

# 3.1 C 语言中的指针基础\_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 3.1 C 语言中的指针基础涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

## 1.C 语言中的指针基础

write  
data ----->ram  
  
read  
ram -----> data

- (1) 通过操作变量名来实现
- ```
int a;  
a = 100;  
printf("a = %d\n",a);
```
- (2) 通过内存地址来进行读写操作

```
int a;  
内存地址的获得方法: &a  
  
规则:  
* + 地址: 访问地址中的内存  
  
*(&a) = 100 ;  
printf("*(&a) = %d\n",*(&a));
```

### 示例代码:

```
#include <stdio.h>  
  
int main()  
{  
    int a = 80;  
  
    *(&a) = 66;  
  
    printf("*(&a) = %d\n",*(&a));  
  
    printf("a = %d\n",a);  
    return 0;  
}
```

### 运行结果:

| 常量      | 变量的类型 |
|---------|-------|
| 10      | int   |
| 3.15    | float |
| 'A'     | char  |
| 0xdff88 | ?     |
| (内存地址)  |       |

思考: 内存地址该用什么样的类型来保存呢?  
答: 为了解决这样的问题, C语言的设计者创建了指针类型, 来保存内存地址。

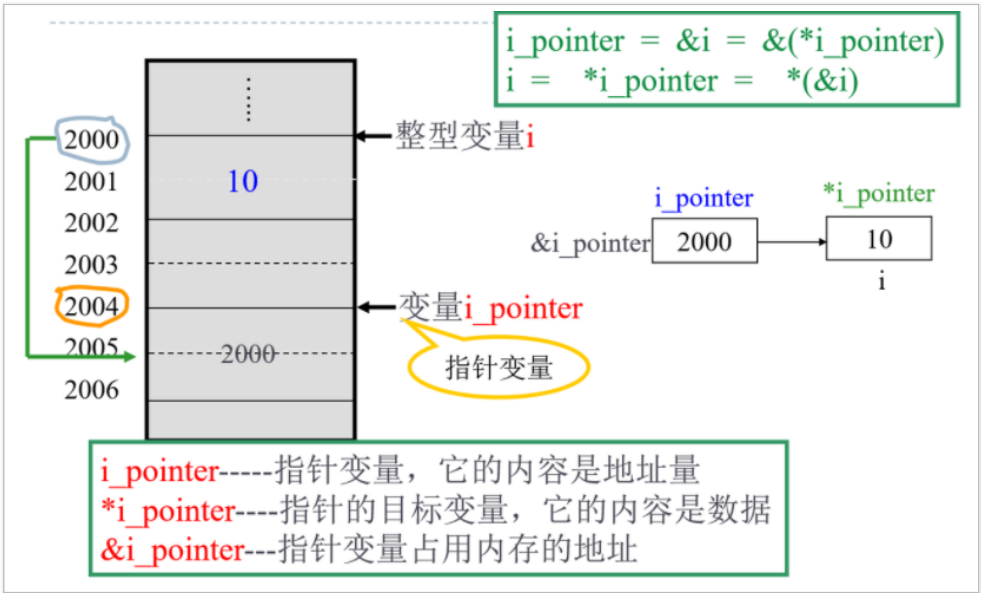
数据类型 \* 指针变量名;  
  
例如:  
char \* p;

```
short *q;
int *m;

int data = 10;

int * p = &data;

*p = 88;
printf("p = %d\n", *p)
```



示例代码:

```
#include <stdio.h>

int main()
{
    int data1 = 0, data2 = 0;
    int * p = &data1;
    int * q = &data2;
    int sum = 0;

    printf("please input two data : ");
    scanf("%d%d", p, q);

    printf("data1 = %d data2 = %d\n", data1, data2);

    sum = *p + *q;
    printf("data1 + data2 = %d\n", sum);

    sum = *p - *q;
    printf("data1 - data2 = %d\n", sum);

    sum = data1 * data2;
    printf("data1 * data2 = %d\n", sum);

    sum = data1 / data2;
    printf("data1 / data2 = %d\n", sum);

    return 0;
}
```

运行结果:

```
please input two data : 20 10
data1 = 20 data2 = 10
data1 + data2 = 30
data1 - data2 = 10
data1 * data2 = 200
data1 / data2 = 2
```

```
int data1 = 10, data2 = 20;  
int *p = NULL;  
int *q = NULL;
```

---

全文完

---

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

