

21世纪高等学校计算机专业核心课程规划教材

C++语言程序设计教程

(第3版)

习题解答与实验指导

杨进才 沈显君 胡珀 编著



清华大学出版社

21 世纪高等学校计算机专业核心课程规划教材

C++ 语言程序设计教程 (第 3 版) 习题解答与实验指导

杨进才 沈显君 胡 珀 编著

清华大学出版社
北京

内 容 简 介

本书是《C++ 语言程序设计教程(第3版)》的配套教学用书。全书包括3章内容:第1章包括《C++ 语言程序设计教程(第3版)》中的全部习题及其详细解答,题型涵盖了程序设计语言考试的主要题型(填空题、选择题、程序填空题、程序分析题、编程题);第2章介绍目前较为流行的C++语言三大开发环境——Visual C++、CodeBlocks 和 Linux C++ 的程序开发过程,包括编辑、编译、调试过程;第3章精心设计了11个上机实验题,并给出了参考答案,供读者循序渐进地学习与上机练习。

本书可以单独使用,适合作为高等学校C++语言的辅助教材和全国计算机等级考试的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C++ 语言程序设计教程(第3版)习题解答与实验指导/杨进才等编著. —北京:清华大学出版社,2015
21世纪高等学校计算机专业核心课程规划教材
ISBN 978-7-302-39044-2

I. ①C… II. ①杨… III. ①C语言—程序设计—高等学校—教学参考资料 IV. ①TP312

中国版本图书馆CIP数据核字(2015)第017138号

责任编辑:魏江江 王冰飞

封面设计:杨 兮

责任校对:白 蕾

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:16 字 数:398千字

版 次:2015年5月第1版 印 次:2015年5月第1次印刷

印 数:20001~22000

定 价:34.50元

产品编号:048544-01



前言

本书是《C++语言程序设计教程(第3版)》的配套教学用书,在习题解答部分给出了题目,因此可以单独使用,本书适合作为高校各专业C++语言的辅助教材和全国计算机等级考试的参考书。

学习计算机编程语言不仅要掌握语言本身的语法规则,更重要的是正确运用语言进行编程。本书通过各种题型的题目加强读者对概念、语法的掌握,通过对编程环境的介绍以及实验辅导教会读者如何编程。

本书共分3章:

第1章习题解答。本章不仅给出了习题的答案,而且对习题进行了详细的讲解。本章采用了多种题型,填空、选择、改错、写运行结果题供学习者理解巩固知识点使用,编程题供上机练习使用。题目与教材的知识点紧密结合,涵盖了《C++语言程序设计教程(第3版)》的所有知识点,其中,习题1~习题12涵盖了等级考试的所有知识点。题目中,填空题的难度稍微简单,其他题目有易有难,总体难度适中,既没有太简单的题目,也没有太难的题目。编程题中大多数是C++语言程序设计课程的经典题目,参考《C++语言程序设计教程(第3版)》以及其他课本,读者是可以独立完成的。

第2章C++开发环境使用指南。本章介绍了Windows系统环境下的Visual C++6.0、CodeBlocks两大主流C++编译器的使用,包括集成环境下的环境设置、编辑、编译、调试、运行、查帮助各个环节的介绍。

Linux是一个使用越来越多的操作系统。随着软件保护力度的加大,Linux将会成为主流的教学操作系统,Linux下的C++将会成为主要的C++编程环境。因此,本书介绍了Linux下的C++编辑器EMACS、编译器g++、调试器GDB的使用。

第3章C++上机实验指导。与其他编程语言一样,掌握C++编程的秘诀是“上机(编程)、上机、再上机”。本章配合课本的内容精心设计了11个实验,基本上每个实验对应教材的一章,每个实验给出了实验目的、实验内容、实验题目。实验题目选取的是经典的趣味性强的题目,以提高学生的编程兴趣。对实验题目给出了参考解答,并就题目的内容提出改进的思考。本章旨在通过不断的编程训练,提高学生的编程能力。

本书第1章部分内容以及第3章全部内容由沈显君博士编写,第1章部分内容以及第2章全部内容由杨进才教授编写,第1章部分内容由胡珀博士编写。编写教材是一项异常艰辛的工作,在此对沈显君、胡珀任劳任怨的合作表示感谢。

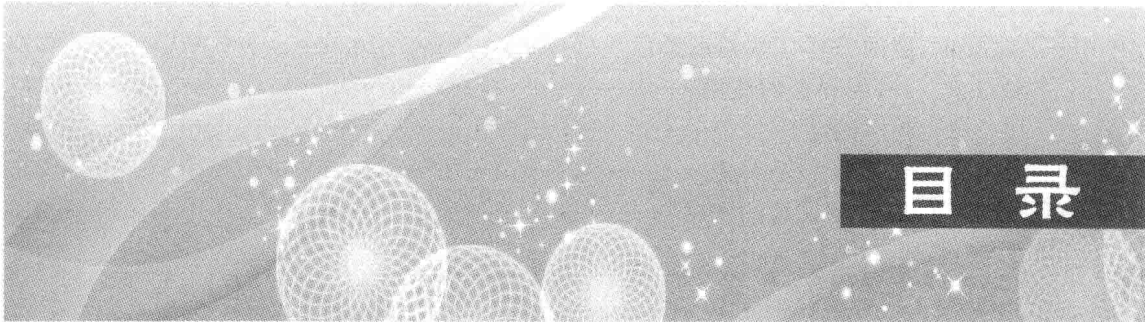


王敬华副教授不仅对全书的风格、内容提出了建设性的建议,对格式的编排等细节方面也提出了宝贵的意见,在此表示由衷的感谢。

由于本书编者水平有限,书中难免有错误、疏漏和不妥之处,恳请广大读者批评指正。

编 者

2015年1月于武昌桂子山



目 录

第 1 章 习题解答	1
1.1 习题 1 解答	1
1.2 习题 2 解答	7
1.3 习题 3 解答	28
1.4 习题 4 解答	48
1.5 习题 5 解答	52
1.6 习题 6 解答	82
1.7 习题 7 解答	111
1.8 习题 8 解答	137
1.9 习题 9 解答	156
1.10 习题 10 解答	167
1.11 习题 11 解答	179
1.12 习题 12 解答	183
第 2 章 C++ 开发环境使用指南	190
2.1 Visual C++ 使用指南	190
2.1.1 启动 Visual C++ 6.0	190
2.1.2 编辑源程序	190
2.1.3 程序的编译与运行	192
2.1.4 查帮助	196
2.1.5 程序的调试	197
2.1.6 多文档工程	199
2.2 CodeBlocks 使用指南	200
2.2.1 CodeBlocks 的安装与配置	200
2.2.2 编辑源程序	203
2.2.3 程序的编译与运行	206
2.2.4 查帮助	208
2.2.5 程序的调试	209
2.2.6 多文档工程	212

2.3	Linux C++使用指南	213
2.3.1	使用 EMACS 编辑源程序	213
2.3.2	g++编译器的使用	217
2.3.3	程序的运行	218
2.3.4	查帮助	218
2.3.5	GDB 调试器的使用	219
第 3 章	C++上机实验指导	222
3.1	上机实验题	222
3.1.1	实验 1 上机环境和 C++基础编程练习	222
3.1.2	实验 2 控制结构编程练习	222
3.1.3	实验 3 函数编程练习	223
3.1.4	实验 4 构造数据类型编程练习	223
3.1.5	实验 5 类与对象编程练习	223
3.1.6	实验 6 继承与派生编程练习	224
3.1.7	实验 7 多态性编程练习	224
3.1.8	实验 8 类模板编程练习	224
3.1.9	实验 9 输入/输出流与文件系统编程练习	224
3.1.10	实验 10 string 类字符串处理编程练习	225
3.1.11	实验 11 异常处理编程练习	225
3.2	上机实验题参考解答	225
3.2.1	实验 1 上机环境和 C++基础编程练习	225
3.2.2	实验 2 控制结构编程练习	226
3.2.3	实验 3 函数编程练习	227
3.2.4	实验 4 构造数据类型编程练习	229
3.2.5	实验 5 类与对象编程练习	234
3.2.6	实验 6 继承与派生编程练习	236
3.2.7	实验 7 多态性编程练习	238
3.2.8	实验 8 类模板编程练习	241
3.2.9	实验 9 输入/输出流与文件系统编程练习	243
3.2.10	实验 10 string 类字符串处理编程练习	245
3.2.11	实验 11 异常处理编程练习	246

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以防止命名冲突。

(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) 叙述机器语言、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;
    cout << "    * " << endl;
    return 0;
}
```

(2) 编写一程序,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 5 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 5 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、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实参。

(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. 10⁸'\ ' '\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】 10⁸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 $x+a\%3 * \text{static_cast}\langle \text{int}\rangle(x+y)\%2/4$ 的值为()。

A. 2.5

B. 2.75

C. 3.5

D. 0.0

【答】 A

【注解】 依次计算“ $a\%3=1$; $\text{static_cast}\langle \text{int}\rangle(x+y)=7$; $7\%2/4=0$; $x+0=2.5$ 。”

(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. $(\text{int})(f * 100 + 0.5) / 100.0$

D. $(\text{int})(f * 100 + 0.5) / 100$

【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. $\text{bool} = \text{char} < \text{int} \leq \text{long} = \text{float} < \text{double}$

B. $\text{char} < \text{bool} = \text{int} \leq \text{long} = \text{float} < \text{double}$

C. $\text{bool} < \text{char} < \text{int} < \text{long} < \text{float} < \text{double}$

D. $\text{bool} < \text{char} < \text{int} < \text{long} = \text{float} < \text{double}$

【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=20
 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() {
```