

## **Projeto CodeLab Teen**

### **Desenvolvendo habilidades para o futuro**

## **Atividade proposta 3:**

### **Introdução**

Na nossa primeira aula, aprendemos sobre instruções, onde um de vocês teve que guiar outro pela sala usando apenas comandos predefinidos. Isso nos ensinou que os computadores, assim como o colega sendo guiado, só fazem exatamente o que lhes é instruído, nem mais, nem menos.

Na segunda aula, mergulhamos no fascinante mundo dos números binários e da criptografia. Descobrimos que os computadores pensam em '0's e '1's e que, com isso, é possível representar qualquer informação, desde o seu nome até uma foto sua. E mais, aprendemos que a informação pode ser protegida e escondida através da criptografia, garantindo que somente quem deve ver, veja!

### **Requisitos**

Nesta parte serão mencionados os materiais necessários para a realização da dinâmica:  
(Inserir materiais)

**Cartões com desafios de programação.**

**Fita para montar o tabuleiro.**

**Um espaço físico (como uma sala ou quadra) com o tabuleiro desenhado.**

**Bombons para o grupo vencedor.**

### **Desenvolvimento da atividade**

Cada grupo começa em um ponto inicial no tabuleiro ou espaço físico. É feita uma pergunta por grupo. Caso eles resolvam o problema, eles avançam uma casa; se errarem, permanecem na mesma casa até a próxima rodada. O objetivo é chegar ao final do tabuleiro ou espaço físico, tendo enfrentado e resolvido os desafios de programação.

## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

#### Cartões

#### CARD's NÍVEL 1

##### Conceito de Variável:

Imaginem que vocês têm uma caixa.

Dentro dessa caixa, vocês podem guardar qualquer coisa: uma bola, um celular, ou mesmo um número.

Agora, imagine que esta caixa tem um nome, e sempre que você chamar esse nome, você pode ver ou mudar o que está dentro dela.

Na programação, essa "caixa" é chamada de variável.

As variáveis são usadas para guardar informações que podem ser usadas ou modificadas mais tarde.

Elas têm um nome, assim como a nossa caixa imaginária, e este nome é usado para acessar o valor armazenado.

Por exemplo, se tivermos uma variável chamada "idade", podemos armazenar o número 12 nela. Mais tarde, podemos mudar o número para 13 quando fizermos aniversário.

O que está dentro delas (seu valor) pode mudar, mas o nome da variável permanece o mesmo. É por isso que são chamadas de "variáveis" - porque o que elas guardam pode variar.

#### 1. Você encontrou uma poção mágica!

Você tem uma variável `saúde = 50`.

Beba a poção e ganhe 30 pontos à sua saúde.

Qual é o valor da sua saúde agora?

#### 2. Você tem uma variável chamada chuva.

Sempre que chove, `chuva = True` de verdadeiro

Quando não chove, `chuva = False` de falso

Hoje está chovendo,

Qual o valor de `chuva`?

## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

3. Você está em uma floresta mágica coletando frutas.

Você tem uma variável ``frutas = 5``.

Você encontrou uma árvore com mais 3 frutas. Adicione-as à sua coleção.

Quantas frutas você tem agora?

4. Em sua aventura, você tem uma mochila para carregar itens.

Inicialmente, ``mochila = 0``.

Você coloca 4 pedras mágicas em sua mochila.

Quanto itens há na mochila agora?

Caso **tenham errado** algum dos cards nível 1, podemos usar os seguintes cards adicionais:

5. Em uma missão para salvar o reino, você tem ``chaves = 2``.

Você usa uma chave para abrir um portal.

Quantas chaves você tem depois disso?

6. Durante uma batalha, você tem ``poções = 3``.

Um inimigo quebra 2 das suas poções.

Quantas poções você tem agora?

## CARD's NÍVEL 2

Conceito de Lógica, tomada de decisão.

Na programação, muitas vezes nos deparamos com problemas que exigem que avaliemos várias condições para tomar uma decisão. A lógica nos permite analisar e comparar informações, levando-nos a uma conclusão ou ação específica. Quando falamos de lógica em programação, muitas vezes nos referimos a instruções condicionais ("se isso, então aquilo") e a comparações (como "maior que", "menor que", "igual a", etc.). Dominar a lógica é crucial para escrever programas que se comportem da maneira que queremos.

## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

1. Se João é mais alto que Maria e Maria é mais alta que Pedro, quem é mais alto?  
R: João
2. Um avião caiu na fronteira entre o Canadá e os EUA. Onde eles enterram os sobreviventes?  
R: Os sobreviventes não são enterrados.
3. Antes do Monte Everest ser descoberto, qual era a montanha mais alta do mundo?  
R: O Monte Everest (ele sempre foi a montanha mais alta, só não havia sido descoberto).
4. Se Ana tem mais livros que Carla e Carla tem menos livros que Bruno, quem tem mais livros?  
R: Ana (ou Bruno, não?)
5. Você está correndo em uma corrida e ultrapassa a pessoa em segundo lugar. Em que posição você está agora?  
R: Segundo Lugar.
6. Complete a sequência 1,1,2,3,5,8,...
7. Complete a sequência 1,4,9,16,...
8. Se um galo põe um ovo em cima de um telhado, para qual lado o ovo rola?  
R: Os galos não põem ovos.

Caso **tenham errado** algum dos cards nível 2, podemos usar os seguintes cards adicionais:

9. Quantos animais o Moisés levou para a Arca?  
R: Nenhum, foi Noé.
10. Todos os gatos de Luísa são pretos. Alguns gatos pretos são alérgicos a peixe. Se um gato de Luísa come peixe, ele é alérgico?  
R: Talvez.
11. Na sala há 5 meninas e 7 meninos. Se 2 meninas e 3 meninos saem, quem tem mais representantes na sala?  
R: Meninos.
12. Se um médico lhe deu 5 pílulas e disse para você tomar uma a cada meia hora, quanto tempo elas vão durar?

## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

R: 2 Horas.

- 13.** Se o trem A parte 15 minutos antes do trem B e ambos têm a mesma velocidade, qual trem chegará primeiro ao destino?

R: Trem A

- 14.** Complete a sequência 5,10,20,40,...

### CARD's NÍVEL 3

Conceito de Condicional.

Em nossa vida cotidiana, frequentemente tomamos decisões com base em certas condições. Por exemplo: "Se estiver chovendo, então levarei um guarda-chuva". Esta é a essência dos condicionais na programação.

No mundo da programação, condicionais permitem que os computadores tomem decisões. Usando condicionais, os programas podem escolher entre diferentes caminhos de ação com base nas circunstâncias.

Se uma condição for falsa, o código dentro do condicional não será executado, e o programa seguirá em frente.

A estrutura básica de um condicional em muitas linguagens de programação segue o padrão "se... então...". Em inglês, usamos as palavras "if" (se) e "else" (senão). Também usamos == para comparar se for igual e != para comparar se for diferente, e os sinais >, <, etc.

**Mostrar os cards para as crianças para elas resolverem.**

**1.**

```
chave = "prateada"
if chave == "dourada":
    resultado = "O baú abriu!"
else:
    resultado = "O baú não abriu."
```

**Qual o valor de resultado no final, o baú abriu ou não?**

## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

2.

```
chovendo = true
if (chovendo == true)
    { João = 'levar guarda-chuva'; }
else { João = 'não levar nada'; }
```

**João levou ou não o guarda chuva?**

3.

```
temperatura_inverno = 10
if temperatura_inverno < 20:
    ação = "agasalhar"
else: ação = "não agasalhar"
```

**Você se agasalhou?**

4.

```
parque= "ensolarado"
if parque != "chuvoso":
    ação = "ir ao parque"
else: ação = "ficar em casa!"
```

**Você ficou em casa?**

**Ensinar o conceito de lógica com mais de uma condicional para ser satisfeita, E ou OU, em computação usamos && e || .**

1.

```
porta= "fechada"
chave = "no trinco"
if porta == "aberta" || chave = "no trinco"
    printf("passagem liberada")
else: printf("passagem não liberada")
```

**A porta está aberta ou fechada?**

## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

2.

```
celular= "ligado"  
bateria= "cheia"  
if celular== "ligado" && bateria == "cheia"  
    printf("mexendo no celular")  
else: printf("não foi possível mexer no celular")
```

**Você conseguiu mexer no celular?**

3.

```
nota1 = 85  
nota2 = 65  
if nota1 > 70 or nota2 > 70:  
    print("Você passou!")  
else: print("Você precisa estudar mais.")
```

**Se nota1 for 68 e nota2 for 72, qual será a saída?**

4.

```
temperatura = 20  
clima = "ensolarado"  
if temperatura > 18 and clima == "ensolarado":  
    print("Vamos para a praia!")  
else: print("Melhor ficarmos em casa.")
```

**Se a temperatura for 22 e o clima for "nublado", qual será a saída?**

Caso **tenham errado** algum dos cards nível 2, podemos usar os seguintes cards adicionais:

5.

```
inverno= "Ensolarado"  
if parque!= "chuvoso":  
    ação = "ir ao parque"  
else: ação = "ficar em casa!"
```

**Você ficou em casa?**

## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

6.

```
cão= "faminto"  
if cão != "cheio":  
    ação = "dar comida"  
else: ação = "não dar comida"
```

**Você deu comida para cachorro?**

#### CARD's NÍVEL 4

##### Loops

Em programação, quando queremos que uma ação seja repetida várias vezes, usamos o que chamamos de "loop" ou "laço". É como se estivéssemos dizendo ao computador: "Continue fazendo isso até que eu lhe diga para parar!" ou "Faça isso 10 vezes!". Por exemplo, podemos usar um loop para dizer: "Mostre 'Olá!' na tela 5 vezes". E o computador mostrará: 'Olá' 5 vezes.

Em programação fazemos o laço usando a palavra "for", que significa 'para'. Nós usamos uma variável para controlar quantas vezes o laço acontece e um número para dizer qual o final (quando parar de fazer determinada ação).

Mostre o código para os alunos: explicando que printf é um termo em programação para imprimir, e  $i = i + 1$  é a soma de  $i + 1$ .

**1.** Você foi enfeitado e está andando em círculos!

**Quantas vezes você diz "Estou perdido!" neste código?**

```
for (i = 0; i < 5; i = i + 1){  
    printf("Estou perdido!"); }
```

\*Explique que também podemos fazer o laço usando while, que significa enquanto. Use o código abaixo:

**2.** Alguém lhe deu uma tarefa repetitiva.

**Quantas vezes você pula antes de cansar?**



## Projeto CodeLab Teen

### Desenvolvendo habilidades para o futuro

```
contagem = 0
while (contagem < 7):
    contagem = contagem + 1
```

#### 3. Quantas vezes a frase "Estourou um balão!" será impressa?

```
balões = 5
while balões > 0:
    print("Estourou um balão!")
    balões = balões - 1
```

Caso **tenham errado** algum dos cards nível 3, podemos usar os seguintes cards adicionais:

#### 4. Quantas vezes a frase "Deu um passo!" será impressa?

```
passos = 0
while passos < 4:
    passos += 1
    print("Deu um passo!")
```

#### 5. Quantas vezes a frase "Deu uma volta na pista!" será impressa?

```
voltas = 3
while voltas > 0:
    print("Deu uma volta na pista!")
    voltas = voltas - 1
```

## CARD's NÍVEL 5

Curiosidades

1. A primeira linguagem de programação foi criada por quem?

- a) Ada Lovelace
- b) Alan Turing
- c) Blaise Pascal

## **Projeto CodeLab Teen**

### **Desenvolvendo habilidades para o futuro**

2. A linguagem de programação Python foi nomeada em homenagem a cobra Python - Verdadeiro ou Falso

#### **Questão desafio a todos os grupos**

Há cinco casas, cada uma de uma cor diferente. Em cada casa mora uma pessoa de uma nacionalidade diferente. Cada uma das cinco pessoas bebe uma bebida diferente, fuma um tipo diferente de cachimbo e tem um animal de estimação diferente.

Dicas:

1. O norueguês vive na primeira casa.
2. O inglês vive na casa vermelha.
3. O verde está à esquerda do branco.
4. O dinamarquês bebe chá.
5. A pessoa que fuma Marlboro vive na casa amarela.
6. Na terceira casa, bebe-se leite.
7. A pessoa da segunda casa fuma Chesterfield e bebe água.
8. Na casa verde, bebe-se café.
9. O sueco tem um cachorro como animal de estimação.
10. O norueguês vive ao lado da casa azul.

Pergunta: Quem tem peixes como animal de estimação?

Resposta: Com base nas dicas fornecidas, e após deduzir as informações, descobrimos que é o alemão que tem peixes como animal de estimação.

(Esta é uma versão simplificada de um quebra-cabeça famoso atribuído a Albert Einstein, que muitas vezes é acompanhado pela alegação de que 98% das pessoas são incapazes de resolvê-lo.)

#### **Conclusão (parte para os monitores)**

Discursão livre.