

## 6.2 指针的本质

2025年6月27日 10:36

指针是一种变量类型。

指针不是地址。

指针类型变量存储的数据是地址。

### 个人理解

如果用现实世界去描述的话，那么可以将计算机的内存空间看成一排抽屉，指针保存的就是这些抽屉的编号；例如: `int *index`; 则指针解引用就是打开这个抽屉，读取指针指向地址的内容就是看看对应抽屉里的东西，如果是 `*index = 5`，这种写操作，则是将一个5放进这个抽屉（抽屉中原有的内容则被替换掉）。

#### 1、指针变量定义方式

`int* a`和`int *a`是等价的，为什么大家都用`int *a`呢？

因为类似`int* a,b,c`的语句会使人很自然地认为这条语句把所有三个变量声明为整型指针，但事实上并非如此，“\*”实际上是\*a的一部分，只对a标识符起作用，其余两个变量只是普通的整型变量。

要声明三个指针变量，正确的语句如下：

```
int *a,*b,*c;
```

#### 2、这种写法 `float a;`

```
int *p;
```

```
p=&a; //毫无意义而且会出错
```

3、“&”和“\*”两个运算符的优先级别相同，但要按自右向左的方向结合，因此`*&a`与`a`等价，我们不会将取值和取地址放在一块这么去使用。

### 习题

1、`printf("%d",i);`对于变量`i`是直接访问

A 正确 B 错误

2、`int i = 5; int* p=&i; printf("p=%d\n", *p);`这样去访问`i`的值的方式是间接访问

A 正确 B 错误

3、“&”和“\*”两个运算符的优先级别相同，但要按自右向左的方向结合，&是取值，\*是取地址

A 正确 B 错误

4、`int *a,b,c`定义了3个整型指针变量，分别是`a,b,c`

4、int \*a,b,c 定义了 3 个整型指针变量，分别是 a,b,c

A 正确 B 错误

5、int a;如果我们编写\*&a 这样的表达式没有意义，因为其与 a 等价的

A 正确 B 错误

6、int i=10;通过&i 拿到的是变量 i 的起始地址，而不是变量 i 的整个空间

A 正确 B 错误

7、int i=10;int \*p1=&i;char c='a';char \*p2=&c;我们通过\*p1 拿到的是 4 个字节的空间，通过\*p2 拿到的是 1 个字节大小的空间，对一个指针变量取值运算，拿到的空间大小，依赖的是指针变量的基类型

A 正确 B 错误

8、对于 32 位操作系统，int i=10;int \*p1=&i;char c='a';char \*p2=&c;我们 sizeof(p1)是 4 个字节，sizeof(p2)是 1 个字节

A 正确 B 错误