　D. 33.3,24

【答】 B

4. 程序填空题

为了使下列程序能顺利运行,请在空白处填上相应的内容:

```
#include ____①____
#include ____②____
____③____;
____④____;
____⑤____;
int main()
{
    float i, j;
    cin ____⑥____ i ____⑥____ j;
    cout ____⑦____ setw(10) ____⑦____ i*j;
    return 0;
}
```

【答】 程序如下：

```
#include <iostream>
#include <iomanip>
using std::cin;
using std::cout;
using std::setw;
int main()
{
    float i, j;
    cin >> i  >> j;
    cout << setw(10)  <<  i * j;
    return 0;
}
```

5. 编程题

(1) 编写一个程序，输出用 * 组成的菱形图案。

【答】 程序如下：

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
    cout << setw(16)<<" * "<< endl;
    cout << setw(17)<<" *** "<< endl;
    cout << setw(18)<<" ***** "<< endl;
    cout << setw(19)<<" ******* "<< endl;
    cout << setw(20)<<" ********* "<< endl;
    cout << setw(19)<<" ******* "<< endl;
    cout << setw(18)<<" ***** "<< endl;
    cout << setw(17)<<" *** "<< endl;
    cout << setw(16)<<" * "<< endl;
}
```

或

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
    cout <<"      *       "<< endl;
    cout <<"    ***       "<< endl;
    cout <<"   *****      "<< endl;
    cout <<" *******  "<< endl;
    cout <<" ********* "<< endl;
    cout <<" *******  "<< endl;
```

```
    cout <<"   *****   "<< endl;
    cout <<"    ***    "<< endl;
    cout <<"     *     "<< endl;
    return 0;
}
```

（2）编写一个程序，输入任意十进制数，将其以八进制、十六进制的形式输出。

【答】 程序如下：

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
    int i;
    cout <<"输入十进制数:";
    cin >> i;
    cout <<"八进制:"<< oct << i <<"十六进制:"<< hex << i << endl;
    return 0;
}
```

（3）仿照本章例题设计一个程序，输入两个数，将它们相除，观察为无限循环小数时按精度从小到大输出的结果。

【答】 程序如下：

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
    double i,j;
    cout <<"输入两个数:";
    cin >> i >> j;
    cout << setprecision(1);
    cout <<"precision(1):"<< i <<"/"<< j <<" = "<< i/j << endl;
    cout << setprecision(2);
    cout <<"precision(2):"<< i <<"/"<< j <<" = "<< i/j << endl;
    cout << setprecision(3);
    cout <<"precision(3):"<< i <<"/"<< j <<" = "<< i/j << endl;
    cout << setprecision(8);
    cout <<"precision(8):"<< i <<"/"<< j <<" = "<< i/j << endl;
    cout << setprecision(9);
    cout <<"precision(9):"<< i <<"/"<< j <<" = "<< i/j << endl;
    cout << setprecision(10);
    cout <<"precision(10):"<< i <<"/"<< j <<" = "<< i/j << endl;
    return 0;
}
```

运行结果：

```
输入两个数:2000  3↙
precision(1):2e+003/3=7e+002
precision(2):2e+003/3=6.7e+002
precision(3):2e+003/3=667
precision(8):2000/3=666.66667
precision(9):2000/3=666.666667
precision(10):2000/3=666.6666667
```

1.2　习题 2 解答

1. 填空题

(1) C++的基本数据类型可分为五大类，即**逻辑型(或布尔型)**、**字符型**、**整型**、**实型**、**空值型**，分别用关键字 **bool**、**char**、**int**、**float/double**、**void** 定义，长度分别为 **1**、**1**、**4**、**4/8**、**不定**字节。整型、字符型的默认符号修饰为 **signed**。

【注解】 另一种填法为：

C++的基本数据类型可分为五大类，即**逻辑型(或布尔型)**、**字符型**、**整型**、**浮点型**、**双精度型**，分别用关键字 **bool**、**char**、**int**、**float**、**double** 定义，长度分别为 **1**、**1**、**4**、**4**、**8** 字节。整型、字符型的默认符号修饰为 **signed**。

(2) 十进制数值、八进制数值、十六进制数值的前缀分别为 **1～9**、**0**、**0x(或 0X)**。

(3) 在 C++预定义的常用转义序列中，在输出流中用于换行、空格的转义序列分别为**\n**、**\t**。

(4) 布尔型数值只有两个，即 **true** 、**false**。在 C++的算术运算式中，它们分别当作 **1**、**0**。

(5) 字符由**单引号''**括起来，字符串由**双引号" "**括起来。字符只能有 **1** 个字符，字符串可以有**多**个字符。空串的表示方法为**"\0"(或" ")**。

(6) 关系运算符操作数的类型可以是**任何基本数据类型**，对其中的**实数**类型数不能进行直接比较。

(7) && 与||表达式**按从左到右**的顺序进行计算，以 && 连接的表达式，如果左边的计算结果为 **false(或 0)**，右边的计算不需要进行，就能得到整个逻辑表达式的结果：**false**；以||连接的表达式，如果左边的计算结果为 **true(或非 0)**，就能得到整个逻辑表达式的结果：**true**。

(8) >>运算符将一个数右移 n 位，相当于将该数**除以** 2^n，<<运算符将一个数左移 n 位，相当于将该数**乘以** 2^n。

(9) 所有含赋值运算的运算符的左边要求是**左值**。

(10) 前置++、--的优先级**高**于后置++、--。

(11) 按操作数数目分，运算符的优先级从高到低排列为**单目**、**双目**、**三目**，按运算符的性质分，优先级从高到低排列为**算术**、**移位**、**关系**、**按位**、**逻辑**。

(12) 在表达式中，会产生副作用的运算符有**++**、**--**、**各类赋值**。

(13) 函数执行过程中通过 **return** 语句将函数值返回，当一个函数不需要返回值时需要使用 **void** 作为类型名。

(14) 在 C++程序中，如果函数定义在后，调用在先，需要进行**函数原型声明**，告诉编译器函数的**(返回)类型**、**函数名**、**形式参数**。其格式和定义函数时的函数头的形式基本相同，它必

须以**分号**；结尾。

（15）函数参数传递过程的实质是将实参值通过**堆栈**一一传送给形参。

（16）递归程序分两个阶段执行，即**调用**、**回代**。

（17）带 inline 关键字定义的函数为**内联函数**，在**编译**时将函数体展开到所有调用处。内联函数的好处是节省**执行时间**开销。

（18）函数名相同，但对应形参表不同的一组函数称为**重载函数**，参数表不同是指**参数个数**、**类型不同**。

（19）确定对重载函数中函数进行绑定的优先次序为**精确匹配**、**对实参的类型向高类型转换后的匹配**、**实参类型向低类型及相容类型转换后的匹配**。

（20）内联函数的展开、重载函数的绑定、类模板的实例化与绑定均在**编译**阶段进行。

2. 选择题

（1）下列选项中，均为常用合理的数值的选项是（　　）。

A. ·25　1L　0Xfffe　　B. '好!'　3333333333　－01U

C. 10^8　'\'　'\x'　　D. 08　FALSE　1e＋08

【答】 A

（2）上述选项中，均为不合理（能通过编译，但不提倡使用）的数值的选项是（　　）。

【答】 B

【注解】 '好!'中用单引号引起来的不止一个字符；3333333333 超过整型数的范围；－01U 从符号看是一个负数，从后缀看是一个无符号数，相互矛盾。

（3）上述选项中，均为不合法（不能通过编译）的数值的选项是（　　）。

【答】 C

【注解】 10^8 不能在编辑器中输入；'\'、'\x'为不合法的转义序列。08 中的 8 不是八进制的合法数字；FALSE 不是 bool 型常量，false 为 bool 型常量。1e＋08 是合法的常量。

（4）下列选项中，均为合法的标识符的选项是（　　）。

A. program　a&b　2me　　B. ccnu@mail　C++　a_b

C. π　变量 a　a b　　D. ____ Line　_123　Cout

【答】 D

【注解】 A 中的 a&b，B 中的 ccnu@mail、C++含有不能构成标识符的符号 &、@、＋；2me 以数字开头，它们均为不合法的标识符。

（5）上述选项中，均为不合法的标识符的选项是（　　）。

【答】 C

【注解】 π 是一个数学符号；变量 a 以汉字开头，以汉字开头与含有汉字的都不能作为标识符；a 和 b 中间有空格，标识符中不能含空格。

（6）若定义“short int i＝32769;”，“cout << i;”的输出结果为（　　）。

A. 32 769　　B. 32 767　　C. －32 767　　D. 不确定的数

【答】 C

【注解】 32 769 为整型常量，十六进制形式为 0x00008001，赋给短整型后，截取低 16 位 0x8001，对应的十进制值为－32 767。

（7）若定义“char c＝'\78';”，则变量 c（　　）。

A. 包含 1 个字符　　B. 包含两个字符

C. 包含 3 个字符　　D. 定义不合法

【答】 D

【注解】 '\78'中的 8 不是八进制的合法数字，'\78'不是一个合法的八进制转义序列。

(8) 若定义“int a=7; float x=2.5, y=4.7;”，则 x+a%3 * static_cast<int>(x+y)%2/4 的值为(　　)。

A. 2.5　　B. 2.75　　C. 3.5　　D. 0.0

【答】 A

【注解】 依次计算“a%3=1; static_cast<int>(x+y)=7; 7%2/4=0; x+0=2.5。”

(9) 设 i 为 int 型、f 为 float 型，则 10 + i+'f'的数据类型为(　　)。

A. int　　B. float　　C. double　　D. char

【答】 A

【注解】 'f'为 char 类型，10、i 为 int 型。

(10) 设变量 f 为 float 型，将 f 小数点后第 3 位四舍五入，保留小数点后两位的表达式为(　　)。

A. (f * 100+0.5)/100　　B. (f * 100+0.5)/100.0

C. (int)(f * 100+0.5)/100.0　　D. (int)(f * 100+0.5)/100

【答】 C

【注解】 D 的结果为整数，A、B 的结果保留了小数点后若干位。

(11) 下列运算要求操作数必须为整型的是(　　)。

A. /　　B. ++　　C. !=　　D. %

【答】 D

(12) 若变量已正确定义并具有初值，下列表达式合法的是(　　)。

A. a:=b++　　B. a=b+3=c++　　C. a=b++=c　　D. a=b++, b=c

【答】 D

【注解】 A 中的 :=不是 C++中的运算符; B、C 中的 b+3、b++不是左值。

(13) 6 种基本数据类型的长度排列正确的是(　　)。

A. bool = char < int ≤ long = float < double

B. char < bool = int ≤ long = float < double

C. bool < char < int < long < float < double

D. bool < char < int < long = float < double

【答】 A

【注解】 bool、char、int、long、float、double 的长度分别为 1、1、4(2)、4、4、8 字节。

(14) 若变量 a 是 int 类型，执行“a='A'+1.6;”，正确的叙述为(　　)。

A. a 的值是字符 C　　B. a 的值是浮点型

C. 不允许字符型数与浮点型数相加　　D. a 的值是'A'的 ASCII 码值加上 1

【答】 D

【注解】 a 的值由 double 型转换成 int 型。

(15) 判断 char 型变量 c 是否为英文字母的表达式为(　　)。

A. 'a'<=c<='z' && 'A'<=c<='Z'

B. 'a'<=c&&c<='z' || 'A'<=c&&c<='Z'

C. 'a'<=c<='z' || 'A'<=c<='Z'

D. ('a'<=c||c<='z') && ('A'<=c||c<='Z')

【答】 B

【注解】 A、C中'a'<=c<='z'的运算顺序为('a'<=c)<='z'。

(16) 下列表达式中没有副作用的是(　　)。

A. cout<<i++<<i++　　B. a=(b=1)+=2

C. a=(b=1)+2　　D. c=a*b+ ++b

【答】 C

【注解】 A中的i++<<i++因计算顺序不同有不同的结果；B中的a=(b=1)+=2等价于a=(b=1)=(b=1)+2；D中的c=a*b+ ++b等价于c=a*b+(b=b+1)。

(17) 下列语句中的x和y都是int型变量，其中错误的语句是(　　)。

A. x=y++;　　B. x=++y;

C. (x+y)++;　　D. ++x=y;

【答】 C

(18) 对于if语句中的表达式的类型，下面描述正确的是(　　)。

A. 必须是关系表达式

B. 必须是关系表达式或逻辑表达式

C. 必须是关系表达式或算术表达式

D. 可以是任意表达式

【答】 D

【注解】 可以是任意表达式甚至是常量。

(19) 以下错误的if语句是(　　)。

A. if(x>y) x++;

B. if(x == y) x++;

C. if(x<y) {x++; y--;}

D. if(x!= y) cout<<x　else cout<<y;

【答】 D

【注解】 cout<<x不是一个完整的语句，应加上分号。

(20) 以下程序的输出结果为(　　)。

```
int main() {
    int a(20),b(30),c(40);
    if(a>b)a=b, b=c, c=a;
    cout<<"a="<<a<<",b="<<b<<",c="<<c;
    return 0;
}
```

A. a=20,b=30,c=40　　B. a=20,b=40,c=20

C. a=30,b=40,c=20　　D. a=30,b=40,c=30

【答】 A

(21) 以下程序的输出结果为(　　)。

```
int main() {
```

```
    int a(1),b(3),c(5),d(4),x(0);
    if(a < b)
        if(c < d) x = 1;
        else if(a < c)
            if (b < d) x = 2;
            else x = 3;
            else x = 6;
            else x = 7;
                cout << x;
    return 0;
}
```

A. 1　　B. 2　　C. 3　　D. 6

【答】 B

(22) 以下程序的输出结果为(　　)。

```
int main() {
    int x(1),a(0),b(0);
    switch(x) {
    case 0: b++;
    case 1: a++;
    case 2: a++; b++;
    }
    cout <<"a = "<< a <<",b = "<< b;
    return 0;
}
```

A. a=2，b=1　　B. a=1，b=1　　C. a=1，b=0　　D. a=2,b=2

【答】 A

(23) 以下程序的输出结果为(　　)。

```
int main()
{
    int a(15),b(21),m(0);
    switch (a % 3)
    {
        case 0: m++;break;
        case 1: m++;
            switch(b % 2)
            {
                default: m++;
                case 0: m++; break;
            }
    }
    cout << m;
    return 0;
}
```

A. 1　　B. 2　　C. 3　　D. 4

【答】 A

(24) 与“y=(x>0 ? 1 : x<0 ? -1 : 0);”的功能相同的 if 语句是(　　)。

A. if(x>0) y=1;
　else if (x<0) y=-1;
　else y=0;

B. if(x)
　if(x>0) y=1;
　else if (x<0) y=-1;

C. y =-1;
　if(x)
　if(x>0) y=1;
　else if (x==0) y=0;
　else y=-1;

D. y = 0;
　if(x>=0)
　if (x>0) y=1;
　else y=-1;

【答】 A

(25) 以下循环的执行次数为(　　)。

```
for(int i = 2; i!= 0;) cout << i -- ;
```

A. 死循环　　B. 0 次　　C. 1 次　　D. 两次

【答】 D

(26) 下列程序的运行结果是(　　)。

```
int main() {
    int a(1),b(10);
    do {
        b -= a; a++; } while (b-- <0);
    cout <<"a = "<< a <<",b = "<< b;
    return 0;
}
```

A. a=3，b=11　　B. a=2，b=8　　C. a=1,b= -1　　D. a=4，b=9

【答】 B

(27) 下面程序段循环执行了(　　)。

```
int k = 10;
    while (k = 3) k = k - 1;
```

A. 死循环　　B. 0 次　　C. 3 次　　D. 7 次

【答】 A

(28) 语句 while(!E)中的表达式!E 等价于(　　)。

A. E == 0　　B. E!=1　　C. E!=0　　D. E == 1

【答】 A

(29) 以下程序段(　　)。

```
x = -1;
do {x = x * x;} while (!x);
```

A. 是死循环　　B. 循环执行 1 次

C. 循环执行两次　　D. 语法错

【答】 B

(30) 与“for(表达式 1;表达式 2;表达式 3)循环体;”功能相同的语句为(　　)。

A. 表达式 1;
while(表达式 2){
循环体;
表达式 3;　}

B. 表达式 1;
while(表达式 2){
表达式 3;
循环体;　}

C. 表达式 1;
do {
循环体;
表达式 3; } while(表达式 2);

D. do {
表达式 1;
循环体;
表达式 3; } while(表达式 2)

【答】 A

(31) 以下循环体的执行次数为(　　)。

```
for(int x = 0,y = 0;(y = 123) && (x < 4); x++);
```

A. 死循环　　B. 次数不定　　C. 4 次　　D. 3 次

【答】 C

(32) 以下不是死循环的语句为(　　)。

A. for(y=0, x=1;x>++y ; x=i++) i=x;

B. for(; ; x++=i);

C. while(1) {x++;}

D. for(i=0; ; i--) sum+=i;

【答】 A

(33) 下列程序的运行结果是(　　)。

```
int main() {
    int a(1),b(1);
    for (; a <= 100; a++) {
        if (b >= 10) break;
        if (b % 3 == 1) {b += 3; continue; }
    }
    cout << a;
    return 0;
}
```

A. 101　　B. 6　　C. 5　　D. 4

【答】 D

(34) 以下循环体的执行次数为(　　)。

```
int i(0);
while (i < 10) {
    if(i < 1) continue;
    if(i == 5) break;
    i++;
}
```

A. 死循环　　B. 1 次　　C. 10 次　　D. 6 次

【答】 A

（35）执行语句序列

```
int n;
cin >> n;
switch(n)
{
    case 1:
    case 2:cout <<'1';
    case 3:
    case 4:cout <<'2';break;
    default:cout <<'3';
}
```

时，若键盘输入1，则屏幕显示（　　）。

A. 1　　B. 2　　C. 3　　D. 12

【答】 D

（36）在下列定义中，正确的函数定义形式为（　　）。

A. void fun(void)　　B. double fun(int x; int y)

C. int fun();　　D. double fun(int x,y)

【答】 A

【注解】 B中形参用分号分隔，C中定义以分号结尾，D中y没有类型。

（37）函数int fun(int x, int y)的声明形式不正确的为（　　）。

A. int fun(int , int);　　B. int fun(int y, int x);

C. int fun(int x, int y)　　D. int fun(int i, int j);

【答】 C

【注解】 函数声明要以“;”结尾。

（38）在C++语言中，函数返回值的类型由（　　）。

A. return语句中的表达式类型决定　　B. 调用该函数时的主调函数类型决定

C. 调用该函数时系统临时决定　　D. 定义该函数时所指定的数据类型决定

【答】 D

（39）若有函数调用语句：

```
fun(a+b, (x, y), (x, y, z));
```

此调用语句中的实参个数为（　　）。

A. 3　　B. 4　　C. 5　　D. 6

【答】 A

【注解】 (x, y)为一个参数。

（40）在C++中，关于默认形参值，正确的描述是（　　）。

A. 设置默认形参值时，形参名不能省略

B. 只能在函数定义时设置默认形参值

C. 应该先从右边的形参开始向左边依次设置

D. 应该全部设置

【答】 C

(41) 对重载函数的要求,正确的为(　　)。

A. 要求参数的个数不同　　B. 要求参数中至少有一个类型不同

C. 要求参数个数相同时类型不同　　D. 要求函数的返回类型不同

【答】 C

(42) 系统在调用重载函数时根据一些条件确定调用哪个重载函数,在下列条件中,不能作为依据的是(　　)。

A. 实参个数　　B. 实参类型　　C. 函数名称　　D. 函数类型

【答】 D

(43) 在下列函数原型声明中,错误的是(　　)。

A. void Fun(int x=0,int y=0);

B. void Fun(int x,int y);

C. void Fun(int x,int y=0);

D. void Fun(int x=0,int y);

【答】 D

(44) 若同时定义了以下函数,则 fun(8, 3.1)调用的是下列(　　)函数。

A. fun (float, int)　　B. fun (double, int)

C. fun (char, float)　　D. fun (double, double)

【答】 D

【注解】 函数重载不允许函数名和形参表都相同,仅返回类型不同;绑定的优先次序是精确匹配→对实参的类型向高类型转换后的匹配→实参类型向低类型及相容类型转换后的匹配。本题没有精确匹配,优先进行向高类型转换后的匹配。

3. 简答题

(1) 若定义"int x=3,y;",那么下列语句执行后 x 和 y 的值分别是多少?

① y=x++ -1;

② y=++x-1;

③ y=x-- +1;

【答】 ① "y=x++-1;"等价于"y=(x=x+1) -1;",执行完后 x、y 的值分别为 4、2。

② "y=++x-1;"等价于"y=(x=x+1) -1;",执行完后 x、y 的值分别为 4、3。

③ "y=x-- +1;"等价于"y=(x=x-1)+1;",执行完后 x、y 的值分别为 2、4。

(2) 运行下面的程序段,观察其输出,如将式中的 && 改为||,运行结果是什么?

```
int a = 1,b = 2,m = 2,n = 123;
cout <<((m = a > b)&&++n)<< endl;
```

【答】 a > b 的值为 0,m=0 的值为 0,短路求值,++n 不计算,因此 n 的值为 123,输出结果为 0。若改成

```
cout <<((m = a > b)||++n)<< endl;
```

a > b 的值为 0, m=0 的值为 0,++n 的值为 124,0||124 的值为 1,n 的值为 124,输出结果为 1。

(3) 运行下面的程序,观察其输出。

```
int x,a;
```

```
x = (a = 3 * 5,a * 4),a + 5;
cout <<"x = "<< x <<" a = "<< a << endl;
```

【答】 逗号运算符的优先级比＝低，先计算 a＝3＊5，再计算 a＊4，a＊4 作为(a＝3＊5，a＊4)的返回结果。

输出结果为 x＝60，a＝15。

(4) 下列表达式在计算时是如何进行类型转换的？

```
10/static_cast < float >(3) * 3.14 + 'a' + 10L * (5 > 10)
```

【答】 类型转换图如图 1-1 所示。

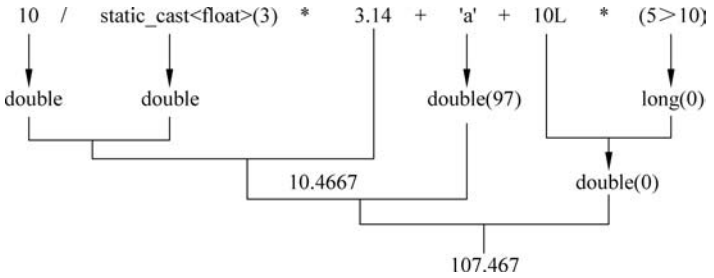

图 1-1 类型转换

4. 程序填空题

(1) 下列程序接受从键盘输入的两个数以及＋、－、＊、/运算符，将两个数进行加、减、乘、除，输出运算结果，请填空。

```
int main() {
    char c;
    float a,b,result(0);
    int tag(1);                          //标志,1:合法,0:数据或操作不合法
    cin >> a >> b >> c;
    switch (c) {
        case ①  : result = a + b; break;
        case ②  : result = a - b; break;
        case ③  : result = a * b; break;
        case ④  :
            if(b == 0) {
                cout <<"divide 0"<< endl;
                tag = 0;
                ⑤
            }
            result = a/b;
            break;
        ⑥
        tag = 0;
        cout <<"invalid operation"<< endl;
        ⑦
    }
    if(tag)
        cout << result << endl;
    return 0;
}
```

【答】 正确的程序如下：

```
#include <iostream>
using namespace std;
int main() {
    char c;
    float a,b,result(0);
    int tag(1);                          //标志,1:合法,0:数据或操作不合法
    cin >> a >> b >> c;
    switch (c) {
        case    '+'   : result = a + b; break;
        case    '-'   : result = a - b; break;
        case    '*'   : result = a * b; break;
        case    '/'   :
            if(b == 0) {
                cout <<"divide 0"<< endl;
                tag = 0;
                break;
            }
            result = a/b;
            break;
          default:
            tag = 0;
            cout <<"invalid operation"<< endl;
            break;                       //此处可以省略 break
    }
        if(tag)
        cout << result << endl;
    return 0;
}
```

(2) 下列程序的功能是输出 100 以内能被 3 整除且个位数为 6 的所有整数,请填空。

```
int main() {
    int i,j;
    for (i = 0;    ①    ;i++) {
        j = i * 10 + 6;
    if (    ②    ) continue;
    cout << j;
    }
    return 0;
}
```

【答】 正确的程序如下：

```
#include <iostream>
using namespace std;
int main() {
    int i,j;
    for (i = 0;i < 10;i++) {
        j = i * 10 + 6;
```

```
    if (j % 3!= 0) continue;
    cout << j;
    }
    return 0;
}
```

（3）斐波纳契数列有如下特点：第1、2个数都是1，从第3个数开始，每个数都是前两个数的和。下列程序的功能是求数列的前m(m>1)个数，按每行5个数输出。

```
int main() {
    int f1(1),f2(1),m;
    cin >> m;
    cout << f1 <<"\t"<< f2 <<"\t";
    for (int i = 2;i < m; i++) {
        ①
        ②
        cout << f2 <<"\t";
        if ( ③ ) cout << endl;
    }
    return 0;
}
```

【答】 正确的程序如下：

```
#include < iostream >
using namespace std;
int main() {
    int f1(1),f2(1),m;
    cin >> m;
    cout << f1 <<"\t"<< f2 <<"\t";
    for (int i = 2;i < m; i++) {
        f2 = f1 + f2;
        f1 = f2 - f1;
        cout << f2 <<"\t";
        if ((i + 1) % 5 == 0) cout << endl;
    }
    return 0;
}
```

5. 程序分析题

（1）写出并分析下列程序的运行结果：

```
#include < iostream >
using namespace std;
void swap(int x,int y) {
    int t;
    t = x,x = y,y = t;
    cout <<"&x:"<< &x <<", &y:"<< &y <<", &t:"<< &t << endl;
}
int main()
{
```

```
    int a = 3,b = 4;
    cout <<"&a:"<< &a <<", &b:"<< &b << endl;
    cout <<"a = "<< a <<",b = "<< b << endl;
    swap(a,b);
    cout <<"a = "<< a <<",b = "<< b << endl;
    return 0;
}
```

【答】 程序结果如下：

```
&a:0013FF7C, &b:0013FF78
a = 3,b = 4
&x:0013FF24, &y:0013FF28, &t:0013FF18
a = 3,b = 4
```

【注解】 函数参数以传值方式不改变实参的值。

(2) 下列程序有错误之处，请指出并改正，然后分析改正后的运行结果。

```
#include < iostream >
using namespace std;
int add(int x,int y,int z = 0) {
    cout <<"(int, int, int = 0)\t";
    return x + y; }
int add(int x, char y) {
    cout <<"(int, char)\t";
    return x + y; }
int main(){
    cout << add(9,8)<< endl;
    cout << add(9.0,8.0)<< endl;
    cout << add(9,8.0)<< endl;
    cout << add(9.0,8)<< endl;
    cout << add(9,'A')<< endl;
    cout << add('A','A' - '0')<< endl;
    return 0;
}
```

【答】

```
cout << add(9.0,8.0)<< endl;
cout << add(9,8.0)<< endl;
cout << add(9.0,8)<< endl;          //不能匹配,修改后的程序
```

```
#include < iostream >
using namespace std;
int add(int x,int y,int z = 0) {
    cout <<"(int, int, int = 0)\t";
    return x + y; }
int add(int x, char y) {
    cout <<"(int, char)\t";
    return x + y; }
double add(double x, double y) {
    cout <<"(double, double)\t";
```

```
    return x + y; }
double add(int x, double y) {
    cout <<"(int, double)\t";
    return x + y; }
double add(double x, int y) {
    cout <<"(double, int)\t";
    return x + y; }
int main(){
    cout << add(9,8)<< endl;
    cout << add(9.0,8.0)<< endl;
    cout << add(9,8.0)<< endl;
    cout << add(9.0,8)<< endl;
    cout << add(9,'A')<< endl;
    cout << add('A','A' - '0')<< endl;
     return 0;
}
```

程序结果如下：

```
(int, int, int = 0)      17
(double,double)          17
(int,double)             17
(double,int)             17
(int, char)              74
(int, int,int = 0)       82
```

6. 编程题

（1）摄氏温度与华氏温度的转换公式为：

$$c=\frac{5}{9}(f-32)$$

其中，c为摄氏温度，f为华氏温度，写出两者互相转换的表达式，将表达式放到程序中，以整数形式输入一种温度值，以整数形式输出转换后的温度值。

【答】 将华氏温度转换成摄氏温度的程序如下：

```
#include <iostream>
using namespace std;
int main()
{
    int c,f;
    cout <<"输入华氏温度:";
    cin >> f;
    cout <<"摄氏温度为:";
    c = ((5/9.0 * (f - 32)) + 0.5);          //将华氏温度转换成摄氏温度的表达式
    cout << c << endl;
    return 0;
}
```