

【C 语言基础笔记 3--- 抽象的指针】

作者: [13097917715](#)

原文链接: <https://ld246.com/article/1581500614489>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

1. 什么是指针

指针：也是一种变量，是一种特殊的变量(专门用来存放其它变量在内存中地址的)

如果一个变量有多个字节(内存中有多个地址)，指针存放的是这个变量的首地址(最低的地址，起始位的地址)

2. 指针的定义

类型 *指针的名字

```
int p; //定义了int类型的指针 *表示p是个指针
```

```
float *q;//定义了float类型的指针 *表示p是个指针
```

```
char p;//定义了char类型的指针 *表示p是个指针
```

3. 指针的简单使用

第一种： 定义一个指针并初始化

```
int a=10;
```

```
int *p=&a; // & -->取地址符号(获取变量a在内存中的首地址)
```

第二种: 定义一个指针没有立马初始化

```
int a=10;
```

```
int *p; //定义了指针p, p存放谁的地址目前是不清楚的
```

```
p=&a;
```

4. 指针的几种基本运算

第一种： 解引用

通过指针访问指向的变量里面的内容，或者通过指针修改指向的变量

使用解引用运算

```
*p; //把指针p解引用(取出指针p指向的地址里面的内容)
```

第二种： 取地址

```
int a=100;
```

```
int *p=&a; //对a取地址
```

小结：指针是C语言提供了一种间接访问变量的方法

5. 举例

```
int a = 100;
int *p = &a;
printf("a = %d\n", *p);    //基本算法--解引用 *p
printf("a的地址%p\n", &a); //查看a在内存中的地址
printf("P中存放的地址%p\n", p); //查看*p中存的地址编码
```

简单来说就是有*号的时候，我们访问的是p指针指向的变量，访问a的值。没有*的时候，我们访问的是p指针指向的变量存在内存的位置，就相当于变量a的门牌号。