

函数参数入栈

本文转载自 [itcolossus](#) [查看原文](#) 2009-04-21 343 [语言/](#) [语言/](#) [system/](#) [system/](#) [参数/](#) [参数/](#) [函数/](#) [函数/](#) [list](#) [list](#) ★★☆☆☆☆

上一篇中我们知道三种函数规范

```
#include <stdio.h>
#include <stdlib.h>
void HP(int a,int b,int c,int d)
{
    int f=5;
    int g=7;
    printf("%p/n%p/n%p/n%p/n%p/n",&a,&b,&c,&d,&f,&g);
}

int main(int argc, char *argv[])
{
    HP(2,3,4,5);
    system("PAUSE");
    return 0;
}
```

输出:

0055FF50
0055FF54
0055FF58
0055FF5C
0055FF44
0055FF40

参数入栈

一, 入栈顺序右----》左

二, 地址偏移高----》低

局部变量入栈

先声明的先入栈

C语言中的可变参数处理

```
void arg_cn(char *fmt,...)
{
    printf("%p/t%s/n",fmt,fmt);
    va_list v_l;
    va_start(v_l,fmt);
    while(*v_l)
    {
        char *str=va_arg(v_l,char *);
        printf("%p/t%s/n",v_l,str);
    }
}
```

```
    va_end(v_l);  
}  
int main(int argc, char *argv[])  
{  
    arg_cn("hehe","12","34","56","78");  
    system("PAUSE");  
    return 0;  
}
```

输出:

```
00403026  hehe  
0022FF58  12  
0022FF5C  34  
0022FF60  56  
0022FF64  78
```

呵呵是不是可变参数的处理就是利用参数入栈的规则进行处理的啊