**Módulo 6** :: **AC** :: **LEI**

**Y86 Sequencial**

Para cada instrução do código *assembly* Y86 fornecido, assinalada com **\*\*\***, apresente:

**(a)** A codificação da instrução, a sua disposição na memória, indicando o endereço onde está armazenada (lembre-se de que se trata de uma arquitetura *little endian*).

|  |  |  |  |
| --- | --- | --- | --- |
| **Endereço** | **Instrução** | **Codificação** | **Memória** |
|  | .pos 0 |  |  |
|  | init: |  |  |
|  | irmovl stack, %esp **# \*\*\*** |  |  |
|  | irmovl stack, %ebp |  |  |
|  | jmp main **# \*\*\*** |  |  |
|  |  |  |  |
|  | main: |  |  |
|  | irmovl $4, %eax |  |  |
|  | pushl %eax |  |  |
|  | irmovl data, %ebx **# \*\*\*** |  |  |
|  | mrmovl $4(%ebx), %eax |  |  |
|  | pushl %eax **# \*\*\*** |  |  |
|  | call soma **# \*\*\*** |  |  |
|  | halt |  |  |
|  |  |  |  |
|  | soma: |  |  |
|  | pushl %ebp |  |  |
|  | rrmovl %esp, %ebp **# \*\*\*** |  |  |
|  | mrmovl $8(%ebp), %eax |  |  |
|  | mrmovl $12(%ebp), %ebx **# \*\*\*** |  |  |
|  | addl %ebx, %eax **# \*\*\*** |  |  |
|  | rrmovl %ebp, %esp |  |  |
|  | popl %ebp **# \*\*\*** |  |  |
|  | ret **# \*\*\*** |  |  |
|  |  |  |  |
|  | .pos 0x100 |  |  |
|  | data: .long 10 |  |  |
|  | .long 24 |  |  |
|  |  |  |  |
|  | .pos 0x200 |  |  |
|  | stack: # Inicio da pilha |  |  |