

## 第一章作业

### 1.1.2 编译器相对于解释器的优点是什么？解释器相对于编译器的优点是什么？

相对于解释器，编译器产生机器语言目标程序的速度通常更快；

相对于编译器，解释器的错误诊断效果更好，因为它逐个语句地执行源程序。

### 1.1.3：在一个语言处理系统中，编译器产生汇编语言而不是机器语言的好处是什么？

编译器可能会产生汇编语言作为其输出，因为汇编语言比较容易输出和调试。

### 1.6.1：对下图中的块结构的 C 代码，指出赋给 w、x、y 和 z 的值

#### 1.6.2：对前一页图 b 中的代码重复练习 1.6.1

```
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;
```

(a) Code for Exercise 1.6.1

```
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;
```

(b) Code for Exercise 1.6.2

$$1.6.1 \quad w = 6 + 7 = 13$$

$$x = 6 + 5 = 11$$

$$y = 8 + 5 = 13$$

$$z = 6 + 5 = 11$$

$$1.6.2 \quad w = 5 + 4 = 9$$

$$x = 3 + 4 = 7$$

$$y = 7 + 6 = 13$$

$$z = 7 + 4 = 11$$

### 1.6.4：下面的 C 代码的打印结果是什么？

```
#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(); }
```

首先调用函数 b， $x = a$ ，a 的定义为  $(x+1)$ ，而 x 的赋值只有全局变量中的声明  $x = 2$ ，故  $x = x + 1 = 3$ ，打印 3；再调用函数 c，函数 c 中的变量 x 的声明为  $x = 1$ ，故  $a = x + 1 = 2$ ，打印 2。

故打印结果为：

3

2