

Exercise 1.6.1: For the block-structured C code of Fig. 1.13(a), indicate the values assigned to w , x , y , and z .

Exercise 1.6.2: Repeat Exercise 1.6.1 for the code of Fig. 1.13(b).

CODE	SCOPE EVALUATION
<pre> int w, x, y, z; int i = 4; int j = 5; { int j = 7; i = 6; w = i + j; } x = i + j; { int i = 8; y = i + j; } z = i + j; </pre> <p>(a) Code for Exercise 1.6.1</p>	<pre> w:13 // 6+7 x:11 // 6+5 y:13 // 8+5 z:11 // 6+5 </pre>
<pre> int w, x, y, z; int i = 3; int j = 4; { int i = 5; w = i + j; } x = i + j; { int j = 6; i = 7; y = i + j; } z = i + j; </pre> <p>(b) Code for Exercise 1.6.2</p>	<pre> w:9 // 5+4 x:7 // 3+4 y:13 // 7+6 z:11 // 7+4 </pre>

Exercise 1.6.3: For the block-structured code of Fig. 1.14, assuming the usual static scoping of declarations, give the scope for each of the twelve declarations.

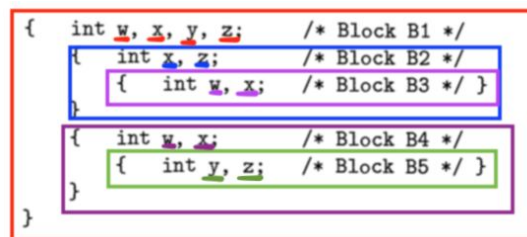


Figure 1.14: Block structured code for Exercise 1.6.3

Each color indicates the scope of the variables

Exercise 1.6.4: What is printed by the following C code?

```

#define a (x+1)
int x = 2;
void b() { x = a; printf("%d\n", x); }
void c() { int x = 1; printf("%d\n"), a; }
void main() { b(); c(); }
                
```

Output:

3
2