**考试科目名称　计算机组织结构期中试卷**

2016—2017学年第 2 学期　教师 唐杰 考试方式：开卷

系（专业）　　 学号　　　 　　　　姓名　　 　　　　成绩

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 题号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 分数 |  |  |  |  |  |  |  |  |  |  |  |

1. 简述从源程序文件到可执行目标文件的转换过程（本题5分）

答：

（1）预处理阶段：预处理程序（cpp）对源程序中以字符#开头的命令进行处理，例如，将#include命令后面的.h文件内容嵌入到源程序文件中。预处理程序的输出结果还是一个源程序文件，以.i为扩展名。

（2）编译阶段：编译程序（cc1）对预处理后的源程序进行编译，生成一个汇编语言源程序文件，以.s为扩展名，

（3）汇编阶段：汇编程序（as）对汇编语言源程序进行汇编，生成一个可重定位目标文件（relocation object file），以.o为扩展名，是一种二进制文件（binary file），因为其中的代码已经是机器指令，数据以及其他信息也都是用二进制表示的，所以它是不可读的，也即打开显示出来的是乱码。

（4）链接阶段：链接程序（ld）将多个可重定位目标文件和标准库函数合并成为一个可执行目标文件（executable object file），可执行目标文件可简称为可执行文件。本例中，链接器将hello.o和标准库函数printf所在的可重定位目标模块printf.o进行合并，生成可执行文件hello。

1. 假设某机器M的时钟频率为2GHz，用户程序P在M上的指令条数为1×109，其CPI为1.5，则P在M上的执行时间是多少？若在机器M上从程序P开始启动到执行结束所需的时间是2秒，则程序P的用户CPU时间占用的百分比是多少？（本题5分）

答：

程序P在机器M上执行所需时钟周期数为1×109 ×1.5 = 1.5×109，所需时间为1.5×109 / (2×109) = 0.75s，程序P的用户CPU时间在2秒钟内占用的百分比是0.75/2×100% = 37.5% 。

1. 已知 [x]补，求x的真值。（本题5分）

（1）[x]补=1110 0001 （2）[x]补=1000 0000

（3）[x]补=0111 1111 （4）[x]补=1111 1111

答：

（1）x= –1 1111B= –31 （2）x= –1000 0000B= –128

（3）x= 111 1111B=127 （4）x= –00000001B = –1

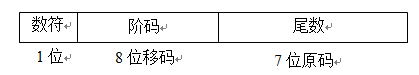
1. 下列几种情况所能表示的数的范围是什么？（本题5分）

（1） 16位无符号整数。

（2） 16位原码定点小数。

（3） 16位补码定点整数。

（4） 下述格式的浮点数（基数为2，移码的偏置常数为128，规格化尾数，不考虑隐藏位）



数符

S1

阶码

S1

尾数

S1

1位

S1

8位移码

S1

7位原码

S1

数符

S1

阶码

S1

尾数

S1

1位

S1

8位移码

S1

7位原码

S1

数符

S1

阶码

S1

尾数

S1

1位

S1

8位移码

S1

7位原码

S1

答：

（1）16位无符号整数范围为0～216–1，即0～65535。

（2）16位原码定点小数表示的范围为 –(1–2-15)～+(1–2-15)。

（3）16位补码定点整数表示的范围为 –215～+(215–1)，即–32768～+32767。

（4）规格化浮点数的表示范围如下。

最大正数：+0.111 1111B×21111 1111B = +(1–2-7)×2127。

最小正数：+0.100 0000B×20000 0000B = +2-1×2-128 = +2-129。

最大负数：–0.100 0000B×20000 0000B =–2-1×2-128 = –2-129。

最小负数：–0.111 1111B×21111 1111B =–(1–2-7)× 2127。

1. 假设某字长为8位的计算机中，带符号整数采用补码表示，x=–68，y=–80，x和y分别存放在寄存器A和B中。请回答下列问题（要求最终用十六进制表示二进制序列）。（本题10分）

（1）寄存器A和B中的内容分别是什么？

（2）若x和y相加后的结果存放在寄存器C中，则寄存器C中的内容是什么？运算结果是否正确？加法器最高位的进位Cout是什么？溢出标志OF、符号标志SF和零标志ZF各是什么？

（3）若x和y相减后的结果存放在寄存器D中，则寄存器D中的内容是什么？运算结果是否正确？此时，加法器最高位的进位Cout是什么？溢出标志OF、符号标志SF和零标志ZF各是什么？

（4）对于带符号整数的减法运算，能否直接根据CF的值对两个带符号整数的大小进行比较？

答：

（1）[–68]补=[–1000100]补=1011 1100B=BCH。[–80]补=[–1010000 ]补=1011 0000B=B0H。所以，寄存器A和B中的内容分别是BCH和B0H。

（2）[x+y]补=[x]补+[y]补=1011 1100 B+ 1011 0000 B= (1) 0110 1100B = 6CH，最高位前面的一位1被丢弃，因此，寄存器C中的内容为6CH，对应的真值为+108，结果不正确。加法器最高位向前面的进位Cout为1。溢出标志OF为1，说明寄存器C中的内容不是正确的结果。结果的第一位（最高位）0为符号标志位SF，即SF=0，表示结果为正数；因为结果不为0，所以零标志ZF=0。

（3）[x–y]补=[x]补+[–y]补=1011 1100 B+ 0101 0000B = (1) 0000 1100B = 0CH，最高位前面的一位1被丢弃，因此，寄存器D中的内容为0CH，对应的真值为+12，结果正确。加法器最高位向前面的进位Cout为1。两个加数的符号位相异一定不会溢出，因此溢出标志OF=0，说明寄存器D中的内容是真正的结果；结果的第一位（最高位）0为符号标志位SF，即SF=0，表示结果为正数；因为结果不为0，所以零标志ZF=0。

（4）对于带符号整数的减法运算，无法直接根据CF的值判断两个带符号整数的大小。

1. 假定一台32位字长的机器中带符号整数用补码表示，浮点数用IEEE 754标准表示，寄存器R1和R2的内容分别为8020 0000H和0080 0000H。不同指令对寄存器进行不同的操作，因而，不同指令执行时寄存器内容对应的真值不同。假定执行下列运算指令时，操作数为寄存器R1和R2的内容，则R1和R2中操作数的真值分别为多少？（本题10分）
   1. 无符号数加法指令
   2. 带符号整数乘法指令
   3. 单精度浮点数减法指令

答：

寄存器R1的内容为1000 0000 0010 0000 0000 0000 0000 0000，寄存器R2的内容为0000 0000 1000 0000 0000 0000 0000 0000。

（1）对于无符号数加法指令，R1 和R2的内容均被解释成无符号整数，即R1的真值为8020 0000H, R2的真值为80 0000H，也即R1的真值为231+221，R2的真值为223。

（2）对于带符号整数乘法指令，R1和R2的内容均被解释为补码整数，由最高位可知， R1为负数， R2为正数。R1的真值为 – 0111 1111 1110 0000 0000 0000 0000 0000B = –7FE0 0000H；R2的真值为+80 0000H。也即R1的真值为–(231–221)，R2的真值为223。

（3）对于单精度浮点数减法指令，R1和R2的内容均为IEEE 754单精度浮点数表示。由R1的内容可知，其符号位为1， 表示负数， 阶码为0000 0000， 尾数部分为010 0000 0000 0000 0000 0000，因为阶码为全0尾数为非0数，故R1是非规格化浮点数，其指数为–126，尾数为0.01B， 故R1表示的真值为–0.01B×2-126 = –2-128。由R2的内容可知，其符号位为0， 表示正数，阶码为0000 0001，尾数部分为000 0000 0000 0000 0000 0000，R1为规格化浮点数，其指数为1–127 = –126，尾数为1.0B， 故R2表示的真值为+1.0×2-126 =2-126。

1. 假定在一个程序中定义了变量x、y和i，其中，x和y是float型变量（用IEEE 754单精度浮点数表示），i是16位short型变量（用补码表示）。程序执行到某一时刻，x= –130、y=7.25、i=130，它们都被写到了主存（按字节编址），其地址分别是&x，&y和&i。请分别给出在大端机器和小端机器上变量x、y和i在内存的存放情况。（本题10分）

答：

x = –130 = –100 00010B = –1.00 0001B×27, 阶码e=127+7=128+6=1000 0110, 所以，用IEEE 754单精度浮点数表示为：1 100 0011 0 000 0010 0000 0000 0000 0000= C302 0000H。

y=7.25= 111.01B= +1.1101B×22，阶码e=127+2=128+1=1000 0001, 所以，用IEEE 754单精度浮点数表示为：0 100 0000 1 110 1000 0000 0000 0000 0000= 40E8 0000H。

i = 130 = 1000 0010B，用16位补码表示为0082H。

上述三个数据在大端机器和小端机器上的存放位置为。

|  |  |  |
| --- | --- | --- |
| 地址 | 大端机器 | 小端机器 |
| &x | C3H | 00H |
| &x + 1 | 02H | 00H |
| &x + 2 | 00H | 02H |
| &x + 3 | 00H | C3H |
| &y | 40H | 00H |
| &y + 1 | E8H | 00H |
| &y + 2 | 00H | E8H |
| &y + 3 | 00H | 40H |
| &i | 00H | 82H |
| &i + 1 | 82H | 00H |

1. 对于一个n（n≥8）位的变量x，请根据C语言中按位运算的定义，写出满足下列要求的C语言表达式。（本题10分）

（1）x的最高有效字节不变，其余各位全变为0。

（2）x的最低有效字节不变，其余各位全变为0。

（3）x的最低有效字节全变为0，其余各位取反。

（4）x的最低有效字节全变1，其余各位不变。

答：

（1）(x>>(n-8))<<(n-8)

（2）x & 0xFF

（3）((x^ ~0xFF) >>8 )<< 8

（3）x | 0xFF

1. 假设某个C语言函数func的原型声明如下：（本题10分）

void func(int \*xptr, int \*yptr, int \*zptr);

函数func的过程体对应的机器级代码用AT&T汇编形式表示如下：

1 movl 8(%ebp), %eax

2 movl 12(%ebp), %ebx

3 movl 16(%ebp), %ecx

4 movl (%ebx), %edx

5 movl (%ecx), %esi

6 movl (%eax), %edi

7 movl %edi, (%ebx)

8 movl %edx, (%ecx)

9 movl %esi, (%eax)

请回答下列问题或完成下列任务。

（1）在过程体开始时三个入口参数对应实参所存放的存储单元地址是什么？（提示：当前栈帧底部由帧指针寄存器EBP指示）

（2）根据上述机器级代码写出函数func的C语言代码。

答：

（1）xptr、yptr和zptr对应实参所存放的存储单元地址分别为：R[ebp]+8、R[ebp]+12、R[ebp]+16。

（2）函数func的C语言代码如下：

void func(int \*xptr, int \*yptr, int \*zptr)

{

int tempx=\*xptr;

int tempy=\*yptr;

int tempz=\*zptr;

\*yptr=tempx;

\*zptr = tempy;

\*xptr = tempz;

}

1. 已知函数f1的C语言代码框架及其过程体对应的汇编代码如图3.3所示，根据对应的汇编代码填写C代码中缺失部分，并说明函数f1的功能。（本题10分）

1 int f1(unsigned x)

2 {

3 int y = 0 ;

4 while ( ) {

5 ;

6 }

7 return ;

8 }

1 movl 8(%ebp), %edx

2 movl $0, %eax

3 testl %edx, %edx

4 je .L1

5 .L2:

6 xorl %edx, %eax

7 shrl $1, %edx

8 jne .L2

9 .L1:

10 andl $1, %eax

答：

1 int f1(unsigned x)

2 {

3 int y = 0 ;

4 while ( x!=0 ) {

5 y ^=x ;

6 x>>=1 ;

7 }

8 return y&0x1 ;

9 }

函数f1的功能返回：( x ^ x>>1 ^ x>>2 ^ ….) & 0x1，因此f1用于检测x的奇偶性，当x中有奇数个1，则返回为1，否则返回0。

1. 已知x = 10，y = – 6，采用5位机器数表示。请按如下要求计算，并把结果还原成真值。（本题20分）
   1. 用原码一位乘法计算[x×y]原。
   2. 用布斯乘法计算[x×y]补。
   3. 用不恢复余数法计算[x/y]原的商和余数。
   4. 用恢复余数法计算[x/y]原的商和余数。

答：

1. [x]原=01010，[y]原=10110

C P Y 说明

0 0 0 0 0 0 1 1 0 P0 = 0

+ 0 0 0 0 y0 = 0，不作加法（加0）

0 0 0 0 0 C, P 和Y同时右移一位

0 0 0 0 0 0 0 1 1 得P1

+ 1 0 1 0 y1 = 1，+X

0 1 0 1 0 C, P 和Y同时右移一位

0 0 1 0 1 0 0 0 1 得P2

+ 1 0 1 0 y2 = 1，+X

0 1 1 1 1 C, P 和Y同时右移一位

0 0 1 1 1 1 0 0 0 得P3

+ 0 0 0 0 y3 = 0，不作加法（加0）

0 0 1 1 1 C, P 和Y同时右移一位

0 0 0 1 1 1 1 0 0 得P4

符号位为0 ⊕ 1 = 1，因此，[X×Y]原 = 00111100，10进制表示为60，但因为是5位机器数，故返回值为11100，其真值溢出。

b）[x]补=01010，[y]补=11010，[-x]补=10110

P Y y-1 说明

0 0 0 0 0 1 1 0 1 0 0 设y-1 = 0，[P0]补 = 0

y0 y-1 = 00，P、Y直接右移一位

0 0 0 0 0 0 1 1 0 1 0 得[P1]补

+ 1 0 1 1 0 y1 y0 =10，+[–X]补

1 0 1 1 0 P、Y同时右移一位

1 1 0 1 1 0 0 1 1 0 1 得[P2]补

+ 0 1 0 1 0 y2 y1 =01，+[X]补

0 0 1 0 1 P、Y直接右移一位

0 0 0 1 0 1 0 0 1 1 0 得[P3]补

+ 1 0 1 1 0 y3 y2 = 10，+[-X]补

1 1 0 0 0 P、Y同时右移一位

1 1 1 0 0 0 1 0 0 1 1 得[P4]补

Y4 y3 = 11，P、Y直接右移一位

1 1 1 1 0 0 0 1 0 0 1 得[P5]补

因此，[X × Y]补=11110 00100，10位真值为-60，但因为是5位机器数表示，所以结果溢出。

C） [|x|]原=1010，[|y|]原=00110，[-|y|]原=11010

余数寄存器R 余数/商寄存器Q 说 明

0 0 0 0 0 1 0 1 0 开始R0 = X

+ 1 1 0 1 0 R1 = X–Y

1 1 0 1 0 0 0 0 0 0 R1< 0，则q 4 = 0

+ 0 0 1 1 0 恢复余数：R1= R1+ Y

0 0 0 0 0 得R1

0 0 0 0 1 0 1 0 0 2R1（R和Q同时左移，空出一位商）

+ 1 1 0 1 0 R2 = 2R1–Y

1 1 0 1 1 0 1 0 0 0 R2 < 0，则q3 = 0

+ 0 0 1 1 0 恢复余数：R1= R1+ Y

0 0 0 0 1 得R1

0 0 0 1 0 1 0 0 0 2R2 （R和Q同时左移，空出一位商）

+ 1 1 0 1 0 R3 = 2R2 –Y

1 1 1 0 0 1 0 0 0 0 R3 < 0，则q2= 0

+ 0 0 1 1 0 恢复余数：R1= R1+ Y

0 0 0 1 0 得R1

0 0 1 0 1 0 0 0 0 2R3 （R和Q同时左移，空出一位商）

+ 1 1 0 1 0 R4 = 2R3 –Y

1 1 1 1 1 0 0 0 0 0 R4 < 0，则q1= 0

+ 0 0 1 1 0 恢复余数：R4 = R4 + Y

0 0 1 0 1 0 0 0 0 0 得R4

0 1 0 1 0 0 0 0 0 2R4 （R和Q同时左移，空出一位商）

+ 1 1 0 1 0 R5 = 2R4 –Y

0 0 1 0 0 0 0 0 0 1 R5 > 0，则q0 = 1

商的符号位：0 ⊕ 1 = 1，商的最高位为0，说明没有溢出，商的数值部分为：0001。

所以，[X/Y]原=10001 (最高位为符号位)，余数为0100。

d）[|x|]原=1010，[|y|]原=00110，[-|y|]原=11010

余数寄存器R 余数/商寄存器Q 说 明

0 0 0 0 0 1 0 1 0 开始R0 = X

+ 1 1 0 1 0 R1 = X–Y

1 1 0 1 0 1 0 1 0 0 R1< 0，则q 4 = 0，没有溢出

1 0 1 0 1 0 1 0 0 2R1（R和Q同时左移，空出一位商）

+ 0 0 1 1 0 R2 = 2R1+Y

1 1 0 1 1 0 1 0 0 0 R2 < 0，则q 3 = 0

1 0 1 1 0 1 0 0 0 2R2 （R和Q同时左移，空出一位商）

+ 0 0 1 1 0 R3 = 2R2 +Y

1 1 1 0 0 1 0 0 0 0 R3 < 0，则q 2 = 0

1 1 0 0 1 0 0 0 0 2R3 （R和Q同时左移，空出一位商）

+ 0 0 1 1 0 R4 = 2R3 –Y

1 1 1 1 1 0 0 0 0 0 R4 < 0，则q1 = 0

1 1 1 1 0 0 0 0 0 2R4 （R和Q同时左移，空出一位商）

+ 0 0 1 1 0 R5 = 2R4 +Y

0 0 1 0 0 0 0 0 0 1 R5 > 0，则q 0 = 1

商的符号位：0 ⊕ 1 = 1，商的最高位为0，说明没有溢出，商的数值部分为：0001。

所以，[X/Y]原=10001 (最高位为符号位)，余数为0100。