












Name	Name	Last commit date
 ..		
 1.s	feat(ac): ex3 aula	42 minutes ago
 2.s	feat(ac): ex3 aula	42 minutes ago
 3-2.s	chore(ac): ex3	now
 3.s	feat(ac): ex3 aula	42 minutes ago
 4.s	chore(ac): ex3	now
 5.s	chore(ac): ex3	now
 6.s	chore(ac): ex3	now
 readme.md	chore(ac): ex3	now

readme.md



Atividade 3

- Autor: Rafael Grossi

1

Altere o código acima para que ele utilize apenas os registradores salvos s0 até s11.
Você pode utilizar o simulador para verificar se o código está correto.

```
main:
    addi s5, zero, 11
    addi s11, zero, 256
    add s8, s11, s5
    ret
```



2

Altere o código acima para que ele utilize valores de imediatos diferentes. Explore valores que são maiores que o permitido e veja qual é o resultado. Você pode utilizar o simulador para verificar se o código está correto. Note a diferença entre o valor do seu código e o valor mostrado, com campo da instrução, no bloco inferior direito da tela (logo abaixo de imm).

```
main:
    addi s5, zero, 4294967296
    addi s11, zero, 1
    add s8, s11, s5
    ret
```



```
main:
    addi s5, zero, -2
    addi s11, zero, -2047
    add s8, s11, s5
    ret
```



3

Altere o código para que ele comece com uma leitura de um número inteiro e some o valor 2 nesse número.

```
main:
    addi t0, zero, 4 # stores on a0
    ecall
    addi s11, zero, 2
    add s8, s11, a0
    ret
```



4

Altere o código anterior para que ele escreva o valor da soma na saída.

```
main:
    addi t0, zero, 4
    ecall
    addi s11, zero, 2
    add a0, s11, a0
    addi t0, zero, 1
    ecall
    ret
```



5

Altere o código anterior para que ele leia dois números e escreva a soma deles na tela.

```
main:
    addi t0, zero, 4
    ecall
    addi s11, zero, 0
    add s8, s11, a0
    addi t0, zero, 4
    ecall
    add a0, s8, a0
    addi t0, zero, 1
    ecall
    ret
```



6

Altere o código anterior para que ele leia dois números e escreva o resultado de uma operação entre eles na tela. Você deve testar mais de uma instrução da tabela acima até ficar confortável com elas.

```
main:
    addi t0, zero, 4
    ecall
    addi s11, zero, 0
    add s8, s11, a0
    addi t0, zero, 4
    ecall
    xor a0, s8, a0
    addi t0, zero, 1
    ecall
    ret
```



```
main:
    addi s11, zero, 3
    addi s8, zero, 2
    and a0, s8, a0
    addi t0, zero, 1
    ecall
    ret
```

