

指针

指针与变量



整型变量的指针

```
#include "stdio.h"

int main()
{ int a=36 , *pa;

  printf("变量a的地址:%d \n变量a的值:%d\n", &a, a);

  pa=&a;

  printf("指针变量pa的值: %d\n", pa);

  printf("指针变量pa所指向的变量的值: %d\n", *pa);

  return 0;

}
```

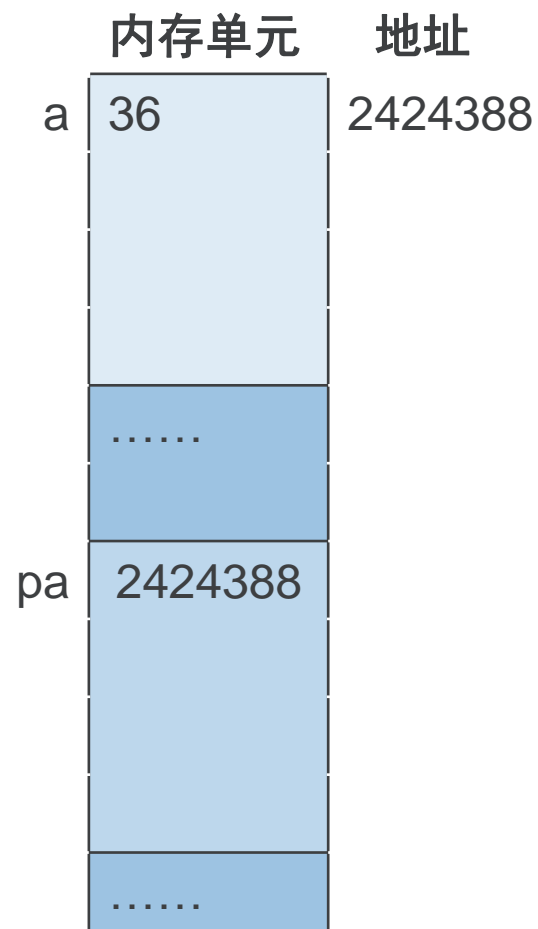
指针运算符 *

*运算的含义是获取*运算符后指针变量的值，
以这个值作为内存地址去寻址，获取内存单元
的内容

*运算符也称为间接引用运算符

运行结果如下图所示：

变量a的地址:2424388
变量a的值:36
指针变量pa的值: 2424388
指针变量pa所指向的变量的值: 36



//输入两个整数，再按值由大到小输出

```
#include "stdio.h"
```

```
int main()
```

```
{  int  *p1,*p2,a,b,t;  //定义指针变量与整型变量
```

```
printf("请输入数据 a,b:");
```

```
scanf("%d,%d",&a,&b);
```

```
printf("初始值: a=%d,b=%d\n",a,b);
```

```
p1=&a; //使指针变量指向整型变量
```

```
p2=&b;
```

```
if(*p1<*p2)//交换指针变量指向的整型变量
```

```
{  t=*p1;    *p1=*p2;    *p2=t;  }
```

```
printf("比较后: a=%d,b=%d\n",a,b);
```

```
return 0;
```

```
}
```

在程序中，当执行赋值操作 $p1=\&a$ 和 $p2=\&b$ 后，指针 $p1$ 保存了变量 a 的地址，指针 $p2$ 保存了变量 b 的地址，引用指针 $*p1$ 与 $*p2$ ，就代表了变量 a 与 b 。

两个指针变量所指向的内容比较之后，交换了两个指针变量所指向的内容，也就是变量 a 和 b 的值。

程序的运行结果如下图所示：

```
请输入数据 a,b:12,45  
初始值: a=12,b=45  
比较后: a=45,b=12
```



陕西科技大学

SHAANXI UNIVERSITY OF SCIENCE & TECHNOLOGY

```
//输入两个整数，再按值由大到小输出
#include "stdio.h"
int main()
{   int   a,b,*p1,*p2,*t;   //定义整型变量与指针变量
printf("input a,b:");
scanf("%d,%d",&a,&b);
printf("初始值: a=%d,b=%d\n",a,b);
p1=&a; p2=&b; //使指针变量指向整型变量
if(*p1<*p2) //交换指针变量的值
{   t=p1;    p1=p2;    p2=t;   }
printf("比较后: a=%d,b=%d\n",a,b);
printf("比较后: 大值%d,小值%d\n",*p1,*p2);
return 0;
}
```

在程序中，当执行赋值操作 $p1=\&a$ 和 $p2=\&b$ 后，指针 $p1$ 保存了变量 a 的地址，指针 $p2$ 保存了变量 b 的地址，引用指针 $*p1$ 与 $*p2$ ，就代表了变量 a 与 b 。

两个指针变量所指向的内容比较之后，交换的是两个指针变量的值。

程序的运行结果如下图所示：

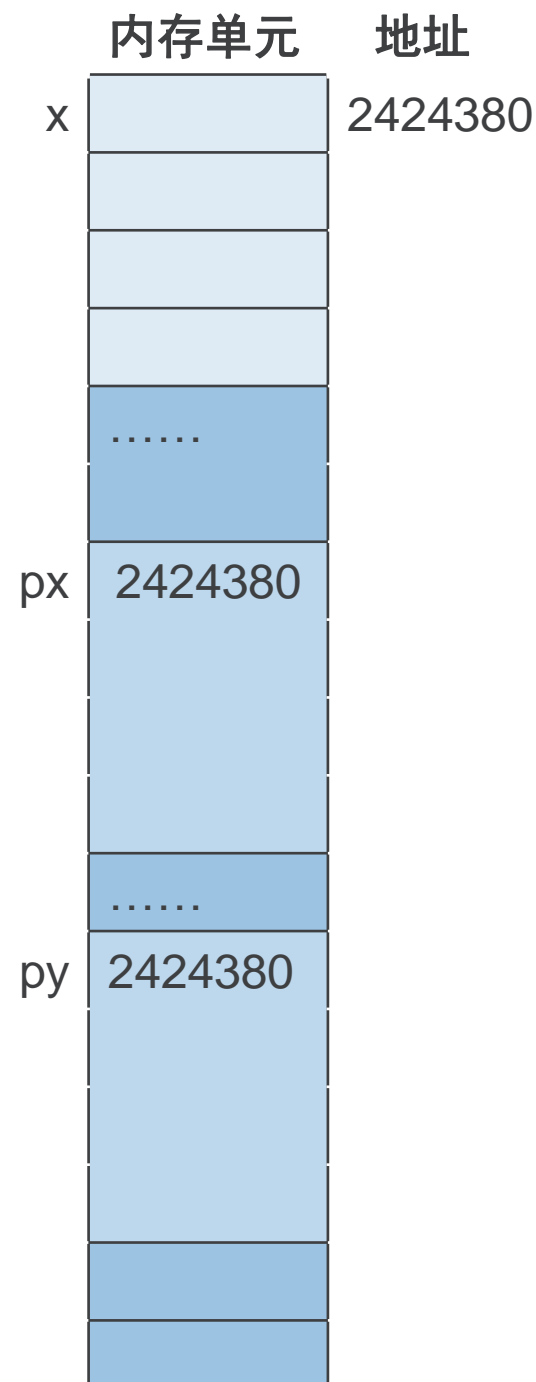
```
输入数据 a,b:12,45
初始值: a=12,b=45
比较后: a=12,b=45
比较后: 大值45,小值12
```



浮点型变量的指针

```
#include "stdio.h"

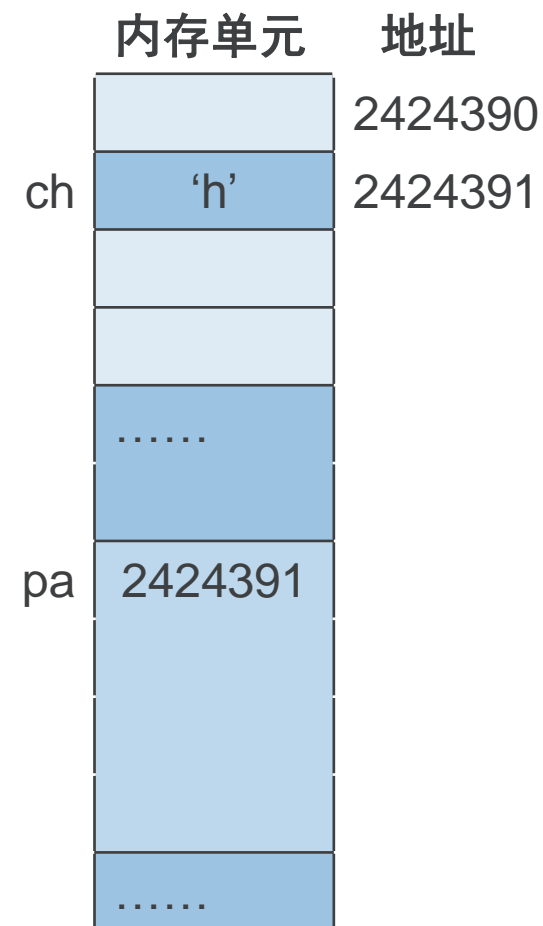
int main()
{
    float x=15.26, *px, *py;
    px=&x;
    py=px;
    printf("%.2f", *py);
    return 0;
}
```



字符型变量的指针

```
#include "stdio.h"
int main()
{
    char ch, *pc ;
    pc=&ch;
    *pc='h' ;
    printf("%c", ch) ;

    return 0;
}
```



THANKYOU

