

考试科目名称 计算机组成原理

学号 \_\_\_\_\_ 姓名 \_\_\_\_\_ 成绩 \_\_\_\_\_

## 一、选择题（每小题 3 分，共 45 分）

1. 下列选项中，可以缩短程序执行时间的措施有（ D ）。
- ① 用主频更高的处理器替换原来的低主频的处理器  
② 优化程序的访问局部性，使得 cache 命中率更高  
③ 优化编译生成的代码使得程序执行总时钟周期数减少  
④ 选用转速更高的磁盘驱动器或将外存换成固态硬盘
- A. 仅①、②、③      B. 仅①、②、④      C. 仅①、③、④      D. 全部
2. 已知计算机 A 的时钟频率为 800MHz，假定某程序在计算机 A 上运行需要 12 秒钟。现在硬件设计人员想设计计算机 B，希望该程序在 B 上的运行时间能缩短为 8 秒钟，使用新技术后可使 B 的时钟频率大幅度提高，但在 B 上运行该程序所需的时钟周期数为在 A 上的 1.5 倍。那么，机器 B 的时钟频率至少应为多少才能达到所希望的要求？（ D ）
- A. 800MHz      B. 1.2GHz      C. 1.5GHz      D. 1.8GHz
3. 假定基准程序 A 在某计算机上运行需要 100 秒钟，其中 90 秒为 CPU 时间，其余为 I/O 时间。若 CPU 速度提高 50%，I/O 速度不变，则运行基准程序 A 的时间变为多少秒？（ A ）
- A. 70      B. 65      C. 60      D. 55
4. -1022 的 32 位补码用十六进制表示为（ A ）。
- A. FFFF FC02H      B. FFFF 03FEH      C. 0000 03FE H      D. 0000 FC02H
5. 假定变量 f 的数据类型为 float，f=-4.098e3，则变量 f 的机器数表示为（ B ）。
- A. 4580 1000H      B. C580 1000H      C. 4608 1000H      D. C608 1000H
6. 假定 T 表示一级门延迟，若一个与门和或门的延迟都为 T，一个异或门的延迟为 2T，则 8 位全先行进位加法器的关键路径延迟至少为（ A ）。
- A. 5T      B. 6T      C. 7T      D. 8T
7. 某 8 位计算机中，假定  $[x]_{\text{补}} = 5\text{FH}$ ， $[y]_{\text{补}} = \text{C8H}$ ，则通过补码加减运算器得到的  $x-y$  的值及其相应的进/借位标志 CF 分别是（ D ）。
- A. -23、0      B. -23、1      C. -105、0      D. -105、1
8. 以下选项中，在原码一位乘法器中包括的部件有（ D ）。
- ① ALU      ② 计数器      ③ 移位寄存器      ④ 控制逻辑      ⑤ 符号生成电路
- A. 仅①、②、③      B. 仅①、②、③、④      C. 仅①、③、④、⑤      D. 全部
9. 以下关于 Booth 补码一位乘法算法要点的描述中，错误的是（ C ）。
- A. 符号位和数值位一起参加运算，无需符号生成部件  
B. 通过循环执行“加/减”和“右移”操作得到乘积  
C. 移位时，将进位位、部分积和乘积部分一起算术右移  
D. 由乘数最低两位决定对部分积和被乘数进行何种运算
10. 对于两个 n 位无符号整数除法运算，以下关于不恢复余数算法要点的描述中，错误的是（ C ）。
- A. 起始时被除数在高位扩展 n 位 0，以将其扩展为 2n 位无符号整数  
B. 为判断中间余数的正/负，需在余数寄存器的最高位前增加一位符号位  
C. 至少需 n+1 次循环执行“加/减”和“左移”操作才能得到 n 位商  
D. 运算结果一定不会发生溢出，故无需通过得到最高位商来判断溢出

11. 若两个 float 型变量 x 和 y 的机器数分别表示为 x: 82C0 0000H, y: 521A 0000H, 则在计算 x+y 时, 第一步对阶操作的结果 $[\Delta E]_{补}$ 为 ( B )。
- A. 1001 1111                      B. 0110 0001                      C. 1010 1001                      D. 0110 0000
12. 某计算机按字节编址, 采用小端方式存储信息。某指令的一个操作数为 16 位, 该操作数采用基址寻址方式, 指令中形式地址 (用补码表示) 为 FF00H, 当前基址寄存器的内容为 C000 0000H, 则该操作数的 MSB 存放的地址是 ( B )。
- A. BFFF FF00H                      B. BFFF FF01H                      C. C000 FF00H                      D. C000 FF01H
13. 通常将在部件之间进行数据传送的指令称为传送指令。以下有关各类传送指令功能的叙述中, 错误的是 ( C )。
- A. 出/入栈指令 (push/pop) 完成 CPU 和栈顶之间的数据传送  
B. 输入/出指令 (in/out) 完成 CPU 和 I/O 端口之间的数据传送  
C. 寄存器传送指令 (move) 完成 CPU 和寄存器之间的数据传送  
D. 访存指令 (load/store) 完成 CPU 和存储单元之间的数据传送
14. 执行完当前指令后, PC 中存放的是后继指令的地址, 通常 PC 的位数和 ( C ) 的位数相同。
- A. 指令寄存器                      B. 程序状态字寄存器                      C. 主存地址寄存器                      D. 指令译码器
15. 以下给出的四种指令类型中, 执行时间最长的指令类型是 ( C )。
- A. RR 型                                  B. RS 型                                  C. SS 型                                  D. RI 型

## 二、综合题 (共 55 分, 答案写在答题纸上)

1. (15 分) 分别给出 C 语句 “int x=-4091;” 和 “int x=4091;” 对应的 RISC-V 机器指令、汇编指令和对应的注释。

参考答案:

-4091 的机器数为: 1111 1111 1111 1111 1111 0000 0000 0101, C 语句 “int x=-4091;” 对应的 RISC-V 机器指令和汇编指令为:

```
1111 1111 1111 1111 1111 00101 0110111  lui x5, 1048575  #R[x5]←FFFF F000H
0000 0000 0101 00101 000 00101 0010011  addi x5, x5, 5    #R[x5]←R[x5]+SEXT[05H]
(FFFF F2B7, 0052 8293)
```

4091 的机器数为: 0000 0000 0000 0000 0000 1111 1111 1011, 4091=4096-5, C 语句 “int x=4091;” 对应的 RISC-V 机器指令和汇编指令为:

```
0000 0000 0000 0000 0001 00101 0110111  lui x5, 1          #R[x5]←0000 1000H
1111 1111 1011 00101 000 00101 0010011  addi x5, x5, -5    #R[x5]←R[x5]+SEXT[FFBH]
(0000 12B7, FFB2 8293)
```

2. (40 分) 下图是某程序的可重定位目标文件在 RISC-V 系统中的反汇编结果, 回答下列问题: (1) 这是 32 位还是 64 位系统? 为什么? (2) main() 中有多少条指令? 其机器代码占多少字节? 哪些汇编指令是伪指令? (3) 栈帧中包含了哪些信息? 各信息对应起始位置在何处? (4) 装入 64 位常数和 32 位常数的处理有何不同? (5) 程序中哪些信息定义在 .rodata 节? 这些信息在何处? 要求给出计算过程。(6) 哪条指令用于调用 printf() 函数? 该指令中 imm 字段内容是什么? 它是如何生成的?

```

410 int main() {
411 5fa: 1101          addi sp,sp,-32
412 5fc: ec06          sd ra,24(sp)
413 5fe: e822          sd s0,16(sp)
414 600: 1000          addi s0,sp,32
415 uint64_t u64 = 0x1234567890abcdefull;
416 602: 00000797      auipc a5,0x0
417 606: 0a678793      addi a5,a5,166 # 6a8 <- libc_csu_fini+0x12>
418 60a: 639c          ld a5,0(a5)
419 60c: fef43423      sd a5,-24(s0)
420 uint32_t u32 = 0x12345678;
421 610: 123457b7      lui a5,0x12345
422 614: 6787879b      addiw a5,a5,1656
423 618: fef42223      sw a5,-28(s0)
424 printf("%llx %lx\n", u64, u32);
425 61c: fe442783      lw a5,-28(s0)
426 620: 863e          mv a2,a5
427 622: fe843583      ld a1,-24(s0)
428 626: 00000517      auipc a0,0x0
429 62a: 07250513      addi a0,a0,114 # 698 <- libc_csu_fini+0x2>
430 62e: ef3ff0ef      jal ra,520 <printf@plt>
431 return 0;
432 632: 4781          li a5,0
433 }
434 634: 853e          mv a0,a5
435 636: 60e2          ld ra,24(sp)
436 638: 6442          ld s0,16(sp)
437 63a: 6105          addi sp,sp,32
438 63c: 8082          ret

```

### 参考答案:

- (1) 这是 64 位系统。可以看出机器代码中用到了+RV64I 中的指令，如 sd、ld。(4 分)
- (2) main ()中有 23 条指令。其机器代码占 68 字节。mv、li 和 ret 是伪指令。(6 分)
- (3) 栈帧中保存了寄存器 ra 和 s0 的内容，分别位于栈帧顶部(sp)+24、+16 的位置；栈帧中还包含局部变量 u64 和 u32，分别位于栈帧顶部(sp)+8、+4 的位置。(8 分)
- (4) 装入 64 位常数的处理过程：编译器将 64 位常数定义在.rodata 节，根据常数与当前指令之间的位移量，通过 addi 指令计算出常数所在的地址，然后通过 ld 指令从该地址中装入 64 位常数（这是一段 PIC 代码）；而对于 32 位常数的装入，则只要通过 lui 指令和 addi 指令直接给出常数即可。(6 分)
- (5) 常数 0x1234567890abcdef 和 printf()函数中的格式串信息定义在.rodata 节中。分别位于对应可重定位文件中相对位置为 0x602+0x0a6=0x6a8 和 0x626+0x072=0x698 的位置。(8 分)
- (6) 指令 jal ra, 520 <printf@plt>用于调用 printf()函数。该指令前 20 位为 imm 字段，其内容为 1110 1111 0011 1111 1111。生成过程如下：

根据目标地址=PC+SEXT[imm[20:1]<<1]]，可知：0x520=0x62e+ SEXT[imm[20:1]<<1]]。0x520-0x62e=1111 1111 1110 1111 0010，因此，imm[20:1]=1 1111 1111 1110 1111 001，按照 imm 字段的编码规则 imm[20|10:1|11|19:12]，可以得到 imm 字段内容为：1 110 1111 001 1 1111 1111。(8 分)