

==== Instruções do Programa de Soma de Vetores ====

movl 1 - 0111 0001 - 71 ;; Carrega 1 em R[0]

nop

nop

nop

movl 12 - 0111 1100 - 7c ;; Carrega 12 em R[0]

nop

add r[1] r[0] ;; r[1] = 1

add r[2] r[0] ;; r[2] = 1

loop:

st r[3] M[r[3]] ;; Armazena r[3] na posição M[r[3]]

nop

add r[3] r[1] ;; Registrador de endereço é incrementado

sub r[0] r[1] ;; Registrador contador é decrementado

nop

brzi fora_loop ;; Se r[0] == 0, fim do loop

nop

nop

nop

ji loop ;; Salta para o loop

nop

nop -

fora_loop:

movl 12 ;; r[0] = 12

nop -

sub r[2] r[2] ;; r[2] = 0

sub r[3] r[3] ;; r[3] = 0

movl 8 ;; r[0] = 8

nop

add r[2] r[0] ;; r[2] = r[2] + 12 ;; Guarda o endereço do segundo vetor

add r[3] r[0] ;; r[3] = r[3] + 12

movl 12 ;; Salva o contador

nop

add r[3] r[0] ;; r[3] = r[3] + 8, contem o primeiro valor

nop

loop:

st r[3] M[r[2]] ;; Armazena r[3] em M[r[2]]

nop

sub r[0] r[1] ;; Decrementa o contador

add r[3] r[1] ;; Incrementa o valor a ser guardado

nop

brzi fora_loop ;; Se r[0] = 0, fim do loop

add r[2] r[1] ;; Incrementa o ponteiro do endereço

nop

nop

ji loop ;; Salta para o loop

nop

nop

fora_loop:

movl 8 ;; r[0] = 8

nop

sub r[2] r[2] ;; r[2] = 0

sub r[3] r[3] ;; r[3] = 0

movh 16 ;; r[0] = 24

nop

nop

nop

nop

nop

add r[2] r[0] ;; r[2] guarda o endereço de R

and r[0] r[3] ;; r[0] = 0

movl 12 ;; r[0] = 12

nop

nop

nop

loop:

st r[3] M[r[2]] ;; Armazena r[3] em M[r[2]]

nop

sub r[0] r[1] ;; Decrementa o contador

add r[2] r[1] ;; Incrementa o endereço

nop

brzi fora_loop ;; Se r[0] = 0, fim do loop

nop

nop

nop

ji loop ;; Salta para o loop

nop

nop

```

movl 12 ;; r[0] = 12
nop
sub r[3] r[3] ;; r[3] = 0
nop

nop
nop
add r[3] r[0] ;; r[3] = 12
sub r[2] r[0] ;; r[2] = r[2] - 12 = 24

/* Loop Unrolling (Repete 12x) */
ld r[0] M[r[3]] ;; Carrega em r[0] o elemento do segundo vetor
nop
add r[3] r[1] ;; Incrementa o endereço do segundo vetor
nop

st r[0] M[r[2]] ;; Salva em R o elemento carregado
nop
add r[2] r[1] ;; Incrementa o endereço de R
nop
/*****/

movh 0 ;; r[0] = 15
nop
nop
nop

movl 12 ;; r[0] = 12
nop
sub r[2] r[1] ;; r[2] = 35
sub r[3] r[3] ;; r[3] = 0

loop:
ld r[3] M[r[2]] ;; Carrega o ultimo elemento do segundo vetor.
nop
nop
sub r[0] r[1] ;; Decrementa o endereço.

ld r[0] M[r[0]] ;; Carrega o ultimo elemento do primeiro vetor.
nop
nop
nop

nop
nop
add r[3] r[0] ;; Soma os elementos
nop

```

```
st r[3] M[r[2]] ;; Armazena em R
nop
sub r[2] r[1] ;; Decrementa o endereço.
nop
```

```
nop
brzi fora_loop ;; Se r[0] = 0, fim do loop.
nop
nop
```

```
nop
ji loop ;; Salta para o loop.
nop
nop
```

```
nop
ji 0000 ;; Finaliza o programa.
nop
nop
```