**EXERCICIOS URI**

URI 1002 –<https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1002%20Area%20do%20Circulo.java>

URI 1024 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1024%20Criptografia.java>

URI 1072 –<https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1072%20Intervalo%202.java>

URI **1171** – PENDENTE

URI 1178 –<https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1178%20Preenchimento%20de%20Vetor%20III.java>

URI 1182 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1182%20Coluna%20na%20Matriz.java>

URI 1168 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1168%20LED.java>

URI **1192** – PENDENTE

URI 1234 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1234%20Senten%C3%A7a%20Dan%C3%A7ante.java>

URI **1238** – PENDENTE

URI **1241** – PENDENTE

URI 1253 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1253%20Cifra%20de%20C%C3%A9sar.java>

URI **1332** – PENDENTE

URI 1607 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1607%20Avance%20as%20Letras.java>

URI 1871 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_1871%20Zero%20vale%20Zero.java>

URI **2025** – PENDENTE

URI 2062 – <https://github.com/CavanhaMan/Cavanha-URI/blob/master/URI_2062%20OBI%20URI.java>

# EXERCICIOS GDB

### **String - Exercicio04 - (22/09) - Validar tamanho do CPF - utilizar funções**

Faça um programa que leia uma variável cpf, do tipo String. Verifique se o tamanho do CPF esta correto, um CPF precisa ter 11 dígitos. Caso ele esteja correto imprima 'valido', senão imprima 'invalido'.

obs.:

- A entrada terá apenas dígitos numéricos.

- Crie uma função que retorna boolean(true, false) e receba a String. Retorne True, se for válido e False, se for inválido.

|  |  |
| --- | --- |
| Entrada | Saída |
| 1122233344 | valido |
| 3424 | invalido |

### **String - Exercicio06 - (22/09) - Ler letra a letra na frase**

Faça um programa que receba como entrada uma frase e então imprima cada letra e sua posição, veja o exemplo abaixo.

|  |  |
| --- | --- |
| Entrada | Saída |
| IFTM | 0 - I  1 - F  2 - T  3 - M |

### **String - Exercicio07 - (22/09) - Tratar data**

Faça um programa que recebe como entrada a variável String 'data'. Então pegue a substring com os dois primeiros caracteres e salve em outra variável String 'd'. Então imprima "Estamos no dia 'd' do mes.".

obs.:

- Utilize a função substring.

|  |  |
| --- | --- |
| Entrada | Saída |
| 09/01/2020 | Estamos no dia 09 do mes. |
| 27/01/2019 | Estamos no dia 27 do mes. |

**Exercício 09 – Frase T**

Faça um programa que recebe como entrada uma frase T. O programa deverá imprimir as 3 primeiras letras de todas as palavras, que tenham mais de 3 caracteres, presentes na frase T.

obs.:

- Não considerar acentos.

- Considerar o “ponto e virgula” como parte das palavras.

Exemplo:

|  |  |
| --- | --- |
| Entrada: | Saída |
| Nos nao vemos as coisas como elas sao, mas como nos somos. | vem  coi  com  ela  sao  com  som |

**Exercício 10 – Strings S1 S2:**

Faça um programa que receba como entrada duas Strings s1 e s2.

O algoritmo deve imprimir a parte/substring da String s1 que começa com a primeira ocorrência de s2 ou toda a frase s1, caso s2 não exista em s1.

obs.:

1. Não considerar acentos.

2. A função substring tem duas implementações: substring(int, int) e substring(int).

|  |  |
| --- | --- |
| Entrada | Saída |
| Uberlandia tem dois campus do IFTM, o Uberlandia Centro e o Uberlandia. | IFTM, o Uberlandia Centro e o Uberlandia. |
| IFTM |