```
main:
    addi a0, zero, 1
                        // a0 equivale ao valor 1
    add a1, a0, a0
                        // a1 equivale ao valor 2 2 pois a0 anteriormente validado com
valor 1
    addi a2, zero, 5
                       // a2 equivale ao valor 5
loop:
    beg a0, a2, fim
                        //o looping começa até que o valor de a0(1) seja igual ao valor de
a2(5) e pula pro fim
    slli a1, a1, 1
                     // a1(2) será multiplicado por 2 até o fim do looping
    addi a0, a0, 1
                       // incrementa += 1 em a0(até chegar a 5)
    j loop
                    //volta pro loop até a0 ser 5
fim:
    nop
                    //final do programa
a) Operação Realizada:
O código realiza um loop que multiplica o valor de a1 por 2 (slli a1, a1, 1) até que a0(1) seja
igual a a2(5).
b) Conteúdo dos Registradores ao Final:
Ao final da execução do programa, os valores dos registradores são:
a0: 5 (valor alcançado no loop)
a1: 32(dec) (o valor é multiplicado por 2 quatro vezes, ele serve como um Shift Left "<<")
/20(hex)
```

.text

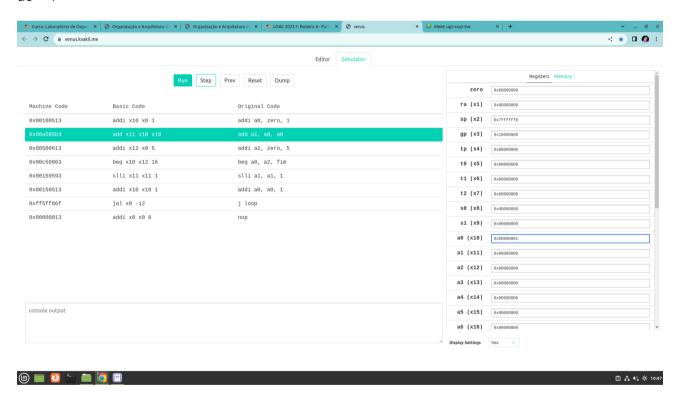
a2: 5 (valor inicializado como 5)

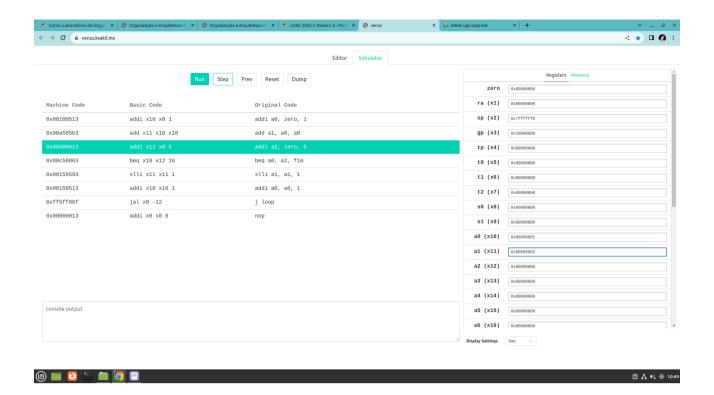
c) Valores de loop e fim:

2147483632 e 268435456 (em decimal)

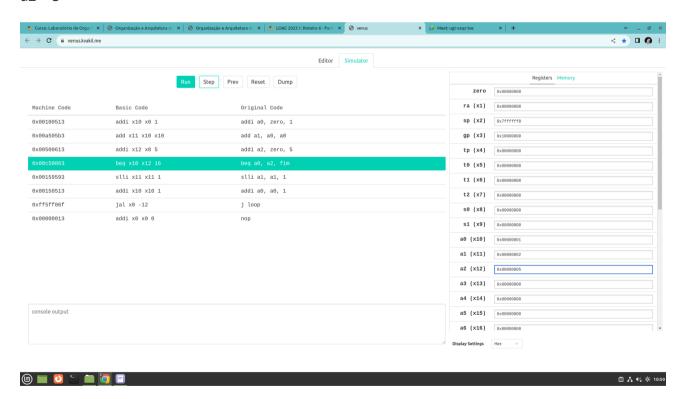
A seguir os passos no simulador:

a0 = 1



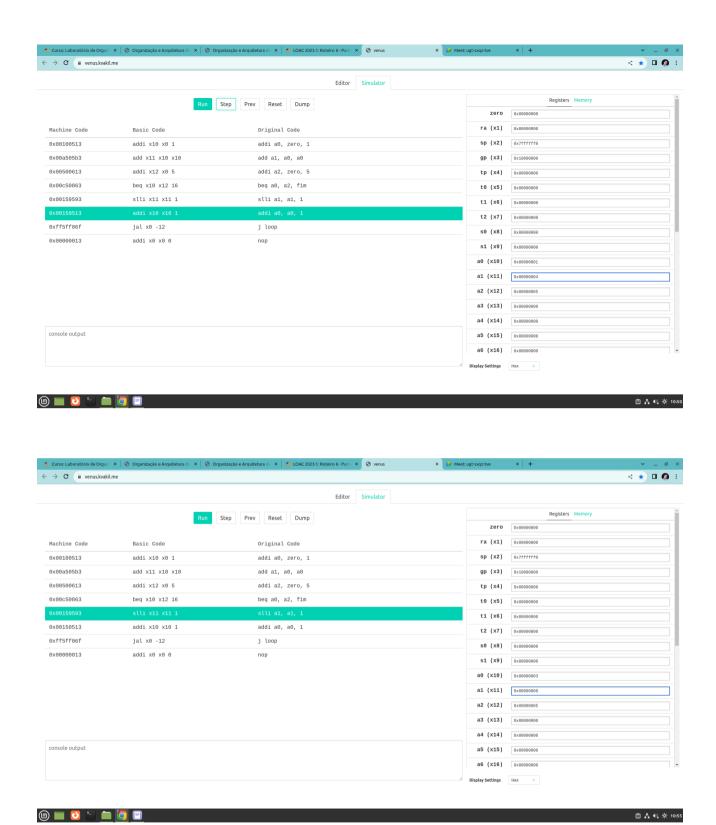


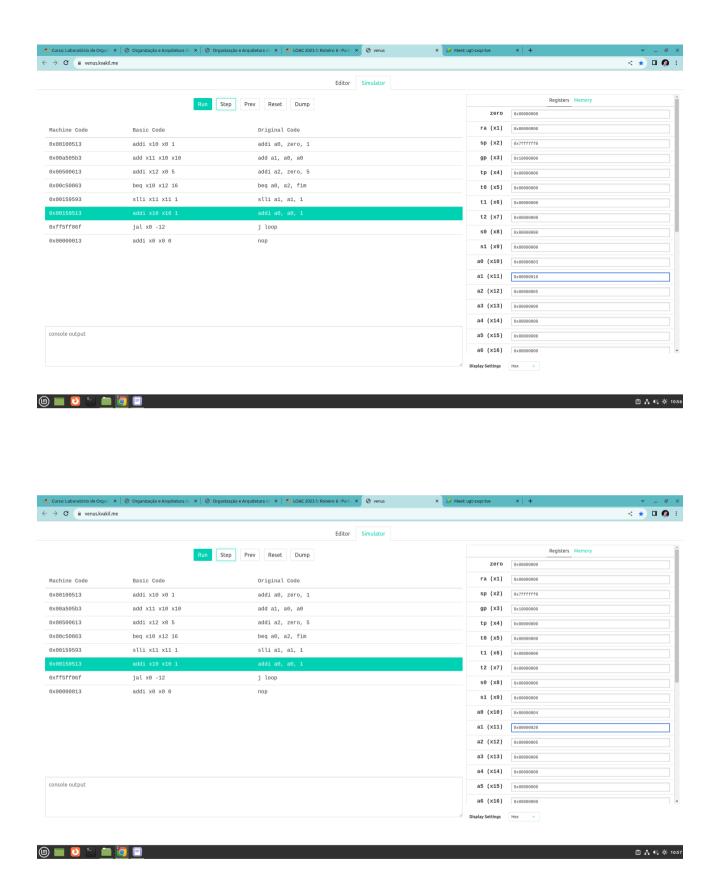
a2 = 5



Começando o looping:

a1 multiplicado por 2 e a0 sendo incrementado em 1





até que a0 chega ao valor 5 e pula pro fim, valores finais em decimal:

