



§. 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、**3月12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

```
Microsoft Visual Studio 调试控制台
Hello, world!
D:\WorkSpace\VS2019-Demo\Debug\cpp-demo.exe (进程 7484)已退出, 代码为 0。
按任意键关闭此窗口. . .
```

例：有效贴图

```
Microsoft Visual Studio 调试控制台
Hello, world!
```



§. 基础知识题 – 浮点数机内存储格式(IEEE 754)理解

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

The screenshot shows the Visual Studio 2022 IDE interface. In the top-left, there's a code editor window titled "demo.cpp" containing the following C++ code:

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

In the bottom-right corner of the code editor, there's a status bar showing "行: 4 字符: 2 空格 SYS CR". A red arrow points to the "CR" button in this status bar, which has a dropdown menu open with options "CRLF", "LF", and "CR". "CRLF" is selected.

Below the code editor is the "输出" (Output) window. It displays the build log:

```
生成开始于 22:23...
1>—— 已启动生成: 项目: demo-cpp, 配置: Debug Win32 ——
1>demo.cpp
1>D:\WorkSpace\VS2022-demo\demo-cpp\demo.cpp(1,1): warning C4335: 检测到 Mac 文件格式: 请将源文件转换为 DOS 格式或 UNIX 格式
1>D:\WorkSpace\VS2022-demo\demo-cpp\demo.cpp(1,10): warning C4067: 预处理器指令后有意外标记 - 应输入换行符
1>MSVCRTD.lib(exe_main.obj) : error LNK2019: 无法解析的外部符号 _main, 函数 "int __cdecl invoke_main(void)" (?invoke_main@@YAHKZ) 中引用了该符号
1>D:\WorkSpace\VS2022-demo\Debug\demo-cpp.exe : fatal error LNK1120: 1 个无法解析的外部命令
1>已完成生成项目“demo-cpp.vcxproj”的操作 - 失败。
===== 生成: 0 成功, 1 失败, 0 最新, 0 已跳过 =====
===== 生成于 22:23 完成, 耗时 01.132 秒 =====
```

A red box highlights the warning and error messages in the output window. At the bottom of the screen, there's a navigation bar with tabs for "错误列表" (Error List) and "输出" (Output), with "输出" currently selected.



§. 基础知识题

1、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

$$\begin{array}{r} a = \textcolor{red}{00000000} \text{ } 00000000 \text{ } 00000000 \text{ } 00000001 \\ -) \text{ } 2 = 00000000 \text{ } 00000000 \text{ } 00000000 \text{ } 00000010 \end{array} \rightarrow a \text{ (红色表示整型提升的填充位)}$$

$$\begin{array}{r} 11111111 \text{ } 11111111 \text{ } 11111111 \text{ } 11111111 \\ -) \text{ } b = \textcolor{red}{11111111} \text{ } \textcolor{red}{11111111} \text{ } 11111111 \text{ } 11111111 \end{array} \rightarrow a-2(\text{int型})$$

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

$$11111111 \text{ } 11111110$$

- (2) 取反 00000000 00000001
(3) 绝对值 1 (十进制表示形式)
(4) 加负号 -1 (十进制表示形式)

本页不用作答



§. 基础知识题

1、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=32743;  
short b=a+31;
```

Step1: $b = a - 2$, 得b二进制补码形式

$$\begin{array}{r} a = \textcolor{red}{00000000} \ 00000000 \ 01111111 \ 11100111 \rightarrow a \ (\text{红色表示整型提升的填充位}) \\ +) \ 2 = 00000000 \ 00000000 \ 00000000 \ 00011111 \rightarrow 31 \\ \hline \end{array}$$

$$\begin{array}{r} 00000000 \ 00000000 \ 10000000 \ 00000110 \rightarrow a+31(\text{int型}) \\ b = \textcolor{red}{00000000} \textcolor{red}{-} \textcolor{red}{00000000} \ 10000000 \ 00000110 \rightarrow b=a+31(\text{二进制补码形式, 删去线表示丢弃的位数}) \end{array}$$

Step2: 求b的十进制表示

(1) 减一 $10000000 \ 00000110$
 $-) \ 00000000 \ 00000001$

$$\hline \ 10000000 \ 00000101$$

- (2) 取反 $01111111 \ 11111010$
(3) 绝对值 32762 (十进制表示形式)
(4) 加负号 -32762 (十进制表示形式)



§. 基础知识题

1、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=65418;  
short b=a;
```

Step1: b=a, 得b二进制补码形式

65418 = 00000000 00000000 11111111 10001010
a = ~~00000000 00000000~~ 11111111 10001010 → a (二进制补码形式, 删除线表示丢弃的位数)
b = 11111111 10001010 → b=a(a赋值给b)

Step2: 求b的十进制表示

(1) 减一 11111111 10001010
 -) 00000000 00000001

 11111111 10001001

(2) 取反 00000000 01110110
(3) 绝对值 118 (十进制表示形式)
(4) 加负号 -118 (十进制表示形式)



§. 基础知识题

1、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=-2035;  
    int b=a;
```

Step1: b=a, 得b二进制补码形式

-2035 = 01111111 11111111 11111000 00001101 → -2035的二进制补码形式

a = ~~01111111 11111111~~ 11111000 00001101 → a (二进制补码形式, 删除线表示丢弃的位数)

b = ~~11111111 11111111~~ 11111000 00001101 → b=a (红色表示整型提升的填充位, 补a符号位)

Step2: 求b的十进制表示

(1) 减一 11111111 11111111 11111000 00001101
 -) 00000000 00000000 00000000 00000001

 11111111 11111111 11111000 00001100

(2) 取反 00000000 00000000 00000111 00000011

(3) 绝对值 2035 (十进制表示形式)

(4) 加负号 -2035 (十进制表示形式)



§. 基础知识题

1、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=65418;  
long long int b=a;
```

Step1: b=a, 得b二进制补码形式

65418 = 00000000 00000000 11111111 10001010 → 65418的二进制补码形式

a = ~~00000000 00000000~~ 11111111 10001001 → a (二进制补码形式, 删除线表示丢弃的位数)

b = 00000000 00000000 00000000 00000000 00000000 11111000 00001101 → b=a

(红色表示整型提升的填充位)

Step2: 求b的十进制表示

b为正值, 补码与原码相同, 因此b = 65418



§. 基础知识题

1、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=4202454307; //需将a的最后7位（题中的1234567）替换为你的学号，否则本作业分数为0  
int b=a;
```

Step1: b=a, 得b二进制补码形式

a = 00000000 00000000 00000000 00000000 11111010 01111100 01011101 00100011 → a (二进制补码形式)

b = ~~00000000 00000000 00000000 00000000~~ 11111010 01111100 01011101 00100011 → b=a

(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 11111010 01111100 01011101 00100011
 -) 00000000 00000000 00000000 00000001

 11111010 01111100 01011101 00100010

(2) 取反 00000101 10000011 10100010 11011101

(3) 绝对值 92512989 (十进制表示形式)

(4) 加负号 -92512989 (十进制表示形式)



§. 基础知识题

1、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=-4202454307; //提示1: 需将a的最后7位（题中的1234567）替换为你的学号，否则本作业分数为0  
unsigned short b=a; //提示2: 本题先确定 -4201234567 什么类型，a是多少，才能进行b=a的计算
```

Step1: b=a, 得b二进制补码形式

10000000 00000000 00000000 00000000 11111010 01111100 01011101 00100011

-4202454307 = 01111111 11111111 11111111 11111111 00000101 10000011 10100010 11011101 (二进制
补码形式)

a = ~~01111111 11111111 11111111 11111111~~ 00000101 10000011 10100010 11011101 → a (二进制
补码形式, 删除线表示丢弃的位数)

b = ~~00000101 10000011~~ 10100010 11011101 → b=a (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) b = $(10100010\ 11011101)_2 = 41693$



§. 基础知识题

2、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

例. $1 + 2 + 3$

表达式一共有2个运算符，因此计算的2个步骤分别是（仿课件P. 85，本页不需要画栈，但要有栈思维，下同）：

步骤①： $1 + 2 \Rightarrow$ 式1

步骤②：式1 + 3

本页不用作答



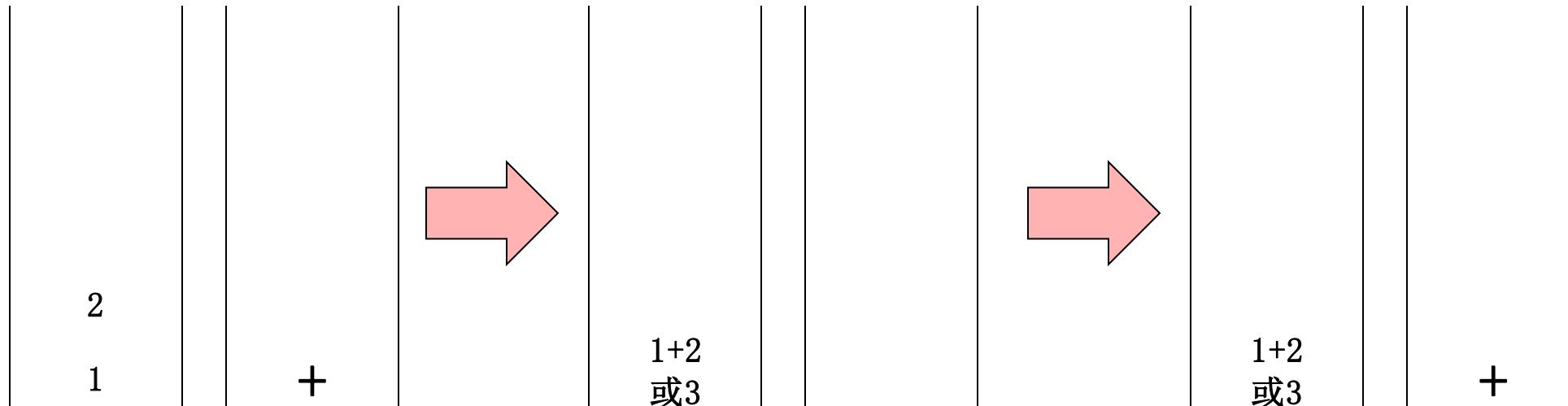
§. 基础知识题

2、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

例. $1 + 2 + 3$



目前准备进栈的运算符如箭头所示，画出从该运算符准备进栈到进栈完成的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，提示：本例是3组，等价于课件P. 69~71）



要进栈的+等于栈顶的+，
左结合，先计算

要进栈的+等于栈顶的+，
左结合，先计算

本页不用作答



§. 基础知识题

2、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

$$A. 21 / 2 + 74 \% 7 - 2.3 + 3.5 * 27$$

表达式一共有6个运算符，因此计算的6个步骤分别是（仿课件P. 85，本页不需要画栈，但要有栈思维，下同）：

步骤①： $21 / 2$

步骤②： $74 \% 7$

步骤③： $① + ②$

步骤④： $③ - 2.3$

步骤⑤： $3.5 * 2.7$

步骤⑥： $④ + ⑤$



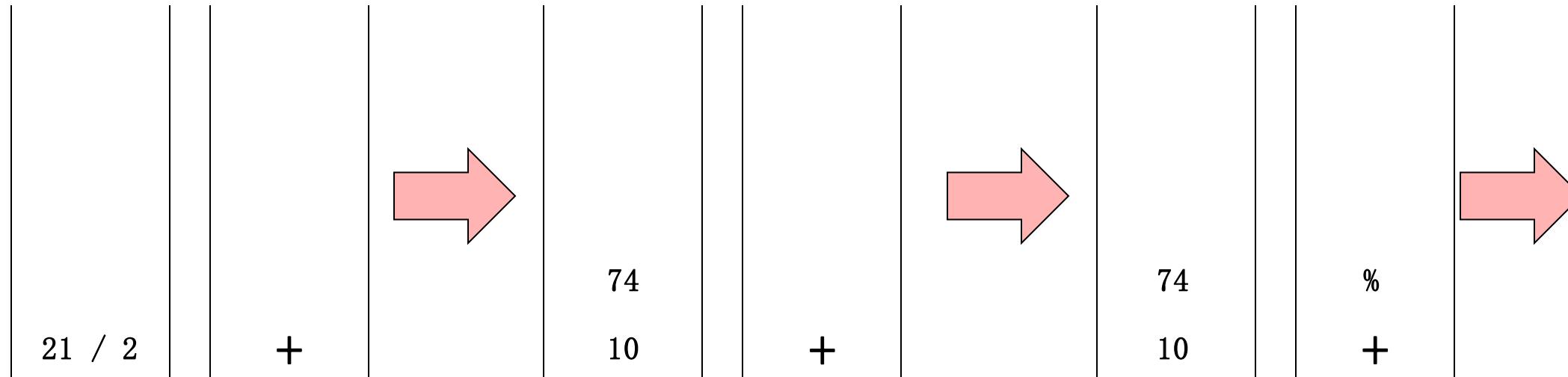
§. 基础知识题

2、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

$$A. 21 / 2 + 74 \% 7 - 2.3 + 3.5 * 27$$



目前准备进栈的运算符如箭头所示，画出从该运算符**准备进栈到进栈完成**的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





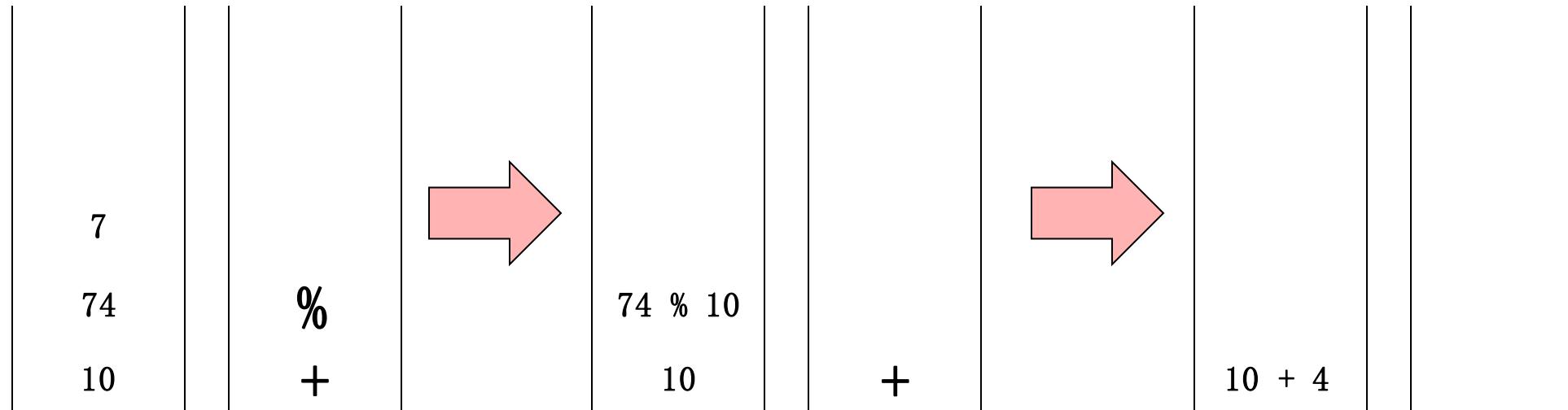
§. 基础知识题

2、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

$$A. 21 / 2 + 74 \% 7 - 2.3 + 3.5 * 27$$



目前准备进栈的运算符如箭头所示，画出从该运算符**准备进栈到进栈完成**的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





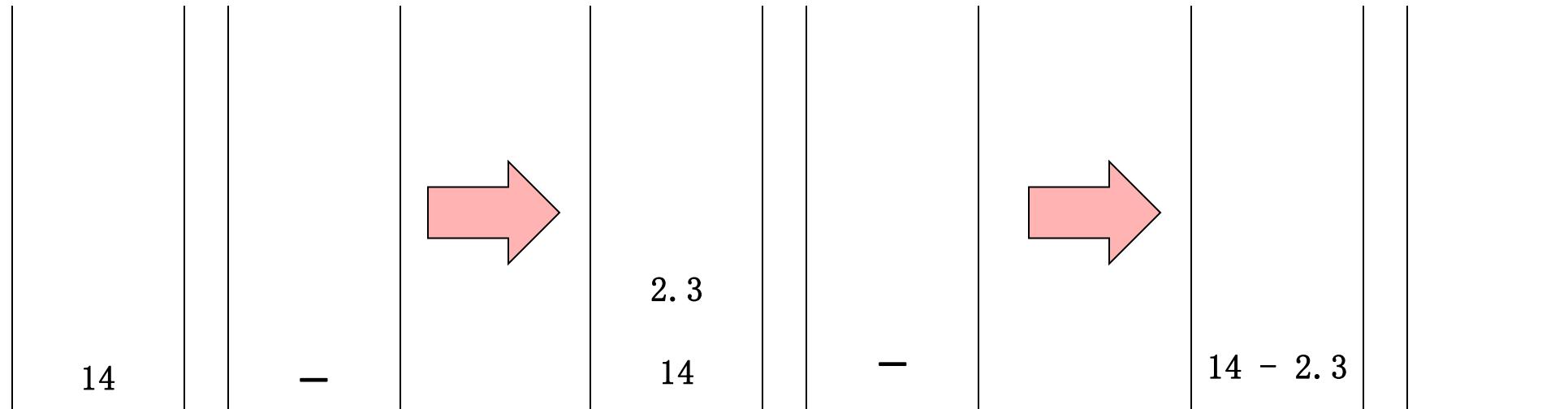
§. 基础知识题

2、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

$$A. 21 / 2 + 74 \% 7 - 2.3 + 3.5 * 27$$



目前准备进栈的运算符如箭头所示，画出从该运算符**准备进栈到进栈完成**的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





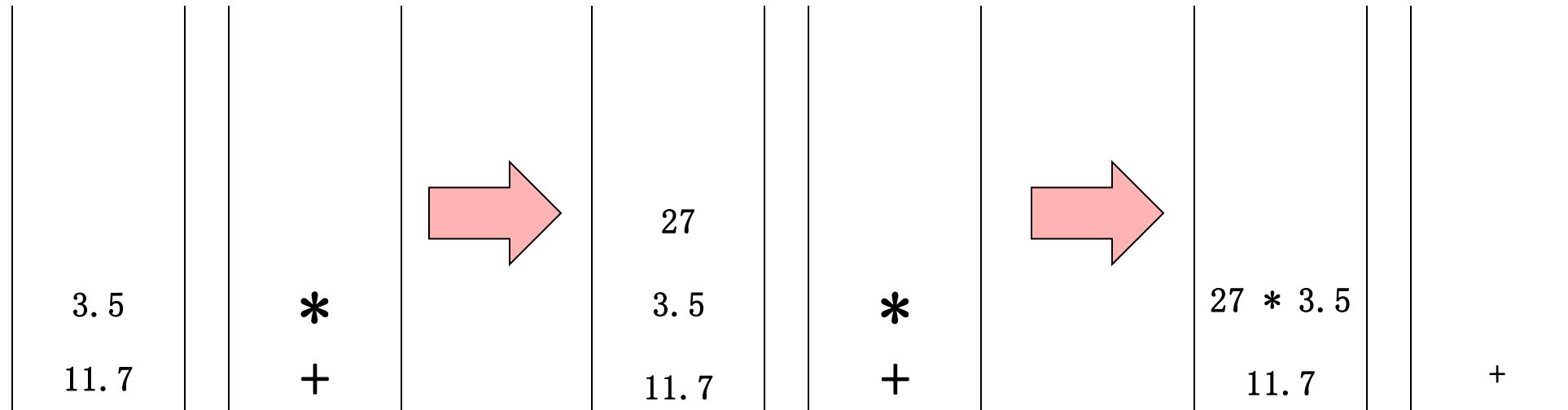
§. 基础知识题

2、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

$$A. 21 / 2 + 74 \% 7 - 2.3 + 3.5 * 27$$



目前准备进栈的运算符如箭头所示，画出从该运算符**准备进栈到进栈完成**的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





§. 基础知识题

2、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. $x = 4 * 3, \quad x = y = 5 * 3$ （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： $4 * 3$

步骤②： $x = ①$

步骤③： $5 * 3$

步骤④： $y = ③$

步骤⑤： $x = y$

步骤⑥： $x = 12, \quad x = y = 15$



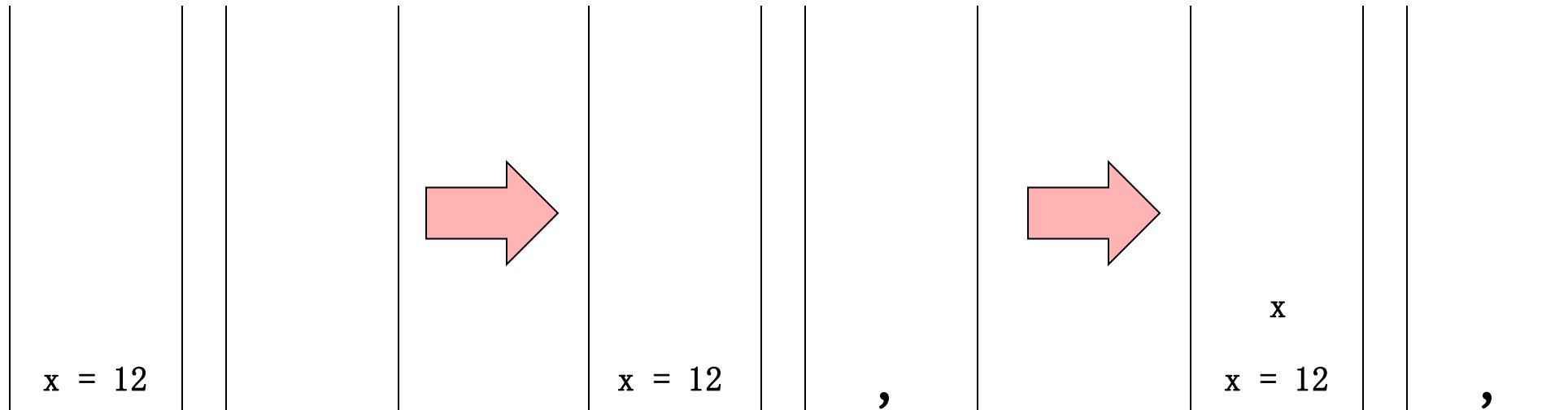
§. 基础知识题

2、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. $x = 4 * 3, x = y = 5 * 3$ (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，画出从该运算符**准备进栈到进栈完成**的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





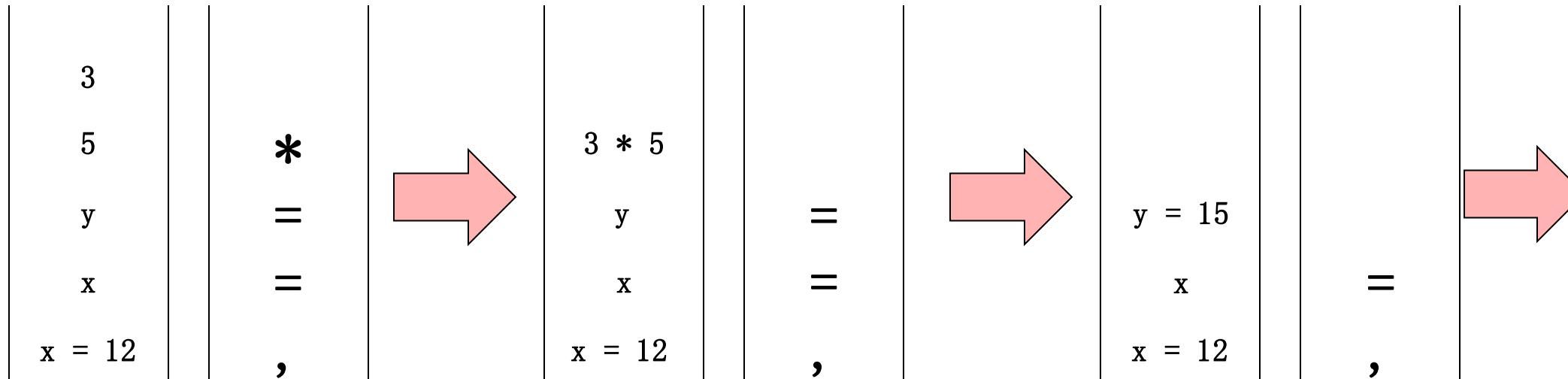
§. 基础知识题

2、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. $x = 4 * 3, x = y = 5 * 3$ (假设所有变量均为int型)



目前已分析到整个表达式的尾部，画出到表达式求值完成的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





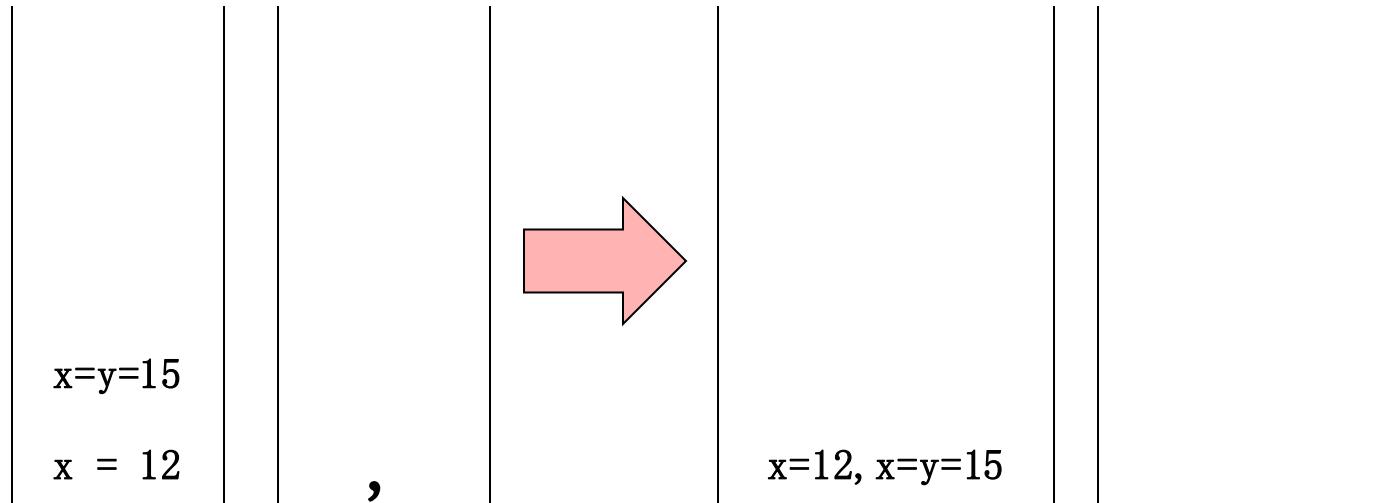
§. 基础知识题

2、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. $x = 4 * 3, x = y = 5 * 3$ (假设所有变量均为int型)



目前已分析到整个表达式的尾部，画出到表达式求值完成的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





§. 基础知识题

2、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. $a + (a - 3 / (b + c) + 5) \% 4$ (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（左右括号不算步骤）：

步骤①： $b + c$

步骤②： $3 / \textcircled{1}$

步骤③： $a - \textcircled{2}$

步骤④： $\textcircled{3} + 5$

步骤⑤： $\textcircled{4} \% 4$

步骤⑥： $a + \textcircled{5}$

后面自行添加，主要是对()的理解，本页中一对括号可以当做一个步骤理解，后续画栈时要分开



§. 基础知识题

2、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

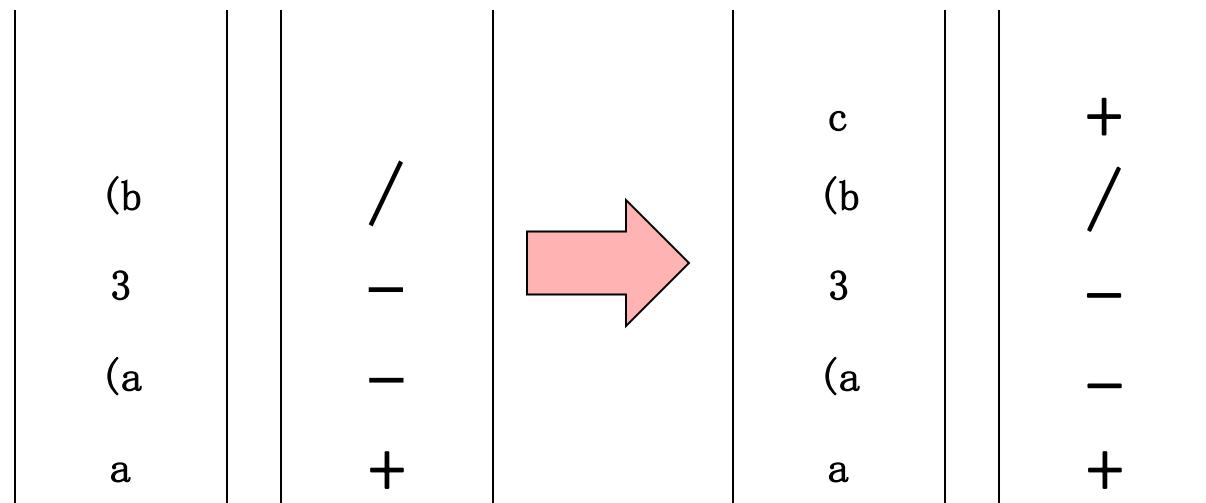
C. $a + (a - 3 / (b + c) + 5) \% 4$ (假设所有变量均为int型)

(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，画出从该运算符准备进栈到进栈完成的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





§. 基础知识题

2、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



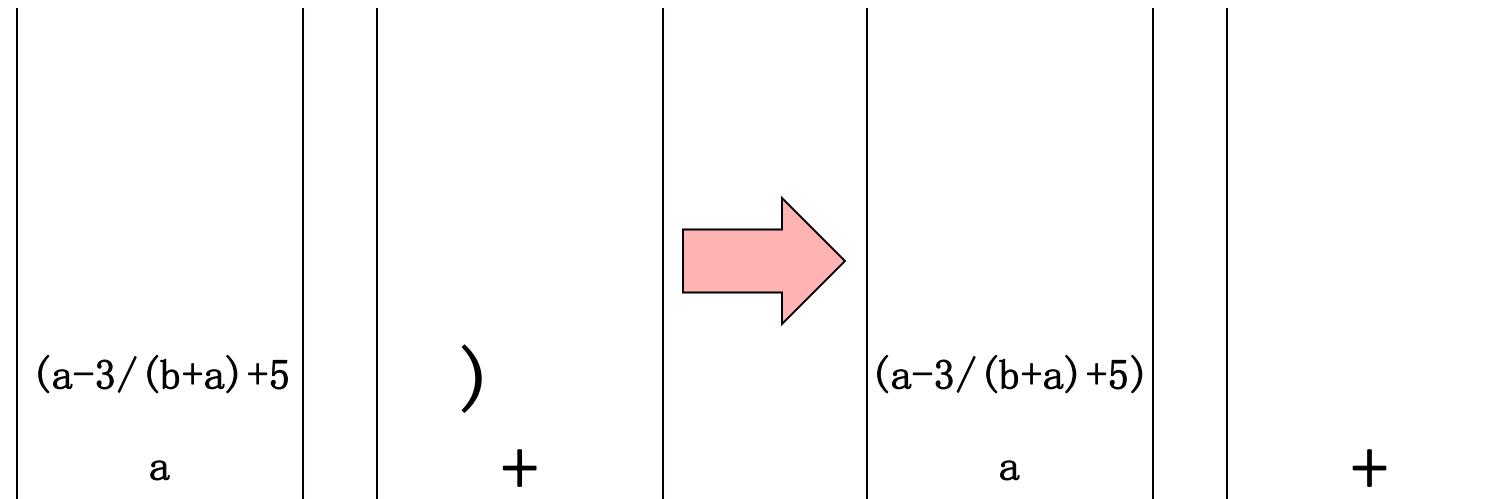
C. $a + (a - 3 / (b + c) + 5) \% 4$ (假设所有变量均为int型)

(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，画出从该运算符准备进栈到进栈完成的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





§. 基础知识题

2、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

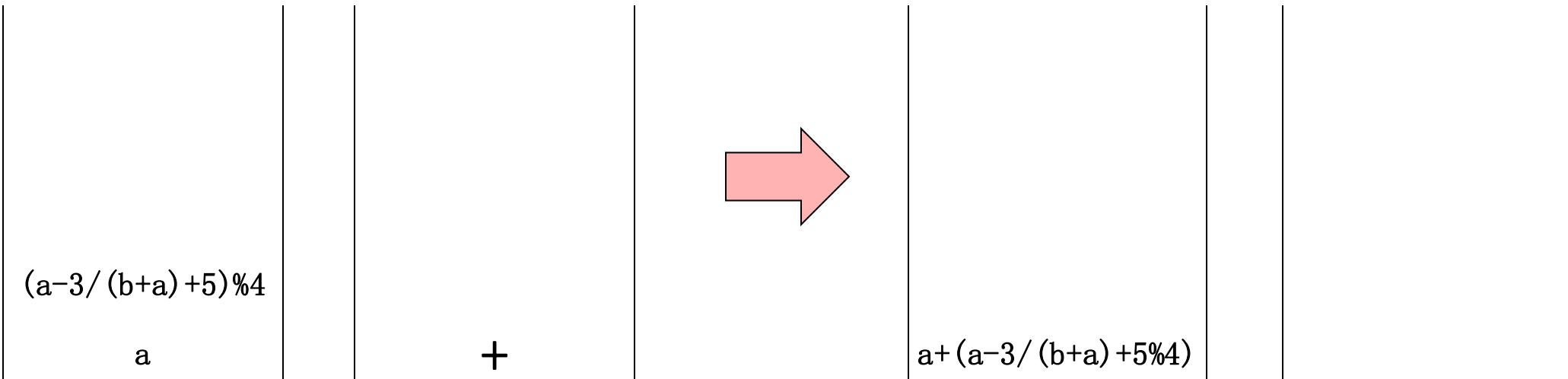
C. $a + (a - 3 / (b + c) + 5) \% 4$ (假设所有变量均为int型)

(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

目前已分析到整个表达式的尾部，画出到表达式求值完成的过程中，当前运算数栈和运算符栈的状态（本页需要画栈，两个栈算一组，最少两组，可能多组，每页三组，不够可加页）





§. 基础知识题

3、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： $2\text{LL} - 32\text{L} * \text{int}(11.7) + 2.3\text{f}$

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

注：回答问题时，4步一定要标清楚，
验证程序不强制，可以验证多步
或最后一步

The screenshot shows a Microsoft Visual Studio interface. The code editor window displays a file named "demo.cpp" with the following content:

```
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
```

To the right of the code editor is the "Immediate Window" titled "Microsoft Visual Studio 调试控制台". It contains the following output:

```
-347.7
float
```

A green vertical bar highlights the line of code being executed: `cout << 2LL - 32L * int(11.7) + 2.3f << endl;`.

本页不用作答



§. 基础知识题

3、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. $x = 4 * 3, x = y = 5 * 3$ （写验证程序时，假设所有变量均为int型）

- (1) $4 * 3 \Rightarrow 12$ int型
- (2) $x = 12 \Rightarrow x$ int型
- (3) $5 * 3 \Rightarrow 15$ int型
- (4) $y = 15 \Rightarrow y$ int型
- (5) $x = y(15) \Rightarrow x$ int型

```
1 # include <iostream>
2 using namespace std;
3
4 int main() {
5     int x, y;
6     x = 4 * 3, x = y = 5 * 3;
7     cout << "x = " << x << " y = " << y << endl;
8     cout << typeid(x).name() << typeid(y).name() << endl;
9     return 0;
10 }
```

Microsoft Visual Studio Debug Console
x = 15 y = 15
intint



§. 基础知识题

3、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. $a + (b - 3 / (b + c) \% 4) * 5$ (写验证程序时，假设所有变量均为int型，abc的值自定义即可)

(1) $b + c$	$\Rightarrow b + c$	int型
(2) $3 / (1)$	$\Rightarrow 3 / (b + c)$	int型
(3) $(2) \% 4$	$\Rightarrow 3 / (b + c) \% 4$	int型
(4) $b - (3)$	$\Rightarrow b - 3 / (b + c) \% 4$	int型
(5) $(4) * 5$	$\Rightarrow (b - 3 / (b + c) \% 4) * 5$	int型
(6) $a + (5)$	$\Rightarrow a + (b - 3 / (b + c) \% 4) * 5$	int型

$3 + (2 - 3 / (2 + 1) \% 4) * 5 = 8$, 与输出结果一致

```
1 # include <iostream>
2 using namespace std;
3
4 int main() {
5     int a = 3, b = 2, c = 1;
6     cout << a + (b - 3 / (b + c) \% 4) * 5 << endl;
7     cout << typeid(a + (b - 3 / (b + c) \% 4) * 5).name() << endl;
8     return 0;
9 }
```

Microsoft Visual Studio Debug Console

8
int



§. 基础知识题

3、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. $2.5F * 3LU + 4ULL * 7U - 'W'$

- (1) $2.5F * 3LU \Rightarrow 7.5F$ float型
- (2) $4ULL * 7U \Rightarrow 28ULL$ unsigned long long型
- (3) $7.5F + 28ULL \Rightarrow 35.5F$ float型
- (4) $35.5F - (\text{char}) 'W'$ $\Rightarrow -51.5F$ float型

```
1 # include <iostream>
2 using namespace std;
3
4 int main() {
5     cout << 2.5F * 3LU + 4ULL * 7U - 'W' << endl;
6     cout << typeid(2.5F * 3LU + 4ULL * 7U - 'W').name() << endl;
7 }
8 // Microsoft Visual Studio Debug Console
// -51.5
// float
```



§ . 基础知识题

3、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. $2\text{UL} \% 7 + 23 \% 3\text{LL} - 2.5\text{F}$

- (1) $2\text{UL} \% 7 \Rightarrow 2\text{UL}$ unsigned long型
- (2) $23 \% 3\text{LL} \Rightarrow 2\text{LL}$ long long型
- (3) $2\text{UL} + 2\text{LL} \Rightarrow 4\text{LL}$ long long型
- (4) $4\text{LL} - 2.5\text{F} \Rightarrow 1.5\text{F}$ float型

```
1 # include <iostream>
2 using namespace std;
3
4 int main() {
5     cout << 2UL % 7 + 23 % 3LL - 2.5F << endl;
6     cout << typeid(2UL % 7 + 23 % 3LL - 2.5F).name() << endl;
7 }
// Microsoft Visual Studio Debug Console
8
1.5
float
```



§. 基础知识题

3、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. $3.2 + 13 \% 7 * \text{static_cast}<\text{unsigned long}>(2.8F * 7LL) \% 2 * 3.2F$

- (1) $13 \% 7 \Rightarrow 6$ int型
- (2) $2.8F * 7LL \Rightarrow 19.6F$ float型
- (3) $\text{static_cast}<\text{unsigned long}>(10.6F) \Rightarrow 19UL$ unsigned long型
- (4) $6 * 19UL \Rightarrow 114UL$ unsigned long型
- (4) $114UL \% 2 \Rightarrow 0UL$ unsigned long型
- (4) $0UL * 3.2F \Rightarrow 0F$ float型
- (4) $3.2 + 0F \Rightarrow 3.2$ double型

The screenshot shows a Microsoft Visual Studio interface. The code editor window displays a file named "HelloWorld" with the following content:

```
# include <iostream>
using namespace std;

int main() {
    cout << 3.2 + 13 % 7 * static_cast<unsigned long>(2.8F * 7LL) % 2 * 3.2F << endl;
    cout << typeid(3.2 + 13 % 7 * static_cast<unsigned long>(2.8F * 7LL) % 2 * 3.2F).name() << endl;
}
```

The Microsoft Visual Studio Debug Console window below shows the output:

```
3.2
double
```



§. 基础知识题

3、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. $\text{long}(2.8F + 3.3) * 2 + (\text{int})1.9 \% 7U - 'p' * 2UL$

- | | | |
|------------------------|-----------------|----------------|
| (1) $2.8F + 3.3$ | => 6.1 | double型 |
| (2) $\text{long}(6.1)$ | => 6L | long型 |
| (2) $6L * 2$ | => 12L | long型 |
| (3) $(\text{int})1.9$ | => 1 | int型 |
| (4) $1 \% 7U$ | => 1U | unsigned int型 |
| (4) $12L + 1U$ | => 13L | long型 |
| (4) $'p' * 2UL$ | => 224UL | unsigned long型 |
| (4) $13L - 224UL$ | => 4294967085UL | unsigned long型 |

```
1 # include <iostream>
2 using namespace std;
3
4 int main() {
5     cout << long(2.8F + 3.3) * 2 + (int)1.9 % 7U - 'p' * 2UL << endl;
6     cout << typeid(long(2.8F + 3.3) * 2 + (int)1.9 % 7U - 'p' * 2UL).name() << endl;
7 }
8
```

Microsoft Visual Studio Debug Console

4294967085
unsigned long



§. 基础知识题

4、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

=> 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

The screenshot shows the Microsoft Visual Studio IDE. On the left, the code editor displays a file named 'demo.cpp' with the following content:

```
1 // demo.cpp
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 }
```

To the right of the code editor is the 'Immediate Window' titled '(全局范围)'. It contains the output of the program: '17 12'.

本页不用作答



§. 基础知识题

4、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 = 3, n = 7;`

A. `a += a - n`

=> `a = a + (a - n)`

(1) `a - n` `a=3 n=7 差为-4`

(2) `a + 差` `a=3 n=7 结果-1存放在中间变量中`

(3) `a = 结果.` `a=-1 n=7`

```
1 # include <iostream>
2 using namespace std;
3
4 int main() {
5     int a = 3, n = 7;
6     a += a - n;
7     cout << a << ' ' << n;
8 }
9 // Microsoft Visual Studio Debug Console
// -1 7
```



§. 基础知识题

4、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 = 3, n = 7;`

B. `n -= a += 5`

=> `a = a + 5, n = n - a`

(1) `a + 5` `a=3 n=7` 结果8存放在中间变量中

(2) `a = (1)结果` `a=8 n=7`

(3) `n - a` `a=8 n=7` 差-1存放在中间变量中

(4) `n = 差` `a=8 n=-1`

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

The screenshot shows a Microsoft Visual Studio code editor with a dark theme. Line 6, which contains the expression `n -= a += 5;`, is highlighted with a gray rectangular selection. Below the code editor is a terminal window titled "Microsoft Visual Studio Debug Console". The console displays the output of the program: `8 -1`. A mouse cursor is visible over the terminal window.



§. 基础知识题

4、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 = 3, n = 7;`

C. `a += a *= a -= a`

=> `a = a - a, a = a * a, a = a + a`

(1) `a - a` `a=3 n=7` 结果0存放在中间变量中

(2) `a = (1)结果` `a=0 n=7`

(3) `a * a` `a=0 n=7` 结果0存放在中间变量中

(4) `a = (3)结果` `a=0 n=7`

(5) `a + a` `a=0 n=7` 结果0存放在中间变量中

(6) `a = (5)结果` `a=0 n=7`

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

Microsoft Visual Studio Debug Console
0 7



§. 基础知识题

4、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 = 8, n = 13;

D. $n \%= a \%= 4$ 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)

答：a对4取模后结果为0，0赋值给a使得a=0,而接下来的运算是n%a，是对0取模，这样的运算无意义，是一种错误，但是编译器并没有将这种错误定义，所以运行时不会报错。但是这样的运算会使得运行时产生不正确的信息，这些信息无法产生正常的输出结果。同时系统在处理这些错误信息时需要更多时间，所以运行时间更长。而推出代码代表windows操作系统对错误类型的判断，正数除以零的错误代码正是0xc0000094.

The screenshot shows a Microsoft Visual Studio IDE window titled "demo.cpp". The code in the editor is:

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     int a = 8, n = 13;
7     cout << (n %= a %= 4) << endl;
8     return 0;
9 }
```

The output window at the bottom is titled "Microsoft Visual Studio 调试控制台" and displays the following message:

D:\WorkSpace\VS2022-demo\Debug\demo-cpp.exe (进程 34000)已退出, 代码为 -1073741676 (0xc0000094)。
按任意键关闭此窗口. . .

A red arrow points from the text "代码为 -1073741676 (0xc0000094)。" to the error code value "-1073741676 (0xc0000094)" in the output window.