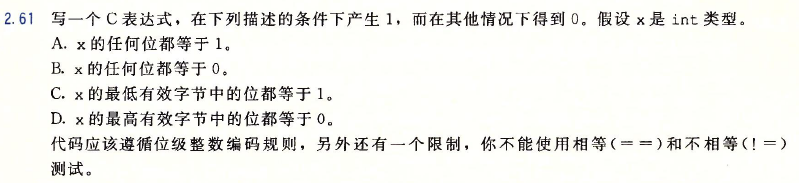
作业2

2.61、2.63、2.65、2.71、2.72、2.77、2.87、2.90



**A. !~x**

**B. !x**

**C. !(x & 0xff)**

**D. !((x >> ((sizeof(int)-1) << 3)) & 0xff)**



**unsigned srl(unsigned x, int k) {**

**unsigned xsra = (int) x >> k;**

**int temp = (int) -1 << ((8 \* sizeof(int)) – k);**

**return xsra & ~temp;**

**}**

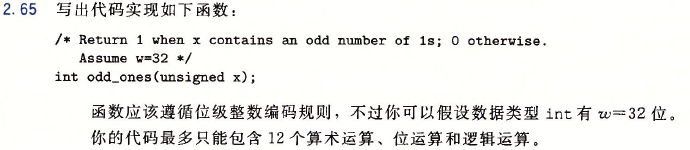
**int sra(int x, int k) {**

**int xsrl = (unsigned) x >> k;**

**unsigned temp = 2 << ((8 \* sizeof(int)) – k – 1) – 1;**

**return xsra & temp;**

**}**



**int odd\_ones(unsigned x) {**

**x ^= x >> 16;**

**x ^= x >> 8;**

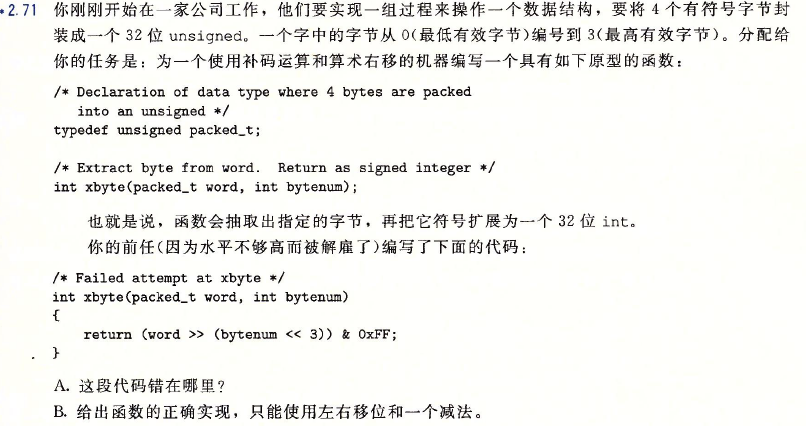
**x ^= x >> 4;**

**x ^= x >> 2;**

**x ^= x >> 1;**

**return (x & 0x1);**

**}**



**A. 得到的结果是unsigned，即该代码只能处理正数而无法正确扩展出负数。**

**B. 将原始数据转化为int，待抽取字节先左移到最高字节再右移到最低字节。**

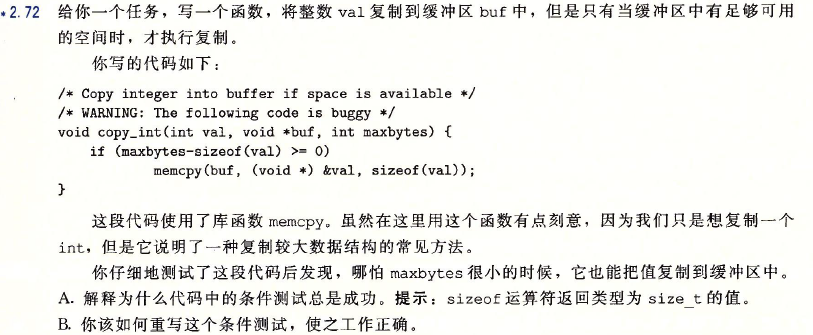
**int xbyte(packed\_t word, int bytenum)**

**{**

**int ret = word << ((3 – bytenum) << 3);**

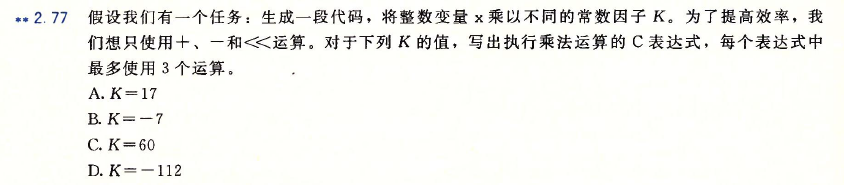
**return ret >> 24;**

**}**



**A. sizeof运算符返回的size\_t是无符号整数，因此判断条件中左边被减数maxbytes也会转换为无符号整数进行运算，得到的差为无符号整数，大于等于0恒成立。**

**B. 判断条件改为if(maxbytes > 0 && maxbytes >= sizeof(val))**



**A. C = (x << 4) + x**

**B. C = x – (x << 3)**

**C. C = (x << 6) – (x << 2)**

**D. C = (x << 4) - (x << 7)**



**-1**

**123/64**

**123/128**

**0.9609375**

**FC00**

**6.09755516e-5**

**1023/2^24**

**1023/1024**

**-14**

**03FF**

**9**

**1**

**6000**

**2.00195312**

**1025/512**

**1**

**1025/1024**

**4001**

**-14**

**0**

**8000**



**1<<(x+149)**

**0**

**-126**

**0**

**-149**

**0**



**0**

**255**

**0**

**x+127**

**128**

**float的k = 8，n = 23，bias = 2^(k-1) - 1 = 127。**

**最小的正非规格化数为2^(1-bias-n) = 2^-149。**

**最小的规格化数为2^(0-bias)\*2 = 2^-126。**

**最大的规格化数为2^(2^k-2-bias) = 2^127。**