

编译原理 作业 1
李昊宸 2017K8009929044

练习 1.1.2

解释器是解释执行的源代码，而编译器是将源代码编译成目标代码。

编译器相对于解释器的优点是什么？

编译器提前将源代码转换成低级语言的代码，在接收输入后可迅速的直接计算出输出，而解释器需要将源代码和输入同时运算，故编译器的执行速度远远快于解释器。

解释器相对于编译器的优点是什么？

解释器的一个优点是跨平台方便，因为它无需像编译器一样需要编译出符合不同机器的低级语言代码。另外解释器通常能比编译器给出更好的错误诊断，因为它可以顺序显式执行执行源程序。

练习 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？

汇编语言首先可以较为简单的由机器进一步转换为机器语言（二进制代码），并且汇编语言方便程序开发者进行错误诊断和逻辑判断，更容易产生输出。

练习 1.6.1

给出赋给 W、X、Y、Z 的值

```
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;
```

w = 13 因 i = 6, j = 7;
x = 11 因 i = 6, j = 5;
y = 13 因 i = 8, j = 5;
z = 11 因 i = 6, j = 5。

练习 1.6.2

```
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;
```

w = 9 因 i = 5, j = 4;
x = 7 因 i = 3, j = 4;
y = 13 因 i = 7, j = 6;
z = 11 因 i = 7, j = 4。

练习 1.6.4

```
#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(); }
```

打印结果: 3
 2