(10 points) The bits have no inherent meaning. Given the pattern:

0xC3500037

(0b 1100 0011 0101 0000 0000 0000 0011 0111)

what does it represent, assuming that it is:

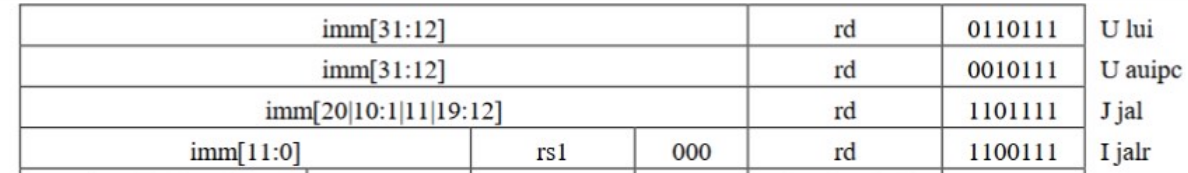
1. a two's complement? （answer it in base 16）（2 points）

2. an unsigned integer? （answer it in base 10） （2 points）

3. a biased notation(移码) （answer it in base 16）（2 points）

4. A single precision floating-point number? (In order to make your calculation easier, you can only keep one decimal place) （2 points）

5. Risk-V instruction （2 points）

hint: 

答案：

1. 0xBCAFFFC9

2. 3276800055

3. 第一位从1改成0 可以得到

0b 0100 0011 0101 0000 0000 0000 0011 0111

0x43500037

4. -208.1

符号位 1（**负**数）

阶码E 100 0011 0 十进制为134 - 127 = **7**

有效数字M 101 0000 0000 0000 0011 0111

(-1) \* (1 + 1/2 + 1/8+…+) \* 2^7 =-208.00083923339844≈-208.1

5. lui x0, 800000 或者lui x0, c3500

1100 0011 0101 0000 0000 0000 0011 0111

Opcode: 011 0111 可以看出为lui

立即数为0000 0 可以看出为x0

1100 0011 0101 0000 0000转换为10进制为800000

（10 points）The following is a piece of code in c language, please translate it into 32-bit RISC-V instructions

int acc(int n){

    if (n == 1)

        return 1;

    else

        return n+acc(n - 1);

}

int main(){

    int ret = acc(3);

}

答案

可以先转换为

int acc(int a0) {

if (a0 == 0)

return 0;

else

return a0 + acc(a0 - 1);

}

int main() {

int a0 = 3;

int result = acc(a0);

return 0;

}

会好写一些，答案如下：

the execution will start from the "main" label

main:

    addi a0, zero, 3

    jal ra, acc

    jalr zero, 0(ra)

acc:

    addi sp, sp, -16

    sw ra, 12(sp)

    sw a0, 8(sp)

    addi a0, a0, -1

    beq a0, zero, .L2

    jal ra, acc

    lw a5, 8(sp)

    add a0, a0, a5

.L2:

    lw ra, 12(sp)

    addi sp, sp, 16

    jalr zero, 0(ra)

ACC实现正确给 5分

分成了2个Loop 给3分（如果功能实现正确，不分成2个也行）

功能基本正确，有点小错误，给9分

完全正确 10分

如果没有采用递归的思想来实现（因为其实这个C语言只是3个数相加），得8分