

## Lec03 作业

### 1、给出指针类型的运算说明。生存期如何？（V，OP）

解：值（V）：指针是一个变量，值为另一个变量的地址。

运算说明（OP）：

1) 指针的算术运算：

a)  $+/ -$ ：该运算会使指针访问另一内存地址，例如：

```
char *a = "abcd" ;    //*a = 'a'
a += 2;                //*a = 'c'
a -= 1;                //*a = 'b'
```

b)  $++/--$ ：该运算会使指针访问下一个/上一个内存地址

2) 指针的取地址运算：

a)  $\&$ ： $\&$ 是一元运算符，能够返回操作数的内存地址。例如：

```
char a = 'a' ;
char *p = &a;          //p 指向 a 的地址
```

3) 指针的间接寻址运算：

a)  $*$ ： $*$ 是一元运算符，能够返回操作数所指向地址的变量的值。例如：

```
char b = *a;           //b 的值为 a 地址里的值
```

4) 指针的比较运算：

a)  $>/</==$ ：一般在指针指向两个相关变量时使用，比如可以用来比较同一数组元素大小。

生存期：指针实现的记录或结构的树或链其第一头指针在堆中分配，生存期比堆栈帧内束定的变量长。

### 2、静态作用域有无闭包机制，比如 basic 语言。为什么？

解：静态作用域有闭包机制。

闭包包含自由（未绑定到特定对象）变量，这些变量不是在这个代码块内或者任何全局上下文中定义的，而是在定义代码块的环境中定义（局部变量）。闭包的访问过程是内部→外部。

静态作用域查找的是距离当前作用域最近的外层作用域中同名标识符的声明。

### 3、C 语言中，在函数外定义一个 static 变量和在函数内定义一个 static 变量的区别是什么？

解：静态变量可以当作全局变量，但只能在其声明的作用域中使用。只有第一次运行函数的时候被创建，并且在函数结束的时候不会消亡。

在函数外定义静态变量是文件作用域的静态变量，有文件作用域的静态变量只能在包含它的.c 文件中使用，如果是声明在头文件中的静态变量，那么在不同的.c 文件包含它的时候，会各自生成一个独立的副本。

在函数内声明的静态变量其实就是一个全局变量，只不过它只能在函数内部用。

### 4、考虑下面的 C 程序：

```
int fun(int *i){ *i += 5; return 4;}
void main(){ int x = 3; x=x+fun(&x);}
```

在 main 中的赋值语句之后，x 的值是什么，假设

- a. 操作数是以从左到右的顺序来求值的。
- b. 操作数是以从右到左的顺序来求值的。

解：a. 从左到右：7

b. 从右到左：12

5、使用 c 语言中的多向选择语句重新编写下面的代码段：

```
if ((k==1) || (k==2)) j= 2*k-1;
if ((k==3) || (k==5)) j= 3*k+1;
if (k==4) j= 4*k-1;
if ((k==6) || (k==7) || (k==8)) j= k-2;
```

解：

```
switch (i){
    case 1:
    case 2:
        j= 2 * k - 1;
        break;
    case 3:
    case 5:
        j= 3 * k + 1;
        break;
    case 4:
        j= 4 * k - 1;
        break;
    case 6:
    case 7:
    case 8:
        j= k - 2;
        break;
}
```

6、思考题：使用回调写排序有什么好处。

解：可以实现不同类型数据的排序，并且可以自定义排序规则，使得比较更加灵活。