

（深圳）

实验报告

开课学期： 2022春季

课程名称：计算机组成原理（实验）

实验名称： 从C语言到机器码

实验性质： 综合设计型

实验学时： 2 地点： T2612

学生班级： 11

学生学号： 200111132

学生姓名： 吴桐

作业成绩：

实验与创新实践教育中心制

2022年3月

|  |
| --- |
| 1、实验结果截图 |
| （需贴出执行文件运行的结果截图）  立方计算  开的 02优化  C:\Users\Wufisher\Documents\Tencent Files\1105889424\Image\C2C\Image1\`698$G%[%V09ISHW2[K6`F2.png  C:\Users\Wufisher\Documents\Tencent Files\1105889424\Image\C2C\Image1\V8564)`VO($%5JBATH_E[ZW.png |
| 2、汇编代码注释（只需写主程序和子程序即可） |
| * 示例： * 将堆栈指针寄存器sp与立即数(-16)相加，再存入堆栈指针寄存器sp，即sp = sp + (-16) * muti: * addiw a3,a0,-1   a0（0）低32位 加上 -1 后64位符号扩展，存入a3   * mv a0,a3   将a3寄存器里面的值存入a0   * sllw a1,a1,a3   将a1低32位逻辑左移a3的低五位，做64位符号扩展并存入a1   * beq a3,zero,.L2   如果a3==0，则跳转到.L2标签   * li a5,0   将0存入a5   * li a0,0   将0存入a0   * .L4:   + srlw a4,a2,a5   a2低32位逻辑右移，位数为a5的低5位，64位符号扩展存放在a4   * + andi a4,a4,1   a4 加 1 结果放在a4   * + addiw a5,a5,1   a5的低32位 加立即数1，符号扩展后放在a5   * + beq a4,zero,.L3   如果a4等于0，则跳转到.L3   * + addw a0,a1,a0   a1和a0低32位相加，符号拓展后放在a0   * .L3:   + srliw a0,a0,1   a0低32 位逻辑右移，符号拓展后放在a0   * + bne a5,a3,.L4   如果a5 a3不相等，跳到L4   * main:   + li a5,0   a5置0   * + li a3,0   a3 置0   * + li a0,7   a0 置7   * + li a1,32   a1 置 32   * + li a6,4096   a6 置4096   * .L14:   + addiw a5,a5,1   a5低32位+1,.符号拓展结果放在a5   * + srliw a3,a3,1   a3低32位逻辑右移1位，符号拓展结果放在a3   * + srlw a4,a1,a5   a1 低32位逻辑右移a5的第5位，符号拓展放在a4   * + mv a2,a3   a2置为a3的值   * + andi a4,a4,1   a4和1取并后放在a4   * + beq a5,a0,.L26   如果a5等于a0跳到.L26   * .L16:   + beq a4,zero,.L14   如果a4等于0跳转到.L14   * + addw a3,a6,a2   a6低32位加上a2，符号拓展结果放在a3   * + addiw a5,a5,1   a5第32位+1，符号拓展放在a5   * + srliw a3,a3,1   a3 低32位逻辑右移1位，符号拓展放在a3   * + srlw a4,a1,a5   a1低32位右移位数为a5低5位，符号拓展放在a4   * + mv a2,a3   a2置为a3的值   * + andi a4,a4,1   a4和1取并后放在a4   * + bne a5,a0,.L16   a5和a0不相等跳转到.L16   * .L26:   + li a2,0   a2置0   * + li a5,0   a5置0   * + li a0,1048576   a0置1048576   * + li a1,15   a1置15   * .L18:   + srlw a4,a3,a5   a3 逻辑右移a5低5位，符号拓展放在a4   * + andi a4,a4,1   a4和1取并，放在a4   * + addiw a5,a5,1   a5 加 1，结果放在a5   * + beq a4,zero,.L17   a4 等于1则跳转到.L17   * + addw a2,a0,a2   a0低32位加a2符号拓展结果在放在a2   * .L17:   + srliw a2,a2,1   a2逻辑右移1位，放在a2   * + bne a5,a1,.L18   a5和a1不相等跳转到L18   * + lui a0,%hi(.LC0)   a0装.LC0地址的高20位绝对值   * + li a1,32   a1装32   * + addi a0,a0,%lo(.LC0)   a0加上.LC0低12位绝对值放在a0   * + tail printf   跳转到printf   * + .size main, .-main   Main设为可用源？（.size有点难理解）   * + .ident "GCC: (GNU) 9.2.0"   "GCC: (GNU) 9.2.0"设为source compatibility |
| 3、机器码注释（只需写主程序和子程序即可） |
| 示例：    1141：二进制为0001 0001 0100 0001  fun3：000，imm：110000，rd/rs1：00010，op：01  c.addi指令：sp = sp + (-16)  0000000000000000 <muti>:     0: fff5069b            addiw a3,a0,-1  1111 1111 1111 0101 0000 0110 1001 1011  Imm:1111 1111 1111, rs1:01010,fun3:000,rs1:01101  Opcode:0011011  Addiw: a3 = a0+(-1)     4: 8536                  mv  a0,a3  1000 0101 0011 0110  Fun3:100,imm:0,rd:01010,rs2:01101,op:10  c.mv: A0=a3     6: 00d595bb            sllw  a1,a1,a3  0000 0000 1101 0101 1001 0101 1011 1011  Func7:0000000,rs2:01101,rs1:01011,fun3:001,rd:01011,op:0111011  Sllw: A1=a1>>a3[4:0]     a: ce89                  beqz  a3,24 <.L2>  1100 1110 1000 1001  Fun3:110,rs1’:101,op:01,offset:00001101(0)     c: 4781                  li  a5,0  c.beqz if a3==0 pc+=26 (pc=.L2)     e: 4501                  li  a0,0  0100 0101 0000 0001  Fun3:010,imm[5]0,rd:01010,imm[4:0]:00000,op:01  c.li: A0=0  0000000000000010 <.L4>:    10: 00f6573b            srlw  a4,a2,a5  0000 0000 1111 0110 0101 0111 0011 1011  Funct7:0000000,rs2:01111,rs1:01100 fun3:101 rd:01110 op:0111011  A4=a2>>a5[4:0]    14: 8b05                  andi  a4,a4,1  1000 1011 0000 0101  Fun3:100 ummi[5]=0 rd`=110 uimm[4:0]:00001 op 01  c.andi: a4=a4&1    16: 2785                  addiw a5,a5,1  0010 0111 1000 0101  Fun3:001 imm：000001 rd:01111 op:01  c.addiw: a5=a5+1    18: c311                  beqz  a4,1c <.L3>  1100 0011 0001 0001  Func3: 110 offset[8|4:3]:000 rs1’:110 offset[7:6|2:1|5]:00100 op:01  c.beqz :If a4==0 pc=.L3    1a: 9d2d                  addw  a0,a0,a1  1001 1101 0010 1101  Fun6: 100111 rd’:010 func2:01 rs2’:011 op:01  c.addw: a0=a0+a1  000000000000001c <.L3>:    1c: 0015551b            srliw a0,a0,0x1  0000 0000 0001 0101 0101 0101 0001 1011  Imm：000000000001，rs1:01010,func3:101,rd:01010,op:0011011  Srliw: A0=a0>>1    20: fed798e3            bne a5,a3,10 <.L4>  1111 1110 1101 0111 1001 1000 1110 0011  Imm:1111111 10001 rs2:01101 rs1:01111 fun3:001 op:1100011  Bne: if a5!=a3 pc = .l4  0000000000000024 <.L2>:    24: 8082                  ret  1000 0000 1000 0010  Func3:100 imm:000000 rs1:00001 op:01  c.jr pc=x1  0000000000000000 <main>:     0: 4781                  li  a5,0  0100 0111 1000 0001  Func: 010 imm:000000 rd:01111 op:01  c.li:a5=0     2: 4681                  li  a3,0  0100 0110 1000 0001  Func:010 imm:000000 rd:01101 op:01  c.li a3=0     4: 451d                  li  a0,7  0100 0101 0001 1101  Func:010 imm:000111 rd:01010 op:01  c.li: A0=7     6: 02000593            li  a1,32  0000 0010 0000 0000 0000 0101 1001 0011  Imm: 0000 0010 0000,rs1:00000,fun3:000,rd:01011,op:0010011  Addi(li):a1=0+32     a: 6805                  lui a6,0x1  0110 1000 0000 0101  Func:011 rd:10000 imm:000001 op:01  c.lui:a6=(1<<12)  000000000000000c <.L14>:     c: 2785                  addiw a5,a5,1  0010 0111 1000 0101  Fun3:001 imm:000001 rd:01111 op:01  c.addiw: a5+=1     e: 0016d69b            srliw a3,a3,0x1  0000 0000 0001 0110 1101 0110 1001 1011  Imm:0000 0000 0001 rs1:01101 fun3:101 rd:01101 op:0011011  Srliw: a3=a3>>1    12: 00f5d73b            srlw  a4,a1,a5  0000 0000 1111 0101 1101 0111 0011 1011  Fun7:0000000,rs2:01111,rs1:01011,fun3:101,rd:01110,op:0111011  Srlw: a4=(a1>>a5[4:0])    16: 8636                  mv  a2,a3  1000 0110 0011 0110  Fun3:100,imm=0,rs1:01100,rs2:01101,op:10  c.mv:a2=a3    18: 8b05                  andi  a4,a4,1  1000 1011 0000 0101  Fun3:100 uimm:000001 rd’:110 op:01  c.andi: a4=a4+1    1a: 00a78e63            beq a5,a0,36 <.L26>  0000 0000 1010 0111 1000 1110 0110 0011  Imm : 000000011100 rs2: 01010,rs1:01111 func3:000 op:1100011  Beq: if a5==a0 pc=.L26  000000000000001e <.L16>:    1e: d77d                  beqz  a4,c <.L14>    1101 0111 0111 1101  Func3:110 offset:10111111 rs1’=110 op:01  c.beqz: If a4==0 pc = .L14  20: 00c806bb            addw  a3,a6,a2  0000 0000 1100 1000 0000 0110 1011 1011  Imm:0000000 rs2:01100 rs1:10000 fun3:000 rd:01101 op:0111011  Addw: a3=a6+a2  24: 2785                  addiw a5,a5,1  0010 0111 1000 0101  Fun3:001 imm:000001 rd:01111 op:01  c.addiw: a5=a5+1    26: 0016d69b            srliw a3,a3,0x1  0000 0000 0001 0110 1101 0110 1001 1011  Imm:0000 0000 0001 rs1:01101 func3:101 rd:01101 op:0011011  Srliw: a3=a3>>1    2a: 00f5d73b            srlw  a4,a1,a5  0000 0000 1111 0101 1101 0111 0011 1011  Func7:0000000 rs2:01111 rs1:01011 fun3:101 rd:01110 op:0111011  Srlw:a4=(a1>>a5[4:0])    2e: 8636                  mv  a2,a3  1000 0110 0011 0110  Func3:100 imm:0 rs1:01100 rs2:01101 op:10  c.mv: a2=a3    30: 8b05                  andi  a4,a4,1  1000 1011 0000 0101  Func3:100 imm:000001 rd’=110 op:01  c.andi: a4=a4+1    32: fea796e3            bne a5,a0,1e <.L16>  1111 1110 1010 0111 1001 0110 1110 0011  Imm:111111101101 rs2:01010 rs1:01111 func3:001 op:1100011  Bne: if a5!=a0 pc=.L16  0000000000000036 <.L26>:    36: 4601                  li  a2,0  0100 0110 0000 0001  Func:010 imm:000000 rd:01100 op:01  c.li: A2=0    38: 4781                  li  a5,0  0100 0111 1000 0001  Func:010 imm:000000 rd:01111 op:01  c.li: A5=0    3a: 00100537            lui a0,0x100  0000 0000 0001 0000 0000 0101 0011 0111  Imm:0000 0000 0001 0000 0000 rd:01010 op:0110111  Lui: a0[31:12]=0x100    3e: 45bd                  li  a1,15  0100 0101 1011 1101  Func:010 imm:001111 rd:01011 op:01  c.li: A1=15  0000000000000040 <.L18>:    40: 00f6d73b            srlw  a4,a3,a5  0000 0000 1111 0110 1101 0111 0011 1011  Func7:0000000 rs2:01111 rs1:01101 fun3:101 rd:01110 op:0111011  Srlw:A4=a3[31:0]>>a5[4:0]    44: 8b05                  andi  a4,a4,1  1000 1011 0000 0101  Func3:100 uimm:000001 rd’:110 op:01  c.andi: a4=a4 & 1    46: 2785                  addiw a5,a5,1  0010 0111 1000 0101  Func3:001 imm:000001 rd:01111 op:01  c.addiw: a5=a5+1    48: c311                  beqz  a4,4c <.L17>  1100 0011 0001 0001  Func3:110 offset:00000100 rs1’110 op:01  c.beqz: If a4==0 pc=.L17    4a: 9e29                  addw  a2,a2,a0  1001 1110 0010 1001  Func6:100111 rd’100 imm:01 rs2’:010 op:01  c.addw: a2=(a2+a0)[31:0]  000000000000004c <.L17>:    4c: 0016561b            srliw a2,a2,0x1  0000 0000 0001 0110 0101 0110 0001 1011  Imm:0000 0000 0001 rs1:01100 fun3:101 rd:01100 op:0011011  Srliw:a2=a2>>1    50: feb798e3            bne a5,a1,40 <.L18>  1111 1110 1011 0111 1001 1000 1110 0011  Imm:111111111000 rs2:01011 rs1:01111 fun3:001 op:1100011  Bne: if a5!=a1 pc+= -16 (pc=.L18)    54: 00000537            lui a0,0x0  0000 0000 0000 0000 0000 0101 0011 0111  Imm :0000 0000 0000 rd:01010 op:0110111  Lui: a0[31:12]=0    58: 02000593            li  a1,32  0000 0010 0000 0000 0000 0101 1001 0011  Imm: 0000 0010 0000 rs1:00000 fun3:000 rd:01011 op:0010011  Li : a1=32 (addi)    5c: 00050513            mv  a0,a0  0000 0000 0000 0101 0000 0101 0001 0011  Imm:0000 0000 0000 rs1:01010 fun3:000 rd:01010 op:0010011  Mv: a0=a0 (addi)    60: 00000317            auipc t1,0x0  0000 0000 0000 0000 0000 0011 0001 0111  Imm: 0000 0000 0000 0000 0000 rd:00110 op:0010111  Auipc: t1 = pc+{imm+12`b0}    64: 00030067            jr  t1 # 60 <.L17+0x14>  0000 0000 0000 0011 0000 0000 0110 0111  Imm:0000 0000 0000 rs1:00110 fun3:000 rd:00000 op:1100111  Jr: pc=t1 (jalr) |