

### Questão 1:

```
section .data
    msg1 db "Informe seu nome: "
    tam_msg1 equ $ - msg1

    msg2 db "Informe seu peso: "
    tam_msg2 equ $ - msg2

    msg3 db " "

section .bss
    nome resb 15
    peso resb 5
    tn resb 15

section .text

global _start

_start:
    mov eax,4
    mov ebx,1
    mov ecx,msg1
    mov edx,tam_msg1
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,nome
    mov edx,15
    int 0x80

    mov [tn], eax

    mov eax,4
    mov ebx,1
    mov ecx,msg2
    mov edx,tam_msg2
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,peso
    mov edx,5
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,nome
    mov edx,[tn]
    sub edx,1
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg3
    mov edx,2
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,peso
    mov edx,5
    int 0x80

    mov eax,1
    int 0x80
```

## Questão 2 - Item 1:

```
section .data
    msg1 db "Primeiro número: "
    tam_msg1 equ $ - msg1

    msg2 db "Segundo número: "
    tam_msg2 equ $ - msg2

    msg3 db "Soma: "
    tam_msg3 equ $ - msg3

section .bss
    x resb 1
    y resb 1
    r resb 1

section .text

global _start

_start:

    mov eax,4
    mov ebx,1
    mov ecx,msg1
    mov edx,tam_msg1
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,x
    mov edx,2
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg2
    mov edx,tam_msg2
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,y
    mov edx,2
    int 0x80

    mov al, [x]
    mov bl, [y]
    sub al,'0'
    sub bl,'0'
    add al, bl
    add al,'0'
    mov [r], al
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg3
    mov edx,tam_msg3
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,r
    mov edx,2
    int 0x80

    mov eax,1
    int 0x80
```

## Questão 2 - Item 2:

```
section .data
    msg1 db "Primeiro número: "
    tam_msg1 equ $ - msg1

    msg2 db "Segundo número: "
    tam_msg2 equ $ - msg2

    msg3 db "Subtração: "
    tam_msg3 equ $ - msg3

section .bss
    x resb 1
    y resb 1
    r resb 1

section .text

global _start

_start:

    mov eax,4
    mov ebx,1
    mov ecx,msg1
    mov edx,tam_msg1
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,x
    mov edx,2
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg2
    mov edx,tam_msg2
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,y
    mov edx,2
    int 0x80

    mov al,[y]
    mov bl,[x]
    sub al,'0'
    sub bl,'0'
    sub al,bl
    add al,'0'
    mov [r],al
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg3
    mov edx,tam_msg3
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,r
    mov edx,2
    int 0x80

    mov eax,1
    int 0x80
```

### Questão 2 - Item 3:

```
section .data
    msg1 db "Primeiro número: "
    tam_msg1 equ $ - msg1

    msg2 db "Segundo número: "
    tam_msg2 equ $ - msg2

    msg3 db "Terceiro número: "
    tam_msg3 equ $ - msg3

    msg4 db "Resultado: "
    tam_msg4 equ $ - msg4

section .bss
    x resb 1
    y resb 1
    z resb 1
    r resb 1

section .text
global _start

_start:
    mov eax,4
    mov ebx,1
    mov ecx,msg1
    mov edx,tam_msg1
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,x
    mov edx,2
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg2
    mov edx,tam_msg2
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,y
    mov edx,2
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg3
    mov edx,tam_msg3
    int 0x80

    mov eax,3
    mov ebx,0
    mov ecx,z
    mov edx,2
    int 0x80

    mov al, [x]
    mov bl, [y]
    mov cl, [z]
    sub al,'0'
    sub bl,'0'
    sub cl,'0'
    add al,bl
    sub al,cl
    add al,'0'
    mov [r],al
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,msg4
    mov edx,tam_msg4
    int 0x80

    mov eax,4
    mov ebx,1
    mov ecx,r
    mov edx,2
    int 0x80

    mov eax,1
    int 0x80
```