|  |  |  |
| --- | --- | --- |
| PC | Instruction | Result |
| 00000000 | addi $8 $0 0  001000 00000 01000 0000000000000000 | $8 = 0 |
| 00000004 | addi $9 $0 3  001000 00000 01001 0000000000000011 | $9 = 3 |
| 00000008 | addi $8 $8 1  001000 01000 01000 0000000000000001 | Loop1: $8 = 1  Loop2: $8 = 2  Loop3: $8 = 3 |
| 0000000C | bne $8 $9 -2  000101 01000 01001 1111111111111110 | Loop1: branch PC=00000008 (no pred)  ENTRIES[3] (0x0C) = 8  Loop2: branch PC=00000008 (pred)  Loop3: branch PC=00000008 (pred)  Trying to jump from C to 8 |
| 00000010 | addi $5 $0 0  001000 00000 00101 0000000000000010 | $5 = 2 |
| 00000014 | addi $6 $0 7  001000 00000 00110 0000000000000111 | $6 = 7 |
| 00000018 | addi $5 $5 1  001000 00101 00101 0000000000000001 | Loop1: $5 = 1  Loop2: $5 = 2  Loop3: $5 = 3  ….. |
| 0000001C | bne $5 $6 -2  000101 00110 00101 1111111111111110 | Jump back to 18 until $5 == $6 |

$8 = 0

addi $8 $0 0

001000 00000 01000 0000000000000000

$9 = 10

addi $9 $0 10

001000 00000 01001 0000000000001010

$8 = 1

addi $8 $8 1

001000 01000 01000 0000000000000001

branch to previous if $8 != $9

bne $8 $9 -1

000101 01000 01001 1111111111111111

addi $8 $0 0

001000 00000 01000 0000000000000000

addi $8 $8 1

001000 01000 01000 0000000000000001

bne $8 $9 -1

000101 01000 01001 1111111111111111