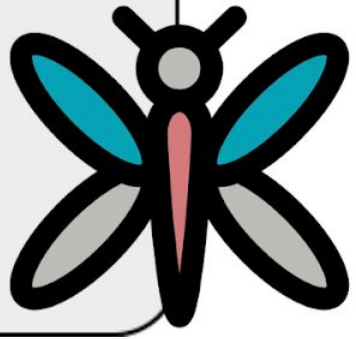
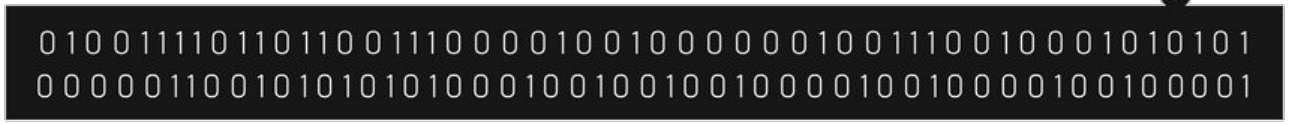
Projeto "Libélulas": Treinamento para OBI Modalidade Programação



Núcleo de Estudos e Pesquisa em Tecnologia da Informação

NEPeTI



INSTITUTO FEDERAL
Goiano

Campus
Ceres

Vai começar...

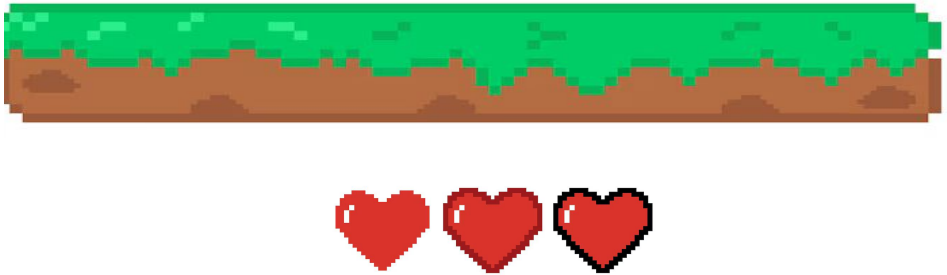
Boas-vindas!

Programação esportiva e divertida!

A jornada do estudante!

INSTITUTO FEDERAL
Goiano
Campus Ceres

bit.ly/**ranking_libelulas**



0101010101010101000010100101010101010101010000101011101110100011100010010101010101010101010100
0101010101010101000010100101010101010101010000101011101010101010101010100001010010101010101010101010

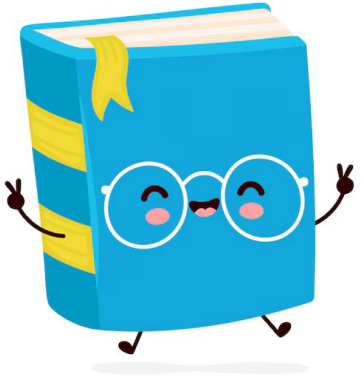
Manipulando Strings

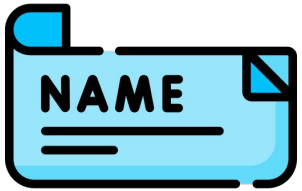
Aula 02

A string

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Variáveis do tipo **string** armazenam **cadeias de caracteres** como nomes e textos;
- Caracteres são **sequências** de símbolos, como letras, números, sinais;
- O tipo **string** é muito **útil** e bastante utilizado para **exibir mensagens.**





Utilização de string

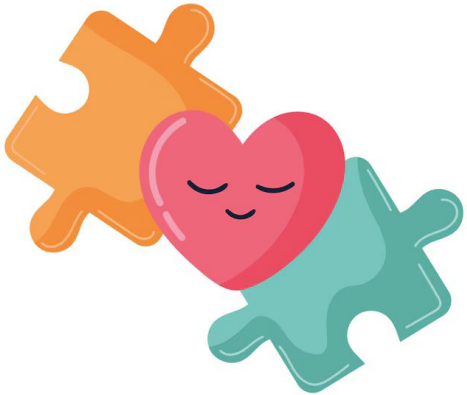
- A string está representada em **frases**, como “João e Maria”;
- É possível imaginar como uma **sequência de blocos**, em que cada letra, número ou espaço em branco ocupa;
- Sendo as **aspas** (“”) o **delimitador** do início e fim das caracteres.

String											
J	o	ã	o		e		M	a	r	i	a

A junção de textos!

INSTITUTO FEDERAL
Goiano
Campus Ceres

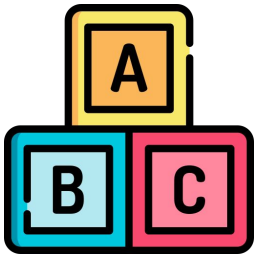
- A **concatenação** ou junção de **textos** é **soma** de duas partes de textos, em parte **única**;
- Para **concatenar** duas strings, pode ser utilizado o **símbolo de adição** (+).



Concatenação

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Utilizando o **símbolo de adição,** “AB” + “C” é **igual** a “ABC”;
- Essa **forma** de concatenação pode ser utilizado **apenas** entre **strings.**

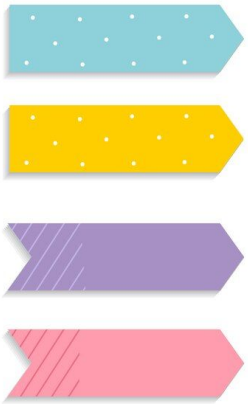


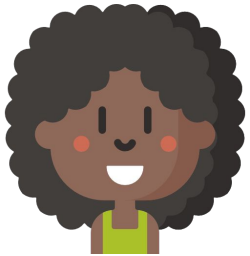
Concatenar significa **unir ou liga**r elementos de forma sequencial, como palavras, frases, caracteres…

Vamos usar marcadores?!

INSTITUTO FEDERAL
Goiano
Campus Ceres

- A construção de **mensagens** com a junção não é sempre prático, principalmente **combinando tipos de dados diferentes** para formar uma string;
- Para isso, utiliza-se marcadores de **posição**;
- Exemplo:
  - **Exibir** que “Maria tem X anos”, em que **X** é do tipo **numérico.**





Exemplo com marcadores

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Para cada **tipo** de marcação, é utilizado um **símbolo**;
- Exemplo com a **variável** nome = “João” e idade = 16;
- Em `"%s tem %d anos." % (nome, idade)`
  - **Leia-se** “João tem 16 anos”

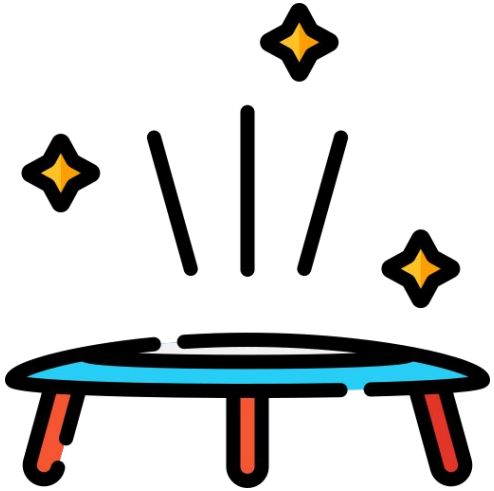
Assim, é possível unir texto + variáveis, fazendo uso da **interpolação (ou composição)** de strings em Python!

Marcadores	
%d	Números inteiros
%s	Strings
%f	Números decimais

Marcadores em Python

INSTITUTO FEDERAL
Goiano
Campus Ceres

- O símbolo de **% após a string**, indica que a variável vai ser referenciada no texto;
- O **%d** é conhecido como o **marcador de string**, indicando que naquela **posição** está um **valor inteiro.**



```
>>> print ("Maria tem %i anos" % idade)
```

Trabalhando com o format

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Outra forma de trabalhar isso é por meio do **format**;
- Então, seguindo o exemplo com as **variáveis anteriores**, pode-se utilizar:
  - `f "{nome} tem {idade} anos"`
    - **Leia-se** “João tem 16 anos”
- No format, é necessário colocar apenas o **“f”** antes das aspas e chaves “**{}**” com as variáveis.

Arredondando dados numéricos

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Em alguns casos, vai ser necessário **arredondar números**, utilizando **marcadores**;
- Com o **format**, é necessário colocar **dois pontos ":"** após a menção à variável, **em seguida** colocar um ponto e a quantidade de casos decimais (arredondamento).

```
print(f"SALARY = U$ {salario:.2f}")
```

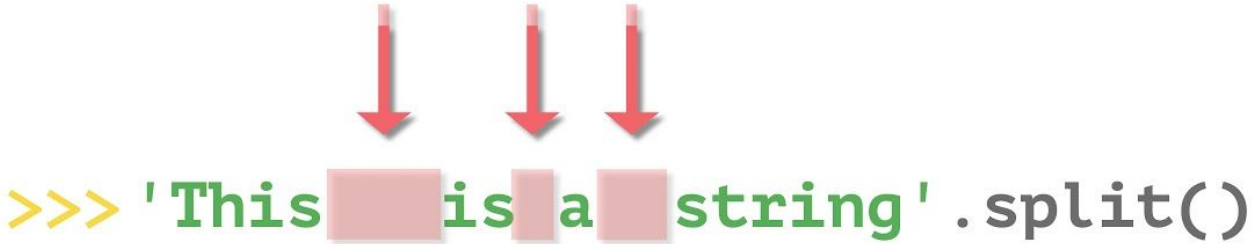
Split

Aula 02

A função mágica!

INSTITUTO FEDERAL
Goiano
Campus Ceres

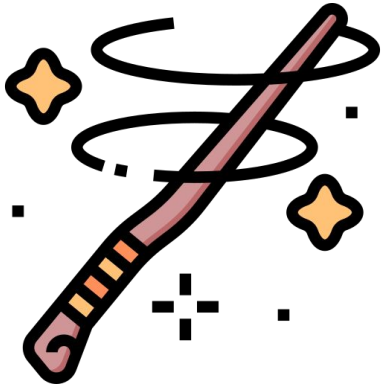
- A função **split** em Python separa a partir de um **caractere** passado como **parâmetro**;
- Ou seja, o método **quebra** o texto (frase) a partir de um **separador.**

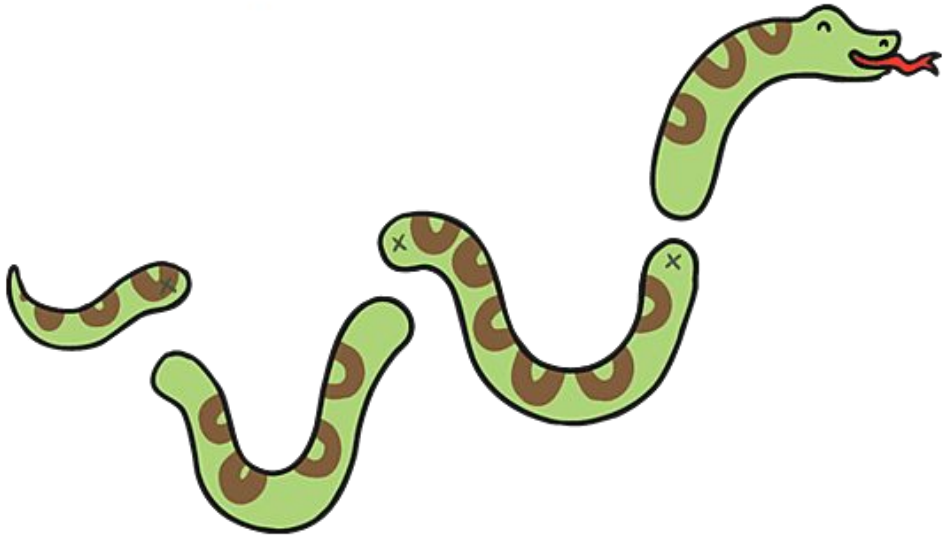


A função mágica!

INSTITUTO FEDERAL
Goiano
Campus Ceres

```
>>> cobra.split()
```





A função mágica!

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Para usar a **função mágica**, devemos definir o **separador** (parâmetro) e **declarar** da seguinte forma:

```
frase = "Harry Potter"

retorno = frase.split()
```

Pode ser declarado colocando o texto e depois o ponto com a função

A hora da magia

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Como **padrão** da **função** sem passar nenhum **parâmetro**, o separador é compreendido como o espaço em branco;
- O **resultado** é o seguinte:

```
['Harry', 'Potter']
```

Usando diferentes parâmetros

INSTITUTO FEDERAL
Goiano
Campus Ceres

- É possível utilizar diferentes **separadores**, como a **vírgula**.

```
frase =  "Olá, seja bem-vindo (a)!"

retorno = frase.split(",")

>> ['Olá', ' seja bem-vindo (a)!']
```

A função split e input

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Para inserir **dois valores** ao **mesmo tempo**, pode ser utilizado a função input e **split,** sendo que **padrão** a função de **entrada** recebe **string**;
- Deve **digitar** os dados em seguida, separados **por espaço.**


```
a, b = input().split()
print (f"Valor de a: {a}, b:{b}",a,b)
```

```
Valor de a: Maria, b:João
```

Operações Lógicas

Aula 02

Variáveis do tipo lógico

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Para armazenar um conteúdo simples, como **verdadeiro ou falso**, é utilizado o tipo **lógico (booleano);**
- Em Python, é codificado **True** para verdadeiro e **False** para falso.

Variáveis do tipo lógico

INSTITUTO FEDERAL
Goiano
Campus Ceres

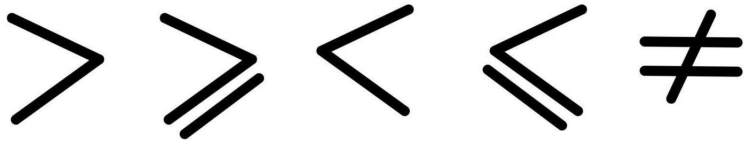
- Para realizar comparações **lógicas** (verdadeiro ou falso), utilizaremos os **operadores relacionais**;
- Os operadores são **representados** por **símbolos**, como de **igualdade**, maior que, menor que, **diferente**, **maior ou igual.**

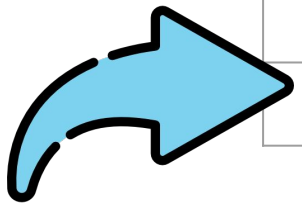
Operadores relacionais

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Para **descobrir** se determinada **sentença** é **falsa** ou **verdadeira**, utiliza-se os seguintes **operadores.**

Operadores	
==	Igualdade
>	Maior que
<	Menor que
!=	Diferente
>=	Maior ou igual

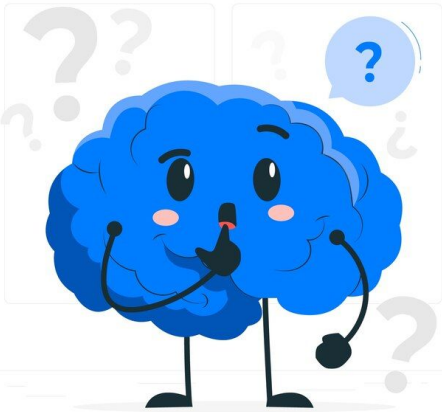




A operação lógica vs atribuição

INSTITUTO FEDERAL
Goiano
Campus Ceres

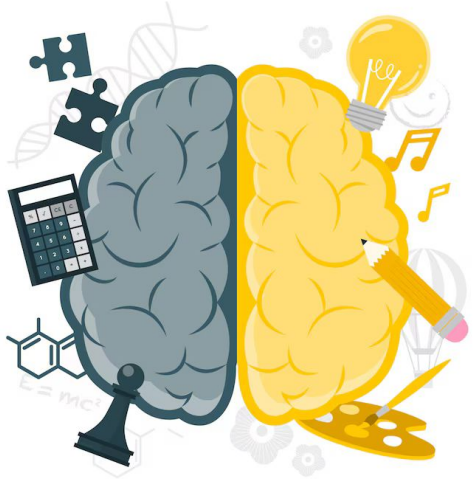
- Os símbolos usados em **Python** lembram a **matemática**, mas **não são exatamente iguais;**
- A igualdade para **operação lógica** deve ser usada com **dois iguais** (==), **diferente** da **atribuição** de valores com apenas **um igual** (=).



Usando operadores relacionais!

INSTITUTO FEDERAL
Goiano
Campus Ceres

- **Operadores relacionais** são utilizados para **comparar** se determinado **valor** é igual**,** diferente, maior e etc;
- Como exemplo, **considere** a = 10 e b = 5:
  - a > b (**True**)
  - a == b (**False**)
  - a != b (**True**)



Operadores lógicos

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Operadores lógicos são denominados por algumas operações básicas:
  - **AND, OR e NOT**
- Responsáveis por **agrupar** operações de **verdadeiro** ou **falso** (booleanas).

Operadores	
NOT	Não
AND	E
OR	Ou

Usando operadores lógicos

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Utilizando as **operações lógicas**, considere a = **True**, b = **True** e c = **False**;
- Para **utilizar** os operadores lógicos, lembre-se que devem estar **entre as variáveis**;
  - Exemplo: **AND** b, c **OR** a;

AND, OR, NOT

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Para utilizar **AND**, é retornado **True** apenas se ambas as variáveis forem **verdadeiras**
  - a **AND** b = **True**
- **OR** é retornado **True** quando uma das variáveis forem verdadeiras
  - a **OR** b = **True**

AND, OR, NOT

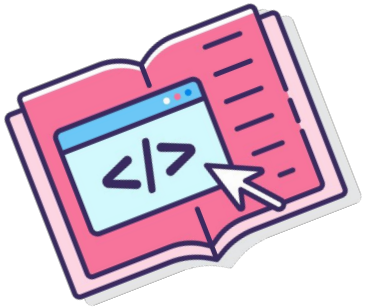
INSTITUTO FEDERAL
Goiano
Campus Ceres

- Enquanto que **NOT** é considerado a negação do valor **lógico** de uma variável, considere:
  - a = **True**
  - **NOT** a = **false**

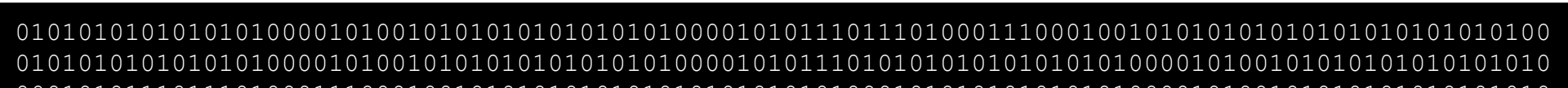
- É a **operação lógica** que inverte o valor booleano, **falso** para **verdadeiro** e **vice-versa.**

Agora é a sua vez!

INSTITUTO FEDERAL
Goiano
Campus Ceres



Prática com a Lista 01 no Beecrowd



Para quem não entrou ainda...

INSTITUTO FEDERAL
Goiano
Campus Ceres

Treinamento OBI 2025 (Iniciante)

POR Thalia Santos de Santana

bcwd.me/d-14379

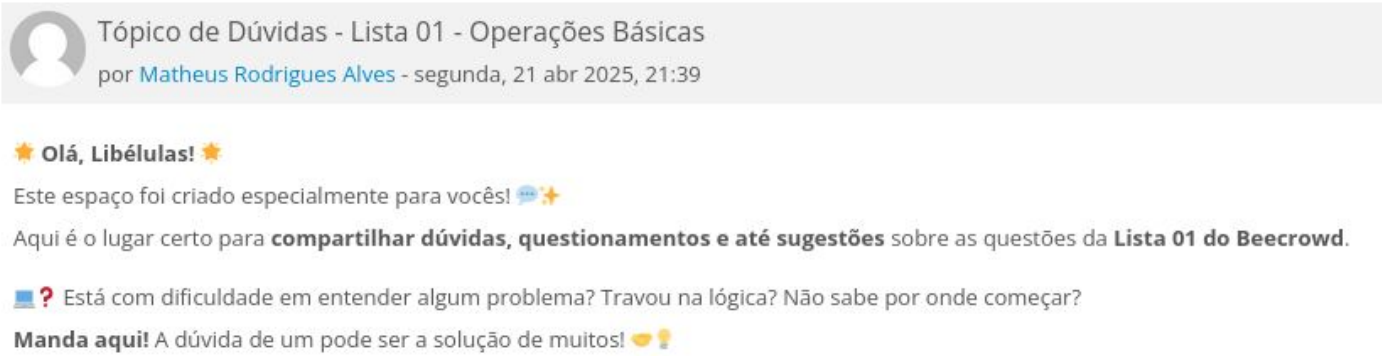
ZPwq.Fw

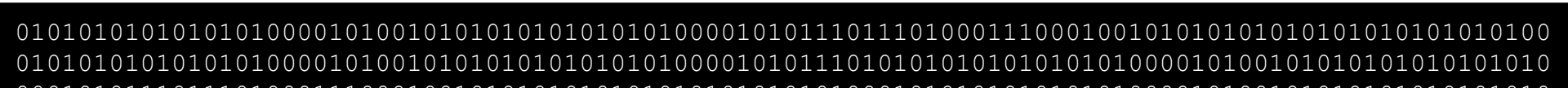
BEECROWD.COM

Fórum no Moodle para dúvidas!

INSTITUTO FEDERAL
Goiano
Campus Ceres

- Na **sala virtual do projeto,** está **disponível** um fórum para dúvidas, que serão **pontuadas** na gamificação:





Resolvendo em sala - 1008

INSTITUTO FEDERAL
Goiano
Campus Ceres

beecrowd | 1008

Salário

Adaptado por Neilor Tonin, URI Brasil

Timelimit: 1

Escreva um programa que leia o número de um funcionário, seu número de horas trabalhadas, o valor que recebe por hora e calcula o salário desse funcionário. A seguir, mostre o número e o salário do funcionário, com duas casas decimais.

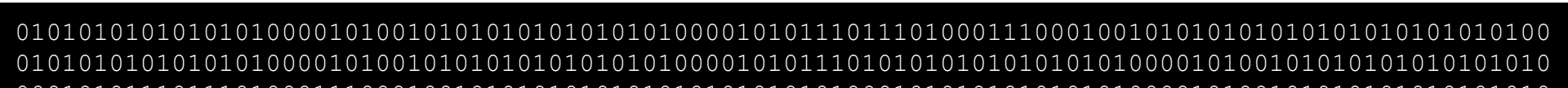
Entrada

O arquivo de entrada contém 2 números inteiros e 1 número com duas casas decimais, representando o número, quantidade de horas trabalhadas e o valor que o funcionário recebe por hora trabalhada, respectivamente.

Saída

Imprima o número e o salário do funcionário, conforme exemplo fornecido, com um espaço em branco antes e depois da igualdade. No caso do salário, também deve haver um espaço em branco após o $.

Exemplos de Entrada	Exemplos de Saída
25 100 5.50	NUMBER = 25 SALARY = U$ 550.00



Exercício - 1008 - Beecrowd

INSTITUTO FEDERAL
Goiano
Campus Ceres

```
numero = int(input())
horas = int(input())
valor = float(input())

salario = valor * horas

print(f"NUMBER = {numero}")
print(f"SALARY = U$ {salario:.2f}")
```

NEPeTI

Treinando em casa
Lista 01 e 02 no Beecrowd

Dúvidas no fórum do moodle!

Lista 01 e 02 para praticar!

INSTITUTO FEDERAL
Goiano
Campus Ceres

DISCIPLINA

TREINAMENTO OBI 2025 (INICIANTE)

#	HOMEWORK	INICIAR	DATA LIMITE
48703	Lista 01 - Operações Básicas (13)	15/04/2025 14:39	06/05/2025 15:00
48795	Lista 02 - Operações Básicas - Parte 2 (8)	22/04/2025 15:00	06/05/2025 15:00

0101010101010101000010100101010101010101010000101011101110100011100010010101010101010101010100
0101010101010101000010100101010101010101010000101011101010101010101010000101001010101010101010

Dúvidas e perguntas?!

@nepeti.ce no Instagram

obi.ce@ifgoiano.edu.br

informatica.ifgoiano.edu.br

Fórum de dúvidas (Moodle)

NEPeTI

Material Licenciado

Este documento está licenciado sob uma licença Creative Commons CC BY-NC-SA 4.0.
Texto da licença: https://creativecommons.org/licenses/by-nc-sa/4.0/deed.pt-br

Esta licença permite que outros remixem, adaptem e criem a partir do seu trabalho para fins não comerciais, desde que atribuam a você o devido crédito e que licenciem as novas criações sob termos idênticos.

Atribuição — Você deve dar o crédito apropriado, prover um link para a licença e indicar se mudanças foram feitas.

NãoComercial — Você não pode usar o material para fins comerciais.

Compartilhalgual — Se você remixar, transformar, ou criar a partir do material, tem de distribuir as suas contribuições sob a mesma licença que o original.

NEPeTI