



## §. 基础知识题

要求:

- 1、完成本文档中所有的题目并写出分析、运行结果
- 2、无特殊说明，均使用VS2022编译即可
- 3、直接在本文件上作答，**写出答案/截图（不允许手写、手写拍照截图）**即可；填写答案时，为适应所填内容或贴图，**允许调整**页面的字体大小、颜色、文本框的位置等
  - ★ 贴图要有效部分即可，不需要全部内容
  - ★ 在保证一页一题的前提下，具体页面布局可以自行发挥，简单易读即可
  - ★ **不允许**手写在纸上，再拍照贴图
  - ★ **允许**在各种软件工具上完成（不含手写），再截图贴图
- 4、转换为pdf后提交
- 5、**9月15日前**网上提交本次作业（在“文档作业”中提交）



## §. 基础知识题

贴图要求：只需要截取输出窗口中的有效部分即可，如果全部截取/截取过大，则视为无效贴图

例：无效贴图

A screenshot of the Microsoft Visual Studio debug console window. The window is titled "Microsoft Visual Studio 调试控制台". The output text is: "Hello, world!", "D:\Workspace\VS2019-Demo\Debug\cpp-demo.exe (进程 7484)已退出, 代码为 0.", and "按任意键关闭此窗口. . .". The screenshot includes the title bar, window controls, and the status bar at the bottom, making it an invalid example of a screenshot.

例：有效贴图

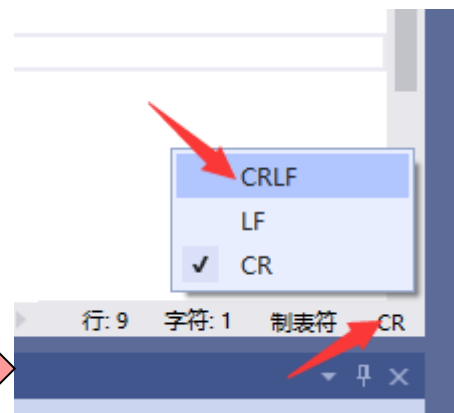
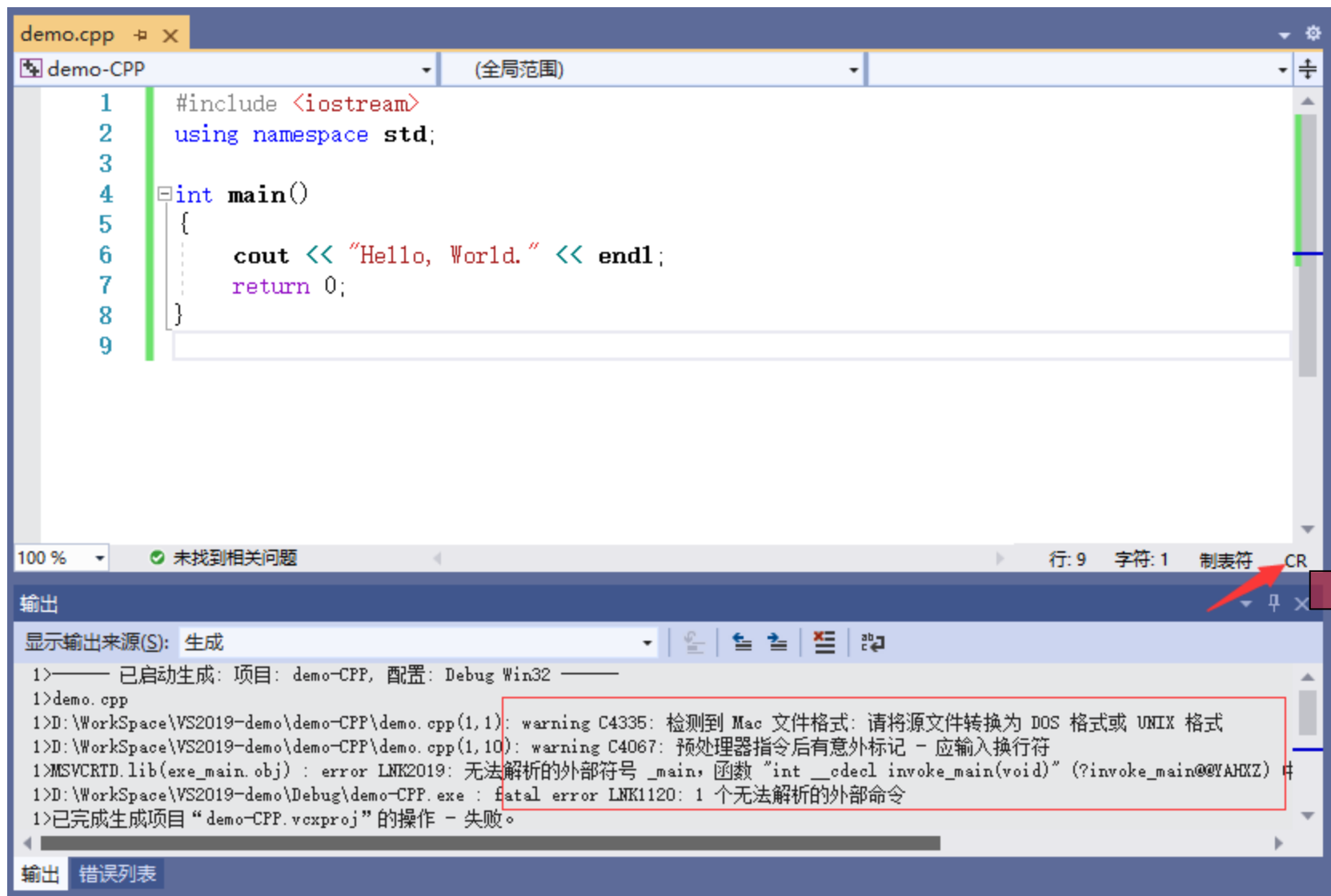
A screenshot of the Microsoft Visual Studio debug console window, showing only the output text: "Hello, world!". The title bar and status bar are not visible, making it a valid example of a screenshot.



## §. 基础知识题

附：用WPS等其他第三方软件打开PPT，将代码复制到VS2022中后，如果出现类似下面的**编译报错**，则观察源程序编辑窗

的右下角是否为CR，如果是，单击CR，在弹出中选择CRLF，再次CTRL+F5运行即可





## §. 基础知识题

1、十进制整数转二进制补码（仿照课件PDF的P. 22，写出具体步骤，包括绝对值、取反、+1）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001” 或 “11010100-00110001”）

A. -106 （假设为1字节整数，其中进制互转部分，直接写答案即可，不需要竖式除法/按权展开相加，下同）

数值	二进制表示	原码	补码
-106	1101010（绝对值）	01101010	10010101
			$\begin{array}{r} +) \quad \underline{\quad 1} \\ 10010110 \end{array}$



## § . 基础知识题

1、十进制整数转二进制补码（仿照课件PDF的P. 22，写出具体步骤，包括绝对值、取反、+1）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001” 或 “11010100-00110001”）

B. -206 （假设为2字节整数）

数值	二进制表示	原码	补码
-206	11001110（绝对值）	00000000 11001110	11111111 00110001
			+ )      1
			11111111 00110010



## §. 基础知识题

1、十进制整数转二进制补码（仿照课件PDF的P. 22，写出具体步骤，包括绝对值、取反、+1）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001”或“11010100-00110001”）

C. -206 （假设为4字节整数）

数值	二进制表示	原码
-206	11001110（绝对值）	00000000 00000000 00000000 11001110

补码

11111111	11111111	11111111	00110001
			+) 1
11111111	11111111	11111111	00110010



## §. 基础知识题

1、十进制整数转二进制补码（仿照课件PDF的P. 22，写出具体步骤，包括绝对值、取反、+1）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001” 或 “11010100-00110001”）

D. -4095 （假设为4字节整数）

数值	二进制表示	原码
-4095	1111 11111111 （绝对值）	00000000 00000000 00001111 11111111
补码		
	11111111 11111111 11110000 00000000	
		<u>          +1          </u>
	11111111 11111111 11110000 00000001	



## §. 基础知识题

1、十进制整数转二进制补码（仿照课件PDF的P.22，写出具体步骤，包括绝对值、取反、+1）

格式要求：多字节时，每8bit中间加一个空格或- (例：“11010100 00110001” 或 “11010100-00110001”)

E. 你的学号对应的int型十进制负数 (例1: 1234567  $\Rightarrow$  -7654321 / 1234000  $\Rightarrow$  -4321)

2152988 -8892512

数值	二进制表示
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

## 原码

-8892512   10000111   10110000   01100000   (绝对值)   00000000   10000111   10110000   01100000

## 补码

11111111 01111000 01001111 10011111

$$\begin{array}{r} +) \quad 1 \\ \hline \end{array}$$

11111111 01111000 01001111 10100000





## §. 基础知识题

2、二进制补码转十进制整数（只考虑有符号数，写出具体步骤，包括-1、取反、绝对值、加负号）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001”或“11010100-00110001”）

A. 1011 0111

补码-1	原码	二进制表示	数值
10110111	01001001	1001001（绝对值）	-73

-) 1  
10110110



## §. 基础知识题

2、二进制补码转十进制整数（只考虑有符号数，写出具体步骤，包括-1、取反、绝对值、加负号）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001”或“11010100-00110001”）

B. 1101 1101 1101 1110

补码-1	原码	二进制表示	数值
1101 1101 1101 1110	00100010 00100010	10001000100010 （绝对值）	-8738
<hr/>			
-) 1			
1101 1101 1101 1101			



## §. 基础知识题

2、二进制补码转十进制整数（只考虑有符号数，写出具体步骤，包括-1、取反、绝对值、加负号）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001”或“11010100-00110001”）

C. 1111 1111 1111 1111 1111 1110 1010 1110

补码-1

原码

1111 1111 1111 1111 1111 1110 1010 1110 00000000 00000000 00000001 01010010

-) 1

1111 1111 1111 1111 1111 1110 1010 1101

二进制表示

数值

101010010 （绝对值）

-338



## §. 基础知识题

2、二进制补码转十进制整数（只考虑有符号数，写出具体步骤，包括-1、取反、绝对值、加负号）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001”或“11010100-00110001”）

D. 1101 1101 0110 0000 0110 1011 1001 0000

补码-1

1101 1101 0110 0000 0110 1011 1001 0000

-) 1

1101 1101 0110 0000 0110 1011 1000 1111

二进制表示

100010100111111001010001110000 （绝对值）

原码

00100010 10011111 10010100 0111 0000

数值

-580883568



## §. 基础知识题

2、二进制补码转十进制整数（只考虑有符号数，写出具体步骤，包括-1、取反、绝对值、加负号）

格式要求：多字节时，每8bit中间加一个空格或-（例：“11010100 00110001”或“11010100-00110001”）

E. 学号对应的int型十进制负数的二进制补码形式（1. E的结果直接拿来当做本题初始数据即可）

补码-1

原码

11111111 01111000 01001111 10100000 00000000 10000111 10110000 01100000

-) 1

11111111 01111000 01001111 10011111

二进制表示

数值

10000111 10110000 01100000 （绝对值） -8892512



## §. 基础知识题

3、求出下列字符串的长度（要求仿照图例标出具体的字符分解）

"\r\n\t\\A\\t\x1b\"\1234\xft\x2f\33" = 15

A. "\b\\nr\v\384\x3fr\2a\" \r\\a\v\f" = 18  
- - - - -



## §. 基础知识题

3、求出下列字符串的长度（要求仿照图例标出具体的字符分解）

"\r\n\t\\A\\t\x1b\"1234\xft\x2f\33" = 15

B. "\138\xa2\214\x6w\383\x65\042\xd5\257\x3e\1325\x6a\175\x2e" = 19



## §. 基础知识题

3、求出下列字符串的长度（要求仿照图例标出具体的字符分解）

"\r\n\t\\A\\t\x1b\"1234\xft\x2f\33" = 15

这两个，第一个是正确的，第二个有问题，请构造测试程序验证，并将构造的测试程序及相应的error或warning信息的截图贴在文档中；对有warning的测试程序的运行结果给出长度分析，对有error的测试程序给出你的理解（主要是对比从八进制和十六进制转义的差异）

C. "\t\\r\nv\293\23456f\"r\\av\f"  
"\t\\r\nv\293\x23456f\"r\\av\f"

```
1 #include<iostream>
2 using namespace std;
3
4 int main()
5 {
6     cout << strlen("\t\\r\nv\293\23456f\"r\\av\f") << endl;
7
8     return 0;
9 }
```

Microsoft Visual Studio 调试控制台

```
C:\Users\user\Desktop\homework\Debug\hello world.exe <进程 22252>已退出，代码为 0。
按任意键关闭此窗口。 . . .
```

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5     cout << strlen("\t\\r\nv\293\x23456f\"r\\av\f") << endl;
6     return 0;
7 }
```

✖ C2022 "2311535": 对字符来说太大

十六进制转义为“xhh”形式，x 后面都会被认为是十六进制数。  
但是x后若存在三位及以上十六进制数，则会超过ASCII码上限，也就是“对字符来说太大”





## §. 基础知识题

3、求出下列字符串的长度（要求仿照图例标出具体的字符分解）

"\r\n\t\\A\t\x1b\"1234\xft\x2f\33" = 15

这两个，都是不完全正确的，请构造测试程序验证，并将你构造的测试程序及相应的error或warning信息的截图贴在文档中；对有warning的测试程序的运行结果给出长度分析，对有error的测试程序给出你的理解

D. "\9876"

"\\*321"

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      cout << strlen("\9876") << endl;
6      return 0;
7  }
```

! C4129 "9": 不可识别的字符转义序列

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      cout << strlen("\*321") << endl;
6      return 0;
7  }
```

! C4129 "\*": 不可识别的字符转义序列

9, \* 不应属于八进制



## §. 基础知识题

3、求出下列字符串的长度（要求仿照图例标出具体的字符分解）

"\r\n\t\\A\\t\x1b\"1234\xft\x2f\33" = 15

这两个，都是不完全正确的，请构造测试程序验证，并将你构造的测试程序及相应的error或warning信息的截图贴在文档中；对有warning的测试程序的运行结果给出长度分析，对有error的测试程序给出你的理解

E. "\xg321"  
"\x\*321"

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      cout << strlen("\xg321") << endl;
6      return 0;
7  }
```

✗ C2153 整数文本必须至少具有一位数  
abc E0022 无效的十六进制数

g, \* 不属于十六进制

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      cout << strlen("\x*321") << endl;
6      return 0;
7  }
```

✗ C2153 整数文本必须至少具有一位数  
abc E0022 无效的十六进制数



## §. 基础知识题

4、给出下列程序段中变量b的值（要综合参考课件P. 45-51 和 P. 86-89，给出包含整型提升+丢弃的过程，具体见下）

例: short a=1;  
short b=a-2;

Step1: b=a-2, 得b二进制补码形式

a = 00000000 00000000 00000000 00000001 → a （红色表示整型提升的填充位）  
-) 2 = 00000000 00000000 00000000 00000010 → 2

-----  
11111111 11111111 11111111 11111111 → a-2(int型)  
b = ~~11111111 11111111~~ 11111111 11111111 → b=a-2(二进制补码形式, 删除线表示丢弃的位数)

Step2: 求b的十进制表示

(1) 减一      11111111 11111111  
      -) 00000000 00000001

-----  
11111111 11111110

(2) 取反      00000000 00000001

(3) 绝对值    1

(4) 加负号   -1 (b的十进制表示形式)

本页不用作答



## §. 基础知识题

4、给出下列程序段中变量b的值（要综合参考课件P. 45-51 和 P. 86-89，给出包含整型提升+丢弃的过程）

```
A. short a=32760;  
   short b=a+14;
```

Step1: b=a+14, 得b二进制补码形式

```
a = 00000000 00000000 01111111 11111000 -> a  
- ) 14 = 00000000 00000000 00000000 00001110 -> 14  
-----  
      00000000 00000000 10000000 00000110 -> a+14(int型)  
b = 00000000 00000000 10000000 00000110 -> b=a+14
```

Step2: 求b的十进制表示

```
(1) 减一      10000000 00000110  
      - ) 00000000 00000001  
      -----
```

```
      10000000 00000101
```

```
(2) 取反      01111111 11111010
```

```
(3) 绝对值    65018
```

```
(4) 加负号    -65018 (b的十进制表示形式)
```



## §. 基础知识题

4、给出下列程序段中变量b的值（要综合参考课件P. 45-51 和 P. 86-89，给出包含整型提升+丢弃的过程）

```
B. unsigned short a=65530;  
   short b=a;
```

```
a = 11111111 11111010 → a(unsigned short型整型提升)  
b = 11111111 11111010 → b=a(short型)
```

b的十进制表示:

```
(1) 减一 11111111 11111010  
      -) 00000000 00000001
```

-----

11111111 11111001

(2) 取反            00000000 00000110

(3) 绝对值          12

(4) 加负号          -12 (b的十进制表示形式)



## §. 基础知识题

4、给出下列程序段中变量b的值（要综合参考课件P. 45-51 和 P. 86-89，给出包含整型提升+丢弃的过程）

```
C. short a=-8191;  
   int b=a;
```

a = 11111111 11111111 11100000 00000001 → a(short型整型提升)

b = 11111111 11111111 11100000 00000001 → b=a(int型)

b的十进制表示:

(1) 减一	11111111	11111111	11100000	00000001
	-)	00000000	00000000	00000000

---

	11111111	11111111	11100000	00000000
--	----------	----------	----------	----------

(2) 取反	00000000	00000000	00011111	11111111
--------	----------	----------	----------	----------

(3) 绝对值	8191
---------	------

(4) 加负号	-8191 (b的十进制表示形式)
---------	-------------------



## §. 基础知识题

4、给出下列程序段中变量b的值（要综合参考课件P. 45-51 和 P. 86-89，给出包含整型提升+丢弃的过程）

```
D. unsigned short a=65530;  
   long long int b=a;
```

```
a = 00000000 00000000 00000000 00000000 00000000 00000000 11111111 11111010
```

-> a(unsigned short型整型提升)

```
b = 00000000 00000000 00000000 00000000 00000000 00000000 11111111 11111010
```

-> b=a(long long型)

b的十进制表示:

由于b是正数，其补码等于原码



## §. 基础知识题

4、给出下列程序段中变量b的值（要综合参考课件P. 45-51 和 P. 86-89，给出包含整型提升+丢弃的过程）

```
E. long long int a=4207654321;  
   int b=a;
```

```
a = 00000000 00000000 00000000 00000000 11111010 11001011 10110101 10110001
```

-> a(long long型整型丢弃)

```
b = 11111010 11001011 10110101 10110001 -> b=a(int型)
```

b的十进制表示:

(1) 减一	11111010 11001011 10110101 10110001
-)	00000000 00000000 00000000 00000001

-----

11111010 11001011 10110101 10110000

(2) 取反	00000101 00110100 01001010 01001111
(3) 绝对值	1235035805
(4) 加负号	-1235035805 (b的十进制表示形式)





## §. 基础知识题

4、给出下列程序段中变量b的值（要综合参考课件P. 45-51 和 P. 86-89，给出包含整型提升+丢弃的过程）

F. long a=-4207654321; //提示：本题先确定 -4207654321 什么类型，a是多少，才能进行b=a的计算  
unsigned short b=a;

a = 00000101 00110100 01001010 01001111 -> a

b = 01001010 01001111 -> b=a(unsigned short型)

b的十进制表示：

由于b是unsigned short型，数据均非负，其补码等于原码



## § . 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

A.  $21 / 2 + 47 \% 3 - 1.3 + 3.5 * 2$

表达式一共有6个运算符，因此计算的6个步骤分别是：

步骤①：  $21/2=10$

步骤②：  $47\%3=2$

步骤③：  $10+2=12$

步骤④：  $12-1.3=10.7$

步骤⑤：  $3.5*2=7.0$

步骤⑥：  $10.7+7.0=17.7$



## §. 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

A.  $21 / 2 + 47 \% 3 - 1.3 + 3.5 * 2$

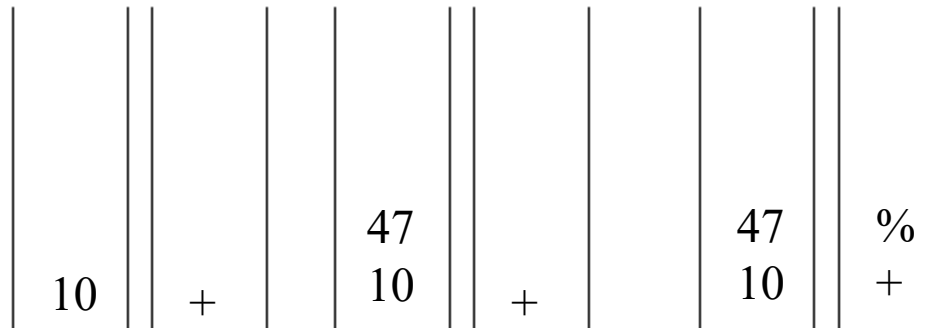
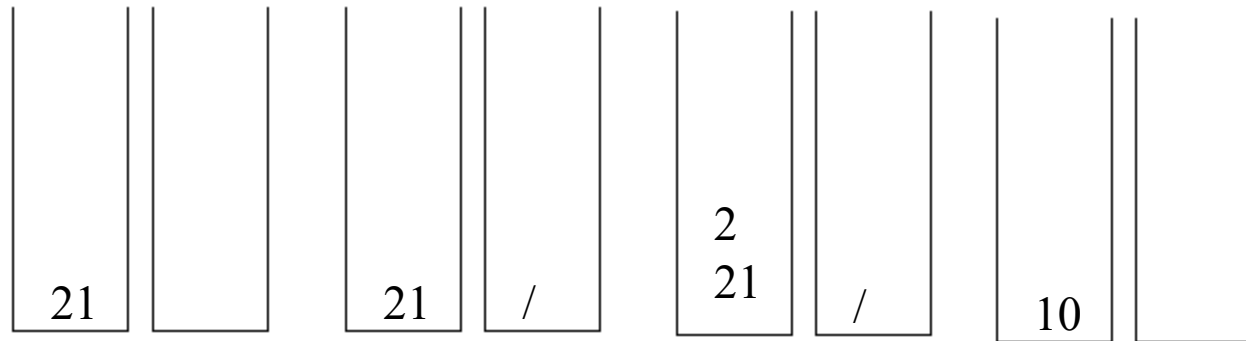


目前准备进栈的运算符如箭头所示，画出从初始分析到该运算符直到该运算符进栈的整个过程

例：以课件的第2个+为例，就是P. 69-P. 71三张图；以-为例，就是P. 75-P. 79五张图

注：尽量画在本页上，从左到右依次排开即可，如果实在空间不够，可以加页

+





## §. 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

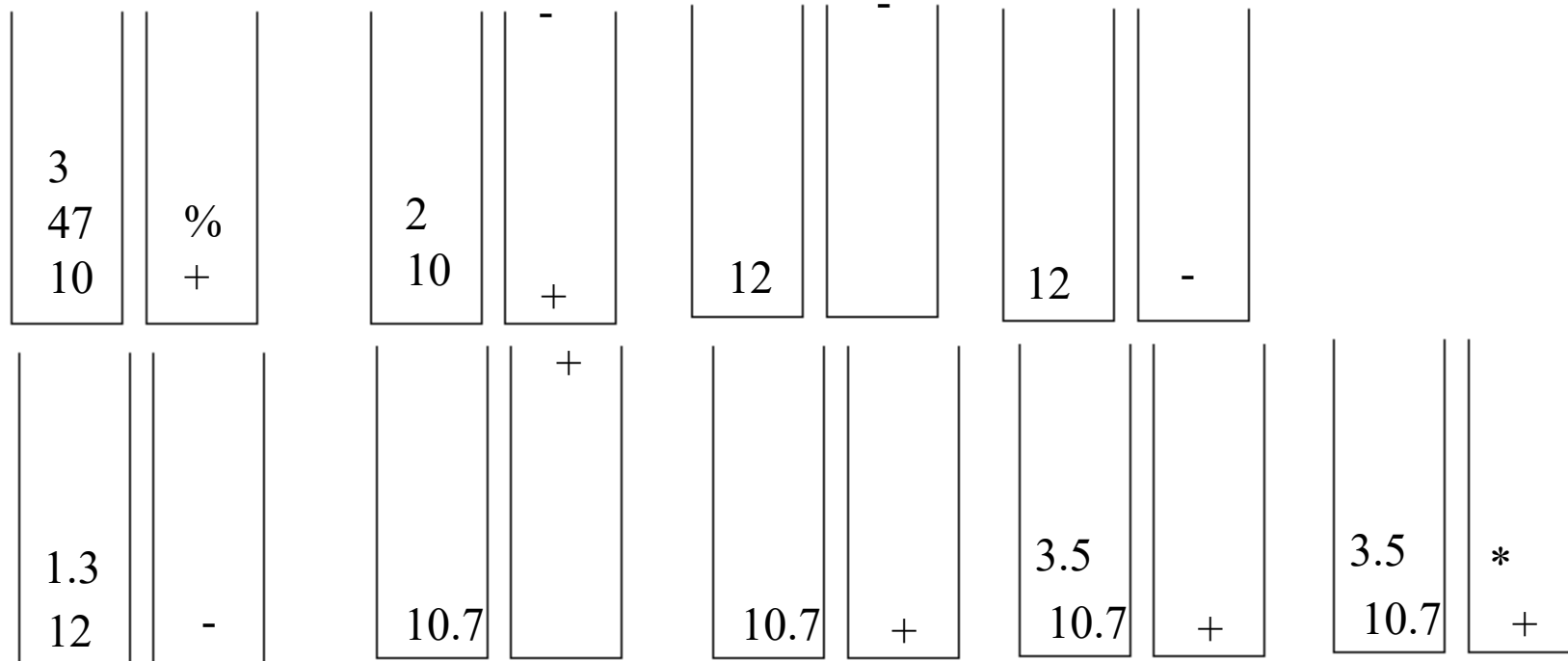
A.  $21 / 2 + 47 \% 3 - 1.3 + 3.5 * 2$



目前准备进栈的运算符如箭头所示，画出从初始分析到该运算符直到该运算符进栈的整个过程

例：以课件的第2个+为例，就是P. 69-P. 71三张图；以-为例，就是P. 75-P. 79五张图

注：尽量画在本页上，从左到右依次排开即可，如果实在空间不够，可以加页





## § . 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

B.  $a = 3 * 5$  ,  $a = b = 6 * 4$  （假设所有变量均为int型）

表达式一共有6个运算符，因此计算的6个步骤分别是：

步骤①：  $3*5=15$

步骤②：  $a=15$

步骤③：  $6*4=24$

步骤④：  $b=24$

步骤⑤：  $a=24$

步骤⑥：  $15, 24=24$



## §. 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

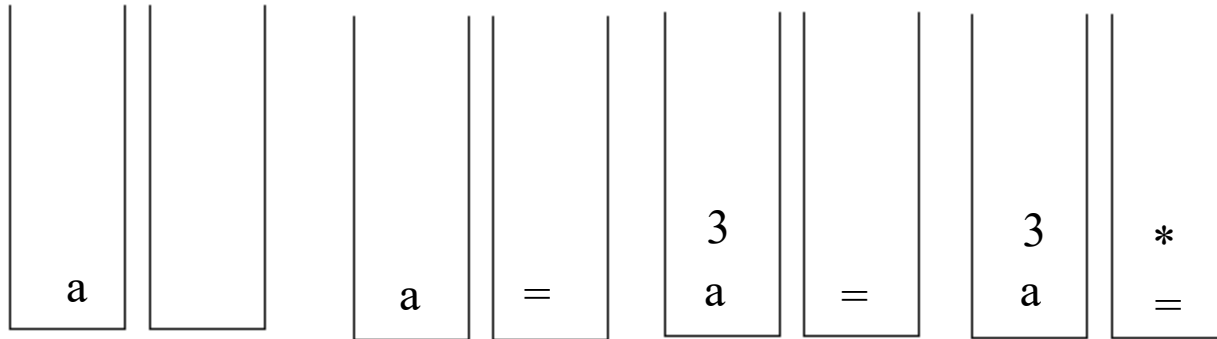
B.  $a = 3 * 5$  ,  $a = b = 6 * 4$  (假设所有变量均为int型)



目前准备进栈的运算符如箭头所示，画出从初始分析到该运算符直到该运算符进栈的整个过程

例：以课件的第2个+为例，就是P. 69-P. 71三张图；以-为例，就是P. 75-P. 79五张图

注：尽量画在本页上，从左到右依次排开即可，如果实在空间不够，可以加页





## §. 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

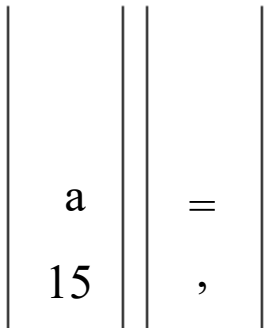
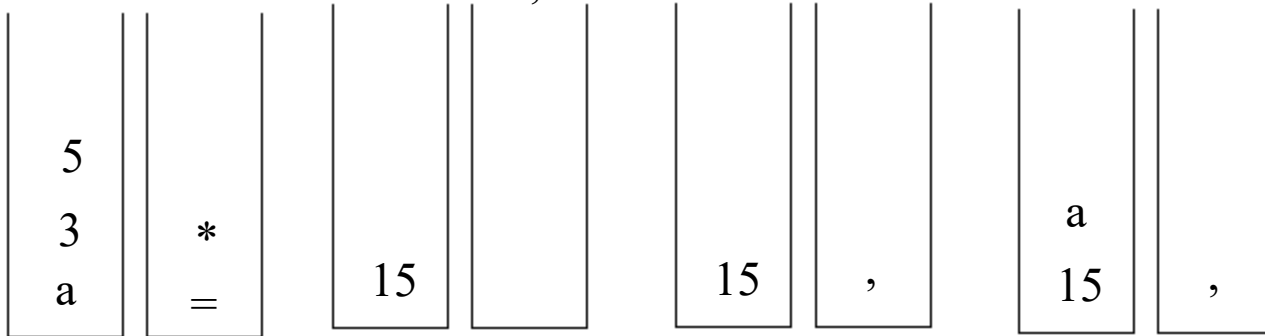
B.  $a = 3 * 5$  ,  $a = b = 6 * 4$  (假设所有变量均为int型)



目前准备进栈的运算符如箭头所示，画出从初始分析到该运算符直到该运算符进栈的整个过程

例：以课件的第2个+为例，就是P. 69-P. 71三张图；以-为例，就是P. 75-P. 79五张图

注：尽量画在本页上，从左到右依次排开即可，如果实在空间不够，可以加页





## §. 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

C.  $a + (b - 3 * (a + c) - 2) \% 3$  (假设所有变量均为int型)

(本题提示：将左右小括号分开处理，

1、“(”**进栈前**优先级最高，**进栈后**优先级最低；

2、“)”优先级最低，因此要将栈中压在“(”之上的全部运算符都计算完成，随后和“(”成对消除即可

表达式一共有\_\_10\_\_个运算符，因此计算的\_6\_个步骤分别是：

步骤①： $a+c$

步骤②： $3*①$

步骤③： $y-②$

步骤④： $③-2$

步骤⑤： $④\%3$

步骤⑥： $a+⑤$

后面自行添加，主要是对()的理解





## §. 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

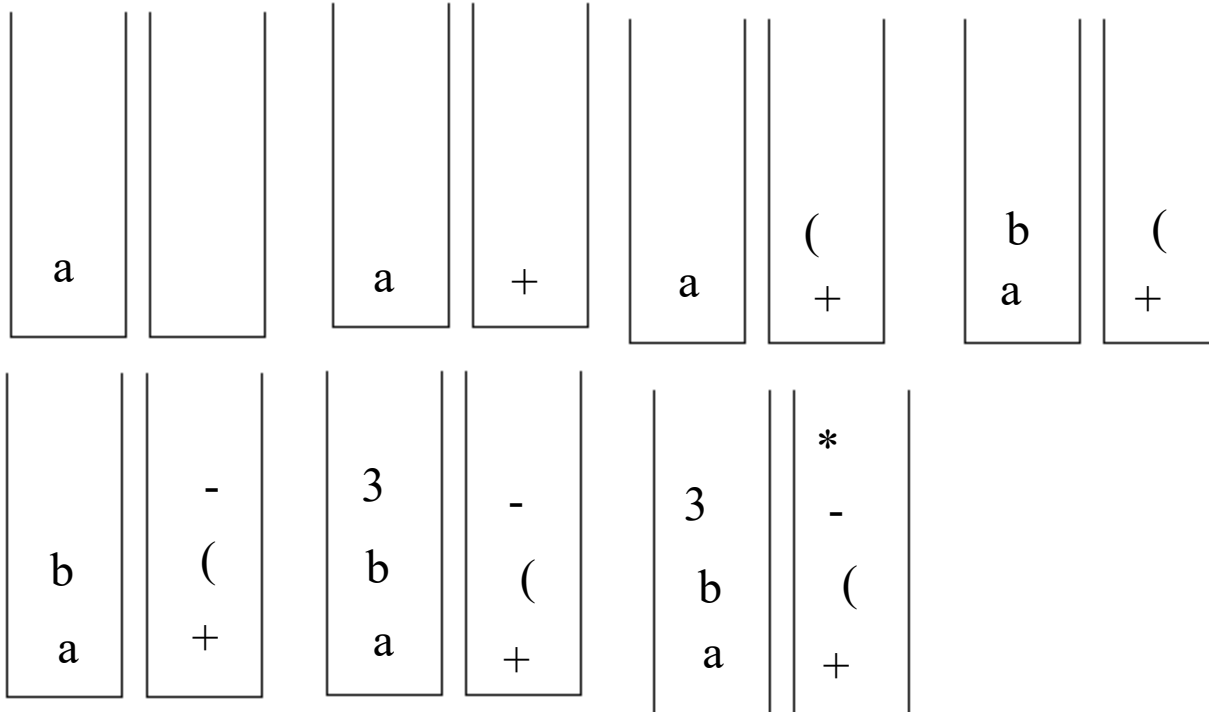
C.  $a + (b - 3 * (a + c) - 2) \% 3$  (假设所有变量均为int型)

(本题提示：将左右小括号分开处理，

1、“(”**进栈前**优先级最高，**进栈后**优先级最低；

2、“)”优先级最低，因此要将栈中压在“(”之上的全部运算符都计算完成，随后和“(”成对消除即可

目前准备进栈的运算符如箭头所示，画出从初始分析到该运算符直到该运算符进栈的整个过程





## §. 基础知识题

5、仿照课件PDF的P. 65-85，用栈方式给出下列表达式的求解过程

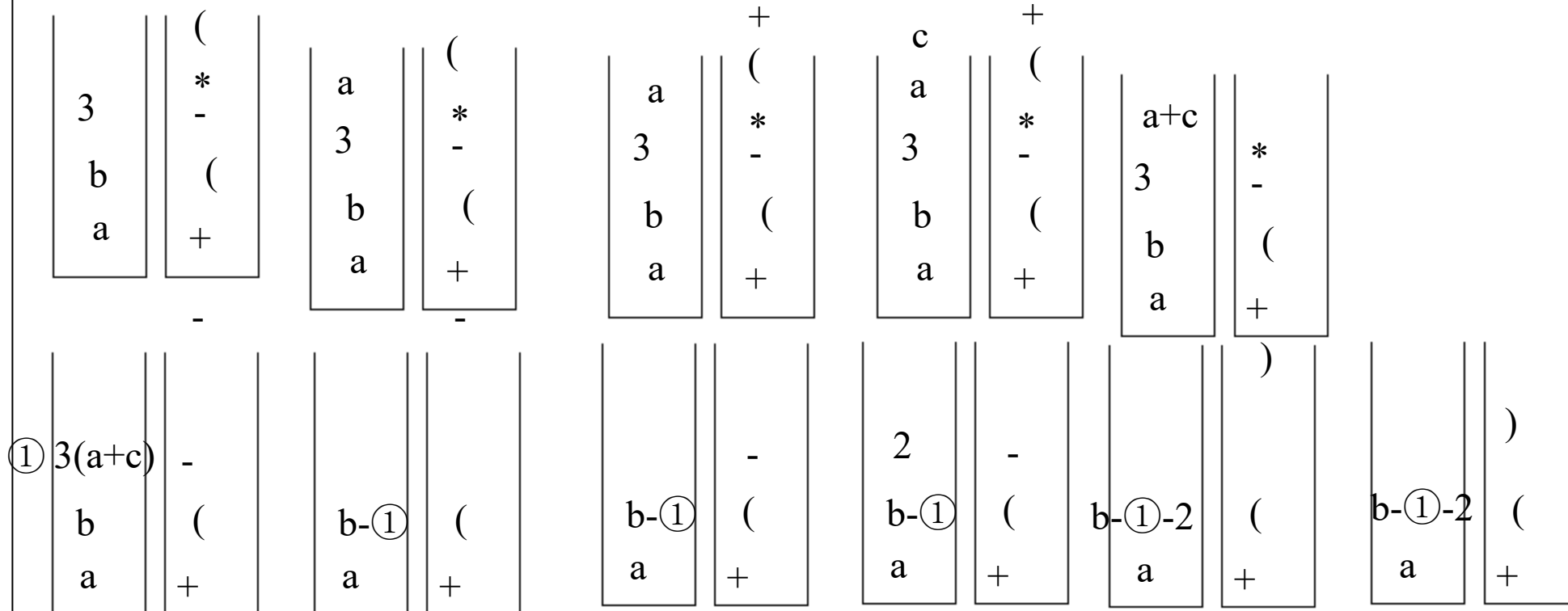
C.  $a + (b - 3 * (a + c) - 2) \% 3$  (假设所有变量均为int型)

(本题提示：将左右小括号分开处理，

1、“(进栈前)优先级最高, (进栈后)优先级最低;

2、“)”优先级最低，因此要将栈中压在“(“之上的全部运算符都计算完成，随后和“(“成对消除即可

目前准备进栈的运算符如箭头所示，画出从初始分析到该运算符直到该运算符进栈的整个过程





## §. 基础知识题

6、求表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图，示例见下）

例： $2LL - 32L * \text{int}(11.7) + 2.3f$

- |   |    |        |            |
|---|----|--------|------------|
| (1) $\text{int}(11.7)$                    | => | 11     | int型       |
| (2) $32L * \text{int}(11.7)$              | => | 352    | long型      |
| (3) $2LL - 32L * \text{int}(11.7)$        | => | -350   | long long型 |
| (4) $2LL - 32L * \text{int}(11.7) + 2.3f$ | => | -347.7 | float型     |

```
demo.cpp  x
demo.cpp  (全局范围)

1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      cout << 2LL - 32L * int(11.7) + 2.3f << endl;
6      cout << typeid(2LL - 32L * int(11.7) + 2.3f).name() << endl;
7      return 0;
8  }
9

Microsoft Visual Studio 调试控制台
-347.7
float
```

本页不用作答



## §. 基础知识题

6、求表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图）

A.  $a = 2 * 5$  ,  $a = b = 4 * 4$

（写验证程序时，假设所有变量均为int型）

(1) $2*5$	=>	10	int型
(2) $a=2*5$	=>	10	int型
(3) $4*4$	=>	16	int型
(4) $b=4*4$	=>	16	int型
(5) $a=b=4*4$	=>	16	int型
(6) $a = 2 * 5$ , $a = b = 4 * 4$	=>	16	int型

```
#include<iostream>
using namespace std;
int main()
{
    int a, b;
    cout << (a = 2 * 5, a = b = 4 * 4) << endl;
    cout << typeid(a = 2 * 5, a = b = 4 * 4).name() << endl;
    return 0;
}
```

Microsoft Visual Studio 调试控制台

```
16
int
```



## §. 基础知识题

6、求表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图）

B.  $a + (b - 2 * (a + c) - 3) \% 4$  （写验证程序时，假设所有变量均为int型，**abc的值自定义即可**）  
 $a=1, b=2, c=3;$

(1) $a+c$	=>	4	int型
(2) $2 * (a + c)$	=>	8	int型
(3) $b - 2 * (a + c)$	=>	-6	int型
(4) $b - 2 * (a + c) - 3$	=>	-9	int型
(5) $(b - 2 * (a + c) - 3) \% 4$	=>	-1	int型
(6) $a + (b - 2 * (a + c) - 3) \% 4$	=>	0	int型

```
#include<iostream>
using namespace std;
int main()
{
    int a=1, b=2, c=3;
    cout << a + (b - 2 * (a + c) - 3) % 4 << endl;
    cout << typeid(a + (b - 2 * (a + c) - 3) % 4).name() << endl;
    return 0;
}
```

Microsoft Visual Studio 调试控制台

```
0
int
```



## §. 基础知识题

6、求表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图）

C.  $2.5F * 2LU + 3U * 5LL - 'b'$

- |                                   |    |     |           |
|-----------------------------------|----|-----|-----------|
| (1) $2.5F * 2LU$                  | => | 5.0 | float型    |
| (2) $3U * 5LL$                    | => | 15  | longlong型 |
| (3) $2.5F * 2LU + 3U * 5LL$       | => | 20  | float型    |
| (4) $2.5F * 2LU + 3U * 5LL - 'b'$ | => | -78 | float型    |

```
#include<iostream>
using namespace std;
int main()
{
    cout << 2.5F * 2LU + 3U * 5LL - 'b' << endl;
    cout << typeid(2.5F * 2LU + 3U * 5LL - 'b').name() << endl;
    return 0;
}
```

Microsoft Visual Studio 调试控制台

```
-78
float
```



## §. 基础知识题

6、求表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图）

D.  $7LU \% 3 + 13LL \% 7 + 2.3$

- |                                  |    |     |                |
|----------------------------------|----|-----|----------------|
| (1) $7LU \% 3$                   | => | 1   | unsigned long型 |
| (2) $13LL \% 7$                  | => | 6   | longlong型      |
| (3) $7LU \% 3 + 13LL \% 7$       | => | 7   | longlong型      |
| (4) $7LU \% 3 + 13LL \% 7 + 2.3$ | => | 9.3 | float型         |

```
#include<iostream>
using namespace std;
int main()
{
    cout << 7LU % 3 + 13LL % 7 + 2.3 << endl;
    cout << typeid(7LU % 3 + 13LL % 7 + 2.3).name() << endl;
    return 0;
}
```

Microsoft Visual Studio 调试控制台

```
9.3
double
```



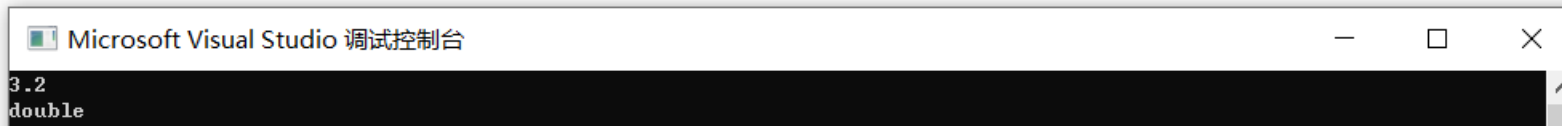
## §. 基础知识题

6、求表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图）

E.  $3.2 + 11 \% 3 * \text{static\_cast}\langle\text{unsigned int}\rangle(1.8F + 2) \% 3 * 4.2F$

- |  |                    |
|--|--------------------|
| (1) $11 \% 3$  | => 2 int型          |
| (2) $1.8F + 2$   | => 3.8 float型      |
| (3) $\text{static\_cast}\langle\text{unsigned int}\rangle(1.8F + 2)$                             | => 3 unsigned int型 |
| (4) $11 \% 3 * \text{static\_cast}\langle\text{unsigned int}\rangle(1.8F + 2)$                   | => 6 unsigned int型 |
| (5) $11 \% 3 * \text{static\_cast}\langle\text{unsigned int}\rangle(1.8F + 2) \% 3$              | => 0 unsigned int型 |
| (6) $11 \% 3 * \text{static\_cast}\langle\text{unsigned int}\rangle(1.8F + 2) \% 3 * 4.2F$       | => 0 float型        |
| (7) $3.2 + 11 \% 3 * \text{static\_cast}\langle\text{unsigned int}\rangle(1.8F + 2) \% 3 * 4.2F$ | => 3.2 double型     |

```
#include<iostream>
using namespace std;
int main()
{
    cout << 3.2 + 11 % 3 * static_cast<unsigned int>(1.8F + 2) % 3 * 4.2F << endl;
    cout << typeid(3.2 + 11 % 3 * static_cast<unsigned int>(1.8F + 2) % 3 * 4.2F).name() << endl;
    return 0;
}
```







## §. 基础知识题

6、求表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图）

F.  $\text{long}(3.78 + 1.33) \% 2 + (\text{int})1.5 \% 7\text{U} - 'Y' * 6\text{L}$

- |  |               |                |
|--|---------------|----------------|
| (1) $\text{long}(3.78 + 1.33)$   | => 5          | long型          |
| (2) $\text{long}(3.78 + 1.33) \% 2$  | => 1          | long型          |
| (3) $(\text{int})1.5$  | => 1          | int型           |
| (4) $(\text{int})1.5 \% 7\text{U}$   | => 1          | unsigned int型  |
| (5) $\text{long}(3.78 + 1.33) \% 2 + (\text{int})1.5 \% 7\text{U}$                   | => 2          | unsigned long型 |
| (6) $'Y' * 6\text{L}$  | => 534        | long型          |
| (7) $\text{long}(3.78 + 1.33) \% 2 + (\text{int})1.5 \% 7\text{U} - 'Y' * 6\text{L}$ | => 4294966764 | unsigned long型 |

```
#include<iostream>
using namespace std;
int main()
{
    cout << (long(3.78 + 1.33) % 2 + (int)1.5 % 7U - 'Y' * 6L) << endl;
    cout << typeid(long(3.78 + 1.33) % 2 + (int)1.5 % 7U - 'Y' * 6L).name() << endl;
    return 0;
}
```

Microsoft Visual Studio 调试控制台

```
4294966764
unsigned long
```



## §. 基础知识题

7、求复合赋值表达式的值（要求给出计算过程、每步计算结果中变量的值、对应的验证程序及结果截图，示例见下）

假设 `int a=5, n = 12;`

例: `a += n`

$\Rightarrow a = a + n$

(1) `a + n`      `a=5` `n=12` 和17存放在中间变量中

(2) `a =` 和      `a=17` `n=12`

```
demo.cpp x
demo-CPP (全局范围)
2 using namespace std;
3 int main()
4 {
5     int a = 5, n = 12;
6     a += n;
7     cout << a << ' ' << n << endl;
8     return 0;
9 }
```

Microsoft Visual Studio 调试控制台

17 12

本页不用作答



## §. 基础知识题

7、求复合赋值表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图，具体见下）

假设 `int a=2, n = 3;`

A. `a -= a + n`

$\Rightarrow a = a - (a + n)$

- (1) `a + n`      `a=2 n=3` 和5存放在中间变量中
- (2) `a -` 和      `a=2` 和`=5` 差`-3`存放在中间变量中
- (3) `a =` 差      `a=-3 n=3`

```
1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      int a = 2, n = 3;
6      a -= a + n;
7      cout << a << ' ' << n << endl;
8      return 0;
9  }
```

Microsoft Visual Studio 调试控制台  
-3 3



## §. 基础知识题

7、求复合赋值表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图，具体见下）

假设 `int a=2, n = 3;`

B. `a += n += 3`

$\Rightarrow a = a + (n = n + 3)$

(1) `n + 3`                      `a=2 n=3 n+3=6` 和6存放在中间变量中

(2) `n = 和`                      `a=2 n=6`

(3) `a + 6`                      `a=2 n=6` 和24存放在中间变量中

(4) `a = 和`                      `a=8 n=6`

```
#include<iostream>
using namespace std;
int main()
{
    int a = 2, n = 3;
    a += n += 3;
    cout << a << ' ' << n << endl;
    return 0;
}
```

Microsoft Visual Studio 调试控制台

8 6



## §. 基础知识题

7、求复合赋值表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图，具体见下）

假设 `int a=2, n = 3;`

C. `a *= a += a /= a`

=> `a = a *(a = a +(a = a / a))`

(1) `a / a`      `a=2 a/a=1` 商1存放在中间变量中

(2) `a = 商`      `a=1`

(3) `a + a`      `a=1 a+a=2` 和2存放在中间变量中

(4) `a = 和`      `a=2`

(5) `a * a`      `a=2 a*a=4` 积4存放在中间变量中

(6) `a = 积`      `a=4`

```
#include<iostream>
using namespace std;
int main()
{
    int a = 2, n = 3;
    a *= a += a /= a;
    cout << a << ' ' << n << endl;
    return 0;
}
```

Microsoft Visual Studio 调试控制台

4 3



## §. 基础知识题

7、求复合赋值表达式的值（要求给出计算过程、每步计算结果及数据类型、对应的验证程序及结果截图，具体见下）

假设 `int a=2, n = 3;`

D. `a %= n %= 3` 本题需要解释，为什么编译不报错，但运行无输出、返回代码为负值、且运行时间比7. ABC长（无法理解或说清楚原因的，给出合理猜测也可）

`a = a %(n = n % 4)`

`n%3=0`，并通过赋值运算符将0赋给了`n`，`a=a%0`，除数不能为0而导致程序出错。但编译时电脑并不知道会产生%0的步骤，所以未报错。`-1073741676`是十六进制值`0xC0000094`，查阅资料指：`EXCEPTION_INT_DIVIDE_BY_ZERO`。可能因为发生错误，电脑产生保护机制使程序中止，所以无输出；运行时间长；没能执行`return 0`，而返回错误代码。

```
demo.cpp x
demo.cpp (全局范围)
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int a = 2, n = 3;
6      a %= n %= 3;
7      cout << a << ' ' << n << endl;
8      return 0;
9  }
10
11
```

选择 Microsoft Visual Studio 调试控制台

D:\Workspace\VS2022-demo\Debug\demo-cpp.exe (进程 3004) 已退出，代码为 **-1073741676**。  
按任意键关闭此窗口。 . . .