

# Task 2

Q1. Count the number of executed instructions in hash and hash\_rv32z. Which one executes more instructions?

hash\_rv32z executes more instructions than hash does.

number of executed instructions	
hash	1627
hash_rv32z	2979

Q2. We designed the custom rv32z in hopes it would accelerate the performance of hash computation. Can you relate your answer in Q1 to justify why rv32z-based hash computation would be faster than rv32im-based hash computation?

By comparing hash.s and hash\_rv32z.s, we can find out that computing hash value in hash\_rv32z is opreated in a single instruction. In contast, hash needs to couples of instructions to calculate updated hash value. Basically, more instructions would increases execution time. Because, they need to fetch more instructions and decode them as well. Therefore, rv32z based hash computation would be faster than rv32im based hash computation.

```
<!-- hash_rv32z.s -->
mv a5, a4
.word 0b1111101 | (0 << (7)) | (0x7 << (7+5)) | (15 << (7+5+3)) | (0 << (7+5+3+5)) | (0 << (7+5+3+5+5))

<!-- hash.s -->
.L3:
    lw a5,-36(s0)
    lbu a5,0(a5)
    mv a4,a5
    lw a5,-20(s0)
    xor a5,a5,a4
    sw a5,-20(s0)
    lw a4,-20(s0)
    li a5,1540485120
    addi a5,a5,-1643
    mul a5,a4,a5
    sw a5,-20(s0)
    lw a5,-20(s0)
    srli a5,a5,15
    lw a4,-20(s0)
    xor a5,a4,a5
    sw a5,-20(s0)
    lw a5,-36(s0)
    addi a5,a5,1
    sw a5,-36(s0)
```