



# 第一章 作业

截止日期: 2025.03.13

- 练习1.1.1: 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?
- 练习1.1.2: 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?
- 练习1.1.3: 对下图中的块结构的C代码, 指出每次赋给w、x、y和z的值

(1)

```
int w, x, y, z;
int i = 5, j = 13;
{
    int j = 17;
    i = j - 10;
    w = i + j;
}
x = i * j;
{
    int i = j;
    y = i * j;
}
z = i + j;
```

(2)

```
int w, x, y, z;
int i = 9, j = 7;
{
    int i = j + 1;
    w = i * 2;
}
x = i - j;
{
    int j = 17;
    i = 11;
    y = i + j;
}
z = i + j;
```



# 第一章 作业

截止日期: 2025.03.13

- 练习1.1.4: 下面的C代码的打印结果是什么?

```
#include <stdio.h>
#define a x
int x = 4;
void b(){x = a * 3; printf("%d\n",x);}
void c(){int x = 1111; printf("%d\n", a + 1);}
int main(){ b(); c();}
```

- 练习1.1.5\*: 有人把程序设计语言分为编译型和解释型两类, 例如C是编译型, Python是解释型。这个分类是否合理? 能否构建C语言的解释器, 或者Python的静态编译器? 谈谈你的看法。