

1.C语言中打印地址用%p

2.取地址是取得首字节的地址 因为一个类型的地址是连续的 所以后面的地址也能得到

& + 变量名 如 &a; // 取a的地址

3.指针变量的大小与指向类型无关

一般32位机器 指针是4个字节 范围 0x00000001 - 0xFFFFFFFF

64位机器 指针是8个字节

4.地址不占用额外的空间

5.间接引用 * * + 地址

```
1 int a = 1;
2 *&a == a; // *&a的值等于a
3 *&a = 100; // 对a的值进行改变 改为100
```

6.指针的本质：存地址的变量

7.

(1)如何声明一个指针变量？

所指向的类型 *变量名进行声明

```
1 int a = 10; //a的值为10
2 int *p = &a; // p中存的是a的地址
```

(2).如何判断一个变量的类型？

去掉变量名，剩下的就是类型

```
1 int *p = &a;
```

p的类型是 int * 类型

(3).如何判断一个指针变量所指向的类型？

去掉一个 * 和变量名后，就是指针所指向的类型

```
1 int *p = &a;
```

指针p的指向类型是指向int类型的

(4) 指针的偏移

指针的偏移不取决于右边，取决于左边所指向的类型

```
1 int *p; // 指针一次偏移四个字节 因为int类型占四个字节
2 char *p; // 指针一次偏移一个字节
```

8.

```
1 int *a, b; // a是个指针变量 b是个int类型的变量
```

永远不要操作一个不属于你的内存，如果一个指针不知道指向哪里，就让它指向NULL

9.二级指针：

```
1 int *p = &a; // p中存的是a的地址
2 int **pp = &p; // pp中存的是指针p的地址
```