

# 中山大学本科生期中考试

## 考试科目：《计算机组成原理》（A 卷）

学年学期：2023 学年第 2 学期

姓 名：\_\_\_\_\_

开课单位：人工智能学院

学 号：\_\_\_\_\_

考试方式：闭卷

年 级：\_\_\_\_\_

考试时长：90 分钟

院 系：\_\_\_\_\_

### 警示

《中山大学授予学士学位工作细则》：“考试作弊者，不授予学士学位。”

-----以下为试题区域，共四道大题，总分 100 分，考生请在答题纸上作答-----

### 一、单项选择题（共15小题，每小题2分，共30分）

1. 下列指令的功能是（ ）。

xori x3, x3, -1

- A. 取负                      B. 求原码                      C. 求反码                      D. 求补码

2. 下列指令集体系结构(ISA)中不属于RISC的是（ ）。

- A. x86                                      B. RISC-V  
C. ARM                                      D. LoongArch（龙芯）

3. 对于32位补码加法运算，数值0x80000000与（ ）相加会产生溢出。

- A. 1                                      B. -1                                      C. 2                                      D. 都不会

4. 关于移位操作，下列（ ）是错误的。

- A. 无论是逻辑还是算术左移，只要不溢出，都会使数据的绝对值放大  
B. 无论是逻辑还是算术右移，都会使数据的绝对值缩小  
C. 无论是逻辑还是算术左移，都是在低位补0  
D. 无论是逻辑还是算术右移，都是在高位补0

5. IEEE 754单精度浮点格式表示的数中，最小的规格化正数是（ ）。

- A.  $1.0 \times 2^{-126}$                       B.  $1.0 \times 2^{-127}$                       C.  $1.0 \times 2^{-128}$                       D.  $1.0 \times 2^{-149}$

6. IEEE 754双精度浮点格式表示的数中，最小的非规格化正数是（ ）。

- A.  $1.0 \times 2^{-149}$       B.  $1.0 \times 2^{-1022}$       C.  $1.0 \times 2^{-1074}$       D. 都不正确
7. 如果一次加法需要4个时间单位，若对64位整数使用快速乘法器进行乘法，所需的时间是（      ）个时间单位。
- A. 6      B. 24      C. 64      D. 252
8. 求 $65 \times 66$ 的乘积，可采用移位和加法来替代实际的乘法。移位和加法的次数分别是（      ）。
- A. 1次移位，0次加法      B. 1次移位，1次加法  
C. 2次移位，1次加法      D. 正确答案不止一个
9. 关于IEEE 754非规格化数的说法正确的是（      ）。
- A. 不可表示数值0  
B. 不可表示无穷  
C. 移码的偏移量与规格化数相同  
D. 表示绝对值比规格化数更小的值
10. 下列RISC-V指令中，（      ）不能设置寄存器x5为0。
- A. `addi x5, x0, 0`  
B. `and x5, x5, x0`  
C. `or x5, x5, x0`  
D. `xor x5, x5, x5`
11. 在C语言中声明的整数数组A: `int A[100]`。若访问的数组元素A[i]（下标为i），下列指令（      ）可完成下标越界的检查。假定x11值为100，x20值为i。
- A. `bge x20, x11, ERROR`  
B. `blt x20, x11, ERROR`  
C. `bgeu x20, x11, ERROR`  
D. `bltu x20, x11, ERROR`
12. 将8位二进制补码进行符号扩展成16位补码，下列（      ）是错误的。
- A. 0000 0000扩展成0000 0000 0000 0000  
B. 1111 1111扩展成1111 1111 1111 1111  
C. 1000 1000扩展成1000 1000 1000 1000  
D. 1000 0000扩展成1111 1111 1000 0000
13. 下列RISC-V指令错误的是（      ）。

- A. `add x2, x2, x2`
- B. `add x2, x2, 2`
- C. `ld x2, 0(x3)`
- D. `sd x2, 0(x3)`
14. 在指令集体系结构是x86或RISC-V的机器上，整型变量x（占内存4个字节）的值为0x12345678。如果x在内存中的地址是0x40000000，则内存地址0x40000001中的值是（ ）。
- A. 0x12                      B. 0x34                      C. 0x56                      D. 0x78
15. RISC-V汇编指令“`add x19, x20, x21`”使用的是R格式（如下图），则字段rd的位模式是（ ）。

<b>funct7</b>	<b>rs2</b>	<b>rs1</b>	<b>funct3</b>	<b>rd</b>	<b>opcode</b>
7 bits	5 bits	5 bits	3 bits	5 bits	7 bits

- A. 10011                      B. 10100                      C. 10101                      D. 10110
- 二、填空题（共8小题，每空1.5分，共30分）**
1. 冯·诺伊曼计算机由五个部件组成，它们是：输入设备、输出设备、控制器、（ ）和（ ）。
2. 8位的二进制数“11000110”，其十六进制表示为（ ）、八进制表示为（ ）。其表示的十进制数原码是（ ）、反码是（ ）、补码是（ ）。
3. 字节(Byte)是（ ）位，字(word)通常是（ ）位，双字(doubleword)通常是（ ）位。
4. 16位补码可表示的最大正整数是（ ），最小负整数是（ ）。
5. 将C源程序转换为计算机上可运行的程序通常分为四个步骤：C程序→(编译器)→汇编语言程序→( )→目标模块→( )→可执行代码→(加载器)→存储器。
6. IEEE754双精度浮点格式中阶码的位数是（ ）位、阶码移码表示的偏移量是（ ）、有效数的位数是（ ）位。
7. 二进制数1.011若要求只保留三位有效数，且向偶数舍入，则结果是（ ）。
8. 在RISC-V中执行过程调用指令“`jal x1, Proc`”时，返回指令地址置于寄存器（ ）中，其值为PC+（ ）。

三、程序设计题（共3小题，每小题4分，共12分）

1. 将以下RISC-V指令转换成C代码。假设变量i、j放在寄存器x22和x24，数组A的基地址放在寄存器x25中。假设数组A的元素每个大小为8字节：

```
Loop:
    slli x10, x22, 3
    add x10, x10, x25
    ld x9, 0(x10)
    bne x9, x24, Exit
    addi x22, x22, 1
    beq x0, x0, Loop
Exit:
    ...
```

2. 将以下C代码转换成RISC-V指令。假设变量f、g、h、i和j分别分配给寄存器x5、x6、x7、x28和x29。假设数组A和数组B的基地址分别在寄存器x10和x11中。假设数组A和数组B的元素每个大小为8字节：

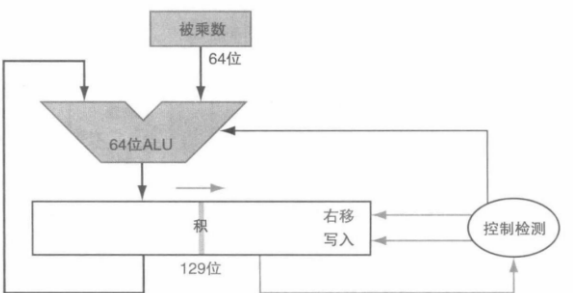
```
B[j]=A[i]+10
```

3. 用RISC-V汇编语言实现以下C代码。提示：记住栈指针必须保持16位对齐。

```
long int sum(long int n){
    if(n==1) return 1;
    else return n+sum(n-1);
}
```

四、综合题（共4小题，每小题7分，共28分）

1. 求-8.1的单精度浮点表示的位模式，并给出求解过程。
2. 给出下列64位无符号数改良版乘法器的工作流程（包括初始化）。



3. 假定某编译器对某段高级语言程序编译生成了两种不同的指令序列S1和S2，在时钟频率为500MHZ的机器M上运行，目标指令序列中用到的指令类型有A、B、C和D四类。4类指令在M上的CPI和两个指令序列所用的各类指令条数如下表所示。

指令指标	A类指令	B类指令	C类指令	D类指令
CPI	1	2	3	4
S1指令数	5	2	2	1
S2指令数	1	1	1	5

- 1) S1和S2各有多少条指令？
  - 2) S1和S2的时钟周期数各是多少？
  - 3) S1和S2的平均CPI各是多少？
4. 假定一台计算机上运算一个程序需要200秒，其中60秒用于执行浮点指令，70秒用于执行L/S指令，30秒用于执行分支指令，其它时间用于执行整型操作指令。
- 1) 如果浮点操作的时间减少20%，总时间将减少多少？
  - 2) 如果只减少整型指令时间，总时间能否减少20%？