



Exemplo de um programa que faz a leitura de 2 strings de no máximo 100 caracteres, e realiza a intercalação entre elas armazenando a sequência resultante em uma terceira string.

```
1  .data
2  ent1: .asciiz "Insira a string 1: "
3  ent2: .asciiz "Insira a string 2: "
4  str1: .space 100
5  str2: .space 100
6  str3: .space 200
7  .text
8  main: la $a0, ent1 # Parâmetro: mensagem
9        la $a1, str1 # Parâmetro: endereço da string
10       jal leitura # leitura(mensagem, string)
11       la $a0, ent2 # Parâmetro: mensagem
12       la $a1, str2 # Parâmetro: endereço da string
13       jal leitura # leitura(mensagem, string)
14       la $a0, str1 # Parâmetro: endereço da string 1
15       la $a1, str2 # Parâmetro: endereço da string 2
16       la $a2, str3 # Parâmetro: endereço da string 3
17       jal intercala # intercala(str1, str2, str3)
18       move $a0, $v0 # Move o retorno da string resultante
19       li $v0, 4 # Código de impressão de string
20       syscall # Imprime a string intercalada
21       li $v0, 10 # Código para finalizar o programa
22       syscall # Finaliza o programa
23  leitura:
24       li $v0, 4 # Código de impressão de string
25       syscall # Imprime a mensagem
26       move $a0, $a1 # Endereço da string para leitura
27       li $a1, 100 # Número máximo de caracteres
28       li $v0, 8 # Código de leitura de string
29       syscall # Faz a leitura da string
30       jr $ra # Retorna para a main
31  intercala:
32       move $v0, $a2 # Salva o endereço de str3 para retorno
33  c1: lb $t0, ($a0) # ch1 = str1[i]
34       beqz $t0, c2 # if(ch1 == NULL) goto c2
35       addi $a0, $a0, 1 # str1[i++]
36       beq $t0, 10, c2 # if(ch1 == '\n') goto c2
37       sb $t0, ($a2) # str3[k] = ch1
38       addi $a2, $a2, 1 # str3[k++]
39  c2: lb $t1, ($a1) # ch2 = str2[j]
40       beqz $t1, c # if(ch2 == NULL) goto c
41       addi $a1, $a1, 1 # str2[j++]
42       beq $t1, 10, c # if(ch2 == '\n') goto c
43       sb $t1, ($a2) # str3[k] = ch2
44       addi $a2, $a2, 1 # str3[k++]
45  c: add $t0, $t0, $t1 # ch1 = ch1 + ch2
46       bnez $t0, c1 # if(ch1 != 0) goto c1
47       sb $zero, ($a2) # str3[k] = NULL
48       jr $ra # Retorna para a main
```

- 1) Elaborar um programa, em código MIPS, que faça a leitura de uma string ASCII e verifique se a mesma é um palíndromo (retorne 1 se for palíndromo e 0 se não for palíndromo).
- 2) Elaborar um programa, em código MIPS, que faça a leitura de uma string e converta todos os caracteres desta string para maiúsculo. Depois de convertida a string, exibir o resultado na tela.
- 3) O Código de César é uma das mais simples e conhecida técnica de criptografia. A técnica consiste em substituir cada letra do texto por outra, que se apresenta no alfabeto abaixo dela um número fixo de vezes. Por exemplo, com a troca de três posições a letra B será substituída por E, e assim por diante. Elaborar um programa, em código MIPS, que utilize a técnica do Código de César com 4 posições, que faça a leitura de uma string e retorne a string codificada.
- 4) Elaborar um programa, em código MIPS, que contenha um menu com as seguintes opções:
  - a) Ler uma string S1 (tamanho máximo de 30 caracteres);
  - b) Imprimir o tamanho da string S1;
  - c) Comparar a string S1 com uma nova string S2 fornecida pelo usuário e imprima o resultado da comparação;
  - d) Concatenar a string S1 com uma nova string S2 e imprimir o resultado da concatenação;
  - e) Imprimir a string S1 de forma reversa;
  - f) Contar quantas vezes um dado caractere aparece na string S1. Esse caractere deve ser informado pelo usuário;
  - g) Substituir a primeira ocorrência do caractere C1 da string S1 pelo caractere C2. Os caracteres C1 e C2 são fornecidos pelo usuário;
  - h) Verificar se uma string S2 é substring de S1. A string S2 deve ser fornecida pelo usuário;
  - i) Retornar uma substring da string S1. O usuário deverá informar a partir de qual posição deve ser criada a substring e qual é o tamanho da substring.
- 5) Elaborar um programa, em código MIPS, que faz a leitura de uma string S, um caractere c, e uma posição i e devolve o índice da primeira posição da string onde foi encontrado o caractere c. A procura deve começar a partir da posição i.