**定理和推导：**

## 定理1：

若x为实数, 则有X向下取整小于等于X, X向上取整大于等于X;



进而可推导：

 推导1

**** 推导2

当x不为整数时：

 推导3

## 定理2：

若x为整数，则x向上取整和向下取整的值都等于x;



## 定理3：

由于上下取整相对于0点是对称的，所以：

, 

进而可推导：

,  推导4

## 定理4

若x为实数， n为整数，有：

, 

结合推导2,可推导：

因, 若x<n, 则

若, 则

而（推导3）, 则可得：x<n 推导5

## 推导6

有两整数a、b和实数x, a!=b 且b!=0, 则有：

若, 则

若,则

**证明：**

设a为a/b的商, r为余数(0<=|r|<|b|, 且q、r均为整数);



 定理4



上面两式减得：

****

因,可得

因(条件), 可得

因(r为余数，b为除数),

且, 结合上式可得：

因, 故

当r、b同号时：



当r、b异号时：

1. **当r>0, b<0时：**

同时除以b得到：,

由此得：

1. **当r<0, b>0时：**

同时除以b得到:

由此得：

由于, 且已证，

可得;

同理可证：当,则

## 推导7

**设有a、b两整数**

**当b>0时, 有：**

**** 且 ****

**当b<0时, 有：**

 且 

**证明1：**

设q为a/b的商, r为余数,



根据定理4, 有：



因b>0, |r|<b, 有：



当r>0时, , ,  成立;

当r=0时, , ,  成立;

当r<0时, , ,  成立;

因此得：

即：

同理可证： 。

**证明2**

设q为a/b的商, r为余数,



根据定理4, 有：



因b<0,|r|<|b|,有：



当r>0时:

 ,  ,  ,  ,  , 成立;

当r=0时:

 ,  ,  成立;

当r<0时：

 , 得到： ;

故: ,  ,  成立;

因此得：



即：

同理可证：

1. **A除以2**

A为无符号数：

**mov** **eax,** A

**shr** **eax,** 1

直接右移

A为有符号数:

**mov** **eax,** A

**cdq** ; 扩展符号位到edx

**sub** **eax,** **edx** ; 如果A>=0; 则dex = 0 ;

; 如果A < 0;则dex = -1;

**sar** **eax,** 1

需要把负数的向下取整修正为向0取整

1. **A除以2^n**

A为无符号数：

**mov** **eax,** A

**shr** **eax,** n

直接右移n位

A为有符号数：

**mov** **eax,** A

**cdq**

**and** **edx,** 2**^**n **-** 1

**add** **eax,** **edx**

**sar** **eax,** n

根据推导7可得：

「⌉ = ⌊⌋

= （A + 2n - 1）>> n

如果A为负数， 则cdq后edx = -1；

**and** **edx,** 2**^**n **-** 1后 edx = 2**^**n **-** 1;正好满足推导7；

如果A>=0, 则则cdq后edx = 0；

**and** **edx,** 2**^**n **-** 1后 edx = 0; 和无符号的情况一样。

1. **MgicNumber无进位或溢出**

A为无符号数：

**mov** **eax,** MagicNumber

**mul** **A**

**shr** **edx,** n

根据公式：



设M = 

上式 = A \* M >> n

N的值越大误差越小，从32起步，n的值随C的值增大而增大。

A为有符号数：

**mov** **ecx,** A

**mov** **eax,** MagicNumber

**imul** **ecx**

**sar** **edx,** n ;隐式右移了32位，实际右移32+n

;后面三行调整定式

**mov** **eax,** **edx**

**shr** **eax,** 1Fh ;得到符号位

**add** **edx,** **eax**

和无符号数相比，多了三行调整定式

1. **MgicNumber无进位或溢出**

A为无符号数**MgicNumber**有进位：

**mov** **ecx,** A

**mov** **eax,** MagicNumber

**mul** **ecx**

**sub** **ecx,** **edx**

**shr** **ecx,** n

**add** **ecx,** **edx**

**shr** **ecx,** m

将上式转换为数学模型

其中

**mul** **ecx**

**sub** **ecx,** **edx**

中的edx为积的高32位，等价积除以232;



化简后得:



设为M, 则上式可简化为



此时M是一个超过了32位的数值，巧妙的运用未简化的式子正好可以实现这个功能，并且不需要使用大数库，而且时钟周期也小于除法的时钟周期，所以几乎所有优秀的编译器的无符号数除法都是通过这个式子变换而来的。

A为有符号数，**MgicNumber**有溢出位：

**mov** **ecx,** A

**mov** **eax,** MagicNumber

**imul** **ecx**

**add** **edx,** **ecx**

**sar** **edx,** n

;后面三行为调整定式

**mov** **eax,** **edx**

**shr** **eax,** 1Fh

**add** **edx,** **eax**

先搞清楚有符号数与无符号数的乘法的汇编代码：

设a为16位正整数,用汇编求a \* 8086h？

求补的实质：Neg(a) = -(10000h - a);

mov ax, 8086h

imul a

此时8086h会被当做负数即：

a \* 8086h

= a \* -(10000h - 8086h)

= a \* (8086h - 10000h)

= 8086h \* a - 10000h \* a

若要等价a\*8086h,则需要加入调整运算;

a \* 8086h = 8086h \* a - 10000h \* a + 10000h \* a;

而10000h \* a刚好对应高16位, imul a的值为dx.ax;

由此可得,调整后的值为：

10000h \* a + dx.ax = (dx + a).ax;

即最终a \* 8086h的汇编指令

mov ax, 8086h

imul a

add dx, a

当进行有符号的除法时, 如果MagicNumber > 7FFFFFFFh,则会把MagicNumber当做负数,后面的结果需要调整，

**除以负数**

1. A除以-2

**mov** **eax,** A

**cdq**

**sub** **eax,edx**

**sar** **eax,**1

**neg** **eax**

1. A除以2的幂的负数

**mov** **eax,** A

**cdq**

**and** **edx,** 2**^**n**-**1

**add** **eax,** **edx**

**sar** **eax,** n

**neg** **eax**

1. A除以非2的幂的负数

当MagicNumber < 0时：

**mov** **eax,** MagicNumber

**imul** **esi**

**sar** **edx,** n

**mov** **eax,** **edx**

**shr** **eax,** 1Fh

**add** **edx,** **eax**



首先要明白除以负数实际就是对除以正数的结果求补，

等价对MagicNumber求补，即：



当MagicNumber > 0时：

**mov** **ecx,** A

**mov** **eax,** MagicNumber

**imul** **ecx**

**sub** **edx,** **ecx**

**sar** **edx,** n

**mov** **eax,** **edx**

**shr** **eax,** 1Fh

**add** **edx,** **eax**

首先要明白除以负数实际就是对除以正数的结果求补，

设M为MagicNumber，对M求补得：



= 

而对应调整除法转换公式为：



将M代进公式得:

= 

将汇编代码转换为数学模型：

 =  = 

根据上式可得：

C = 

而C是负数，|C|= -C

-C = 