REDUX-V: Algorítmo Customizado

Nico I. G. Ramos GRR20210574

Setembro 2024

1 Assembly

```
; int s = 0;
; for (int i = 0; i < 8; i++) {
;   s += 7 / (1 << i);
; }

_setup:
0x00  xor r0, r0          ; r0 = 0
0x01  addi 1              ; r0 = 0000 0001
0x02  mov r2, r0          ; r2 = r0
                          ; r2 := 1
0x03  fpe 5, 0,
0x04      3, 3            ; setup fixed point

_loop:
0x05  xor r0, r0          ; r0 = 0
0x06  addi 1              ; r0 = 0000 0001
0x07  srr r0, r0          ; r0 = 0000 0010
0x08  srr r0, r0          ; r0 = 0010 0000
                          ; r0 := _loop_end
0x09  brzr r2, r0         ; PC = r2 == 0 ? _loop_end : PC + 1

0x0A  xor r0, r0          ; r0 = 0
0x0B  addi 15             ; r0 = 1111 1111
0x0C  mov r1, r0          ; r1 = 1111 1111
0x0D  ji 2                ; PC = PC + 2
0x0E  ji -8               ; PC = PC - 8
0x0F  xor r0, r0          ; r0 = 0
0x10  addi 4              ; r0 = 4

0x11  slr r1, r0          ; r1 := 1111.0000
0x12  slr r1, r2          ; r1 = r1 >> r2
0x13  xor r0, r0          ; r0 = 0
0x14  addi 8              ; r0 = 1111 1000
0x15  ji 2                ; PC = PC + 2
0x16  ji -8               ; PC = PC - 8
0x17  or r1, r0           ; r1 = 1111.1000
0x18  fpe 0,
0x19        add, r3, r1    ; r3 += r1
0x1A   xor r0, r0          ; r0 = 0
0x1B   addi 1              ; r0 = 1
0x1C   srr r2, r1          ; r2 = r2 << 1
0x1D   ji -7               ; PC = _loop
0x1E   0
0x1F   0

_loop_end:
0x20
```

2 Código de Máquina

```
10110000
01000001
01111000
01010101
00011011
10110000
01000001
11110000
11110000
00001000
10110000
01001111
01110100
00010010
00011000
10110000
01000100
11100100
11100110
10110000
01001000
00010010
00011000
10100100
01010000
11001101
10110000
01000001
11111001
00011001
00000000
00000000
```