

lulipro - 代码钢琴家

学习知识的最大价值在于愉悦自己的大脑，其次才是让自己生存。

博客园 首页 新随笔 联系 管理

用我所学去讲C语言指针

文章更新，更加详细的介绍请看这篇：<https://www.cnblogs.com/lulipro/p/7460206.html>

很多人不敢讲C的指针，有些人讲不清，有些人怕讲错。初生牛犊不怕虎，就让我讲讲。

下面开始。

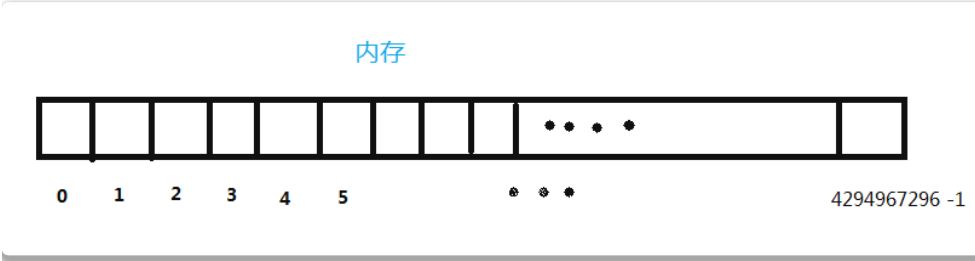
一、指针的定义

指针是内存单元的编号。内存单元是以字节为单位的。所以指针就是字节的编号。

比如我们的个人电脑，内存一般4GB吧，那么一共有： $4 \times 1024 \times 1024 \times 1024 = 4294967296$ 字节，也就是4294967296个编号。一个字节拥有一个编号，

范围从 0 ~ 4294967296-1。

画个图表示：（注意字节由8位bit组成，为了直观我没画出来）

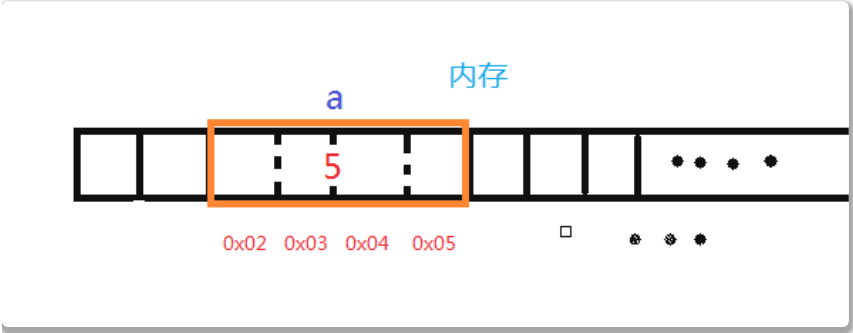


但是呢，一般我们是用16进制来表示的这些编号，但效果都一样。

二、变量与内存的关系

现在来看C 变量在内存中的样子。我们使用自动变量（局部变量）来讲解。

`int a = 5;` //假设a存储在编号为0x02开始的位置



说明：

内存可以存储数据，所以我们把每个字节当做是一个“箱子”。数据存入内存就好比在箱子里面放数据

但是C语言的不同数据类型占用的字节数是不都一样的，所以，每种数据类型占的“箱子”的个数不都一样。

比如char型，只要一个字节就够了，所以一个字符只需一个“箱子”。

而int型需要4（一般是4个字节）个“箱子”才放得下。

double型则需要8个“箱子”。

公告



昵称：lulipro
园龄：3年
粉丝：165
关注：22
+加关注

随笔分类

- Android(3)
- Arduino(9)
- C/C++(8)
- C51(6)
- C语言和操作系统(5)
- FLTK(1)
- FreeRTOS
- Linux学习(3)
- MySQL(1)
- STM32开发(1)
- VisualStudio2017使用(2)
- Win32汇编
- Windows操作系统使用(3)
- 菜米油盐酱醋茶(2)
- 扩展知识(3)
- 设计模式(1)
- 树莓派(5)
- 数据结构(17)
- 刷题记录(4)
- 算法(7)

阅读排行榜

- 1. 一次性搞清楚equals和hashCode(52994)
- 2. C语言指针详解(33642)
- 3. 树莓派wiringPi库详解(26310)
- 4. 在Ubuntu上安装网易云音乐(19613)
- 5. 如何编写自己的Arduino库？(17479)

评论排行榜

- 1. 一次性搞清楚equals和hashCode(14)
- 2. 树莓派wiringPi库详解(12)
- 3. C语言指针详解(10)
- 4. Java中的异常和处理详解(8)

推荐排行榜

- 1. C语言指针详解(20)
- 2. 一次性搞清楚equals和hashCode(14)
- 3. Java中的异常和处理详解(12)
- 4. 【算法】表达式求值-逆波兰算法介绍(7)
- 5. 树莓派wiringPi库详解(6)

来分析上图中用橙色框起来的4个字节的内存块，这里就存储了a这个变量。

我们从4个方面去讨论这个内存块：

1、内存的数据

我们的变量赋值为5，所以内存的数据就是 0000 0000 0000 0000 0000 0000 0000 0101 (大端模式)

每个字节地址： 0x02 0x03 0x04 0x05

2、*内存的名字（对于我们的程序使用的内存来说，并不是每一个内存块都有名字）

名字就是变量名a

3、内存的地址

变量a占用了4个字节，那么，哪一个字节的地址，才是变量a的地址呢？答：第一个低地址字节的地址，也就是0x02

4、内存的宽度

这里的a变量占用了4个字节，这就是他的宽度。

这里提一下，为后面讲指针的宽度做铺垫：)

三、定义和使用指针变量

什么是指针变量。我们知道int类型变量用来存储整形值，12、1003、9823.....

那么，同样的道理，指针变量就是用来保存地址的变量。

定义指针变量：在指向的变量的类型上加个*，如下：

```
int* p_int;           //指向int类型变量的指针

double* p_double;    //指向double类型变量的指针

int (*p_func) (int,int); //指向返回类型为int，有2个int形参的函数的指针

int (*p_arr) [3];     //指向含有3个int元素的数组的指针

struct Student *p_struct; //结构体类型的指针

int** p_pointer;     //指向 一个整形变量指针的指针
```

既然我们已经定义指针变量了，那么接下来就用指针变量来存储其它变量的地址。

取地址符号 &

```
#include<stdio.h>

int Add(int a,int b);

struct Student
{
    int age;
    double score;
    char name[31];
};

int main(void)
{
    int val_int =100 ;
    double val_double = 12.00;
```

```
int arr[3]={1,2,3};

struct Student stu={
    19,
    98.00,
    "Jack"
};

int*p = NULL;

/*****/

int* p_int = &val_int;

double* p_double = &val_double;

int(*p_func)(int,int) = &Add; //or =Add

int(*p_arr)[3] = &arr;

struct Student *p_struct = &stu;

int** p_pointer = &p;

return 0;
}

int Add(int a,int b)
{

    return a+b;

}
```



四、指针的属性和使用

我认为指针有2个属性：①指针的值 ②指针的宽度

指针的值很好理解，比如第一个例子 `int a = 5; int*p = &a;` 那么p的值就是a的地址0x02

指针的宽度：由指针的类型决定。

如果说指针的值标记了某个数据在内存的起始位置，那么，指针的宽度就决定了从起始地址

对应的字节开始，往后还有多少个字节，也是属于这个内存数据的。

生活情景：“老王，去帮我去仓库拿个货，我的货从78号箱开始，并且有2个。” 于是老王取出 78和79号箱子的货物递给你

假如你对老王说：“老王，帮我去仓库取我的货，我的从78号箱开始，一共

有几个箱子，我也记不住了”“那我怎么取，取少了你就有损失了，取多了别人也不干了，你不要难为我啊”

这点我的[另一篇随笔](#)也讲到了。



```
#include<stdio.h>

int main(void)
{

    int a =256 ;

    int* p=&a;
```

```
printf("%d\n", *((unsigned char*)p) ); //读取内存数据时，取的宽度的比存时的宽度小，数据缺失

return 0;
}
```

```
#include<stdio.h>
int main(void)
{
    int arr[2] = {1,2};

    arr[2] = 100;    //ops! 访问越界
    return 0;
}
```

前面我们讲了内存块的4点要素，第4点就是内存块的宽度，他是和指针的宽度是——对应的。一个指针变量指向了这个内存块，

那么指针的宽度就是这个内存块的宽度。这也表明，我们在使用指针的时候，不要越界，也不要取少，取少了取出的数据会不完整，

越界就更严重了，程序会挂掉的，因为你访问了不属于你的内存。

扩展：有木有 没有宽度的指针呢？ 答：这个可以有 void* p;

void* 类型的指针对应与C# or java中的Object类型变量，它可以指向任何类型变量。不过他只保存了内存的起点，没有保存宽度信息，如果你想

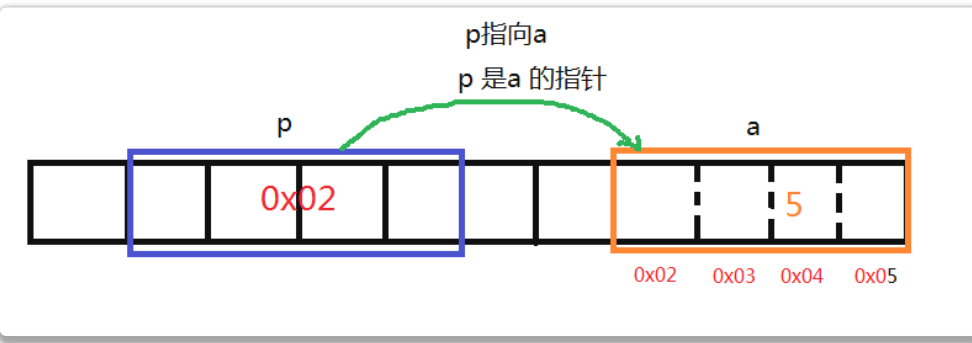
取出原来的数据，你必须做出正确的强制转换。

五：一对相反操作的运算符 * 和 &

& 取地址符 &a 就获取了a的指针，然后我们就可以用int*p变量出承接这个地址 p = &a; 我们就说p指向了a

* 解地址符 *操作符有一个很形象的动词：解。p保存了a的起始地址和延续宽度，那么，*p则是确定起始地址，量出宽度，

获取这个内存块。因此我们可以通过a 的地址p去操作（读/写）这个内存块了。



p也是一个变量，他自己也有地址，这就长产生了指针的指针。

注意：我所说的指针的宽度并不是说指针变量占用的内存大小，而是通过这个指针指向的内存块的宽度。指针的占用的字节数是一定的，一般情况下，指针变量都占用4个字节。上面图中我也画了4个字节。

六、为什么要使用指针


你可能会问：我有变量名，为什么还要绕一转，用指针去使用内存数据？

但并不是这样，下面是2个例子。

①通过动态申请的内存是匿名的，没有名字，只能通过指向这块内存的指针去使用。

malloc() realloc() calloc() 这里不扩展了。

②一个经典的题目：用函数交换2个变量的值。



```
#include<stdio.h>
void swap_2b(int a,int b);
void swap_hack(int *pa,int *pb);
void swap_normal(int*pa,int*pb);


int main()
{
    int a = 5;
    int b = 3;
    swap_2b(a,b);           //Can`t swap;
    swap_normal(&a,&b);      //OK
    swap_hack(&a,&b);        //OK

    return 0;
}

//2b青年写的函数
void swap_2b(int a,int b)
{
    int t;
    t=a;
    a=b;
    b=t;
}

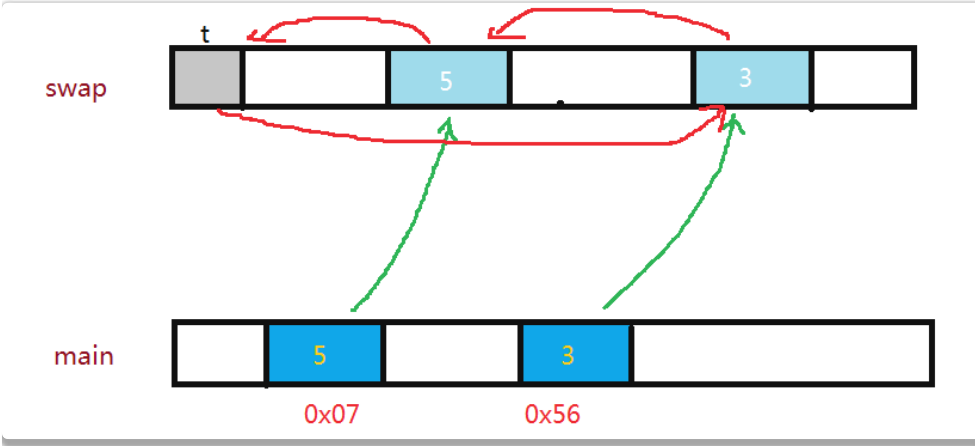
//程序员写的函数
void swap_normal(int*pa,int*pb)
{
    int t;
    t=*pa;
    *pa=*pb;
    *pb=t;
}

//黑客写的函数
void swap_hack(int *pa,int *pb)
{
    *pa = *pa^*pb;
    *pb = *pa^*pb;
    *pa = *pa^*pb;
}
}
```



我们发现只有2b青年没用指针哈。

原因很简单，C语言函数是按值传递的，2b青年写的代码之所以达不到效果是因为它操作的内存块错了。



无论他在函数里面怎么换a和b的值，main函数中的a和b都不会变。因为swap函数中的a和b是随函数调用新生成的变量，是副本，而不是源对象。

但是传递指针就不一样了，因为内存数据的指针是独一无二的。一个人有唯一的身份证一样。

七、野指针和NULL指针

指针变量在使用前或者使用完，好的习惯是赋值为NULL，NULL 是编号为0的字节的地址。指向NULL表示不指向任何变量。

NULL就像剑鞘，野指针就像暗箭，如果你不像被暗箭所伤，就让他归鞘。

最后：由于内容比较多，可能有许多地方表达不当，或有疏漏，错误，希望大家及时指出，谢谢 ：)

作者：lulipro-代码钢琴家

出处：<http://www.cnblogs.com/lulipro/>

本文版权归作者和博客园共有，欢迎转载，但未经作者同意必须保留此段声明，且在文章页面明显位置给出原文连接，否则保留追究法律责任的权利。为了更好的阅读体验，请访问原博客地址。限于本人水平，如果文章和代码有表述不当之处，还请不吝赐教。

分类： C语言和操作系统

好文要顶

关注我

收藏该文

lulipro

关注 - 22

粉丝 - 165

+加关注

4

0

« 上一篇：【C51】单片机芯片之——图解74HC595

» 下一篇：C/C++程序编译流程

posted @ 2015-12-24 17:33 lulipro 阅读(7454) 评论(4) 编辑 收藏

评论列表

- #1楼

2018-07-20 16:53

yxxcmd

lived

支持(0) 反对(0)
- #2楼

2018-12-11 16:51

Speakslowly

got it

支持(0) 反对(0)

#3楼 2018-12-28 18:03 短腿小兽

交换两个变量的值的例子中，程序员的没有交换成功呀！！

支持(0) 反对(0)

#4楼[楼主] 2018-12-28 18:21 lulipro

@ 短腿小兽

可能是因为后面的 swap_hack又交换回来了，请注释掉后面的 swap_hack ，再试。

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册，访问网站首页。

【推荐】超50万VC++源码: 大型组态工控、电力仿真CAD与GIS源码库！

相关博文：

- C语言第八讲.指针*
- C语言 指针
- 我爱我所学
- C语言指针
- c语言指针

最新新闻：

- 蒂姆·库克，中国是一道坎，iPhone是一座山
 - 微软开放Cortana技术：打造你自己的虚拟助手
 - 商汤科技在日本设立自动驾驶研发中心
 - 寻找中国马斯克：商业航天强势破局、资本踴争窗口期
 - 科技股收盘|美三大股指周五微幅调整 奈飞逆市涨近4%
- » 更多新闻...