



INSTITUTO FEDERAL
São Paulo
Câmpus São Carlos

Algoritmos com Scratch

ALG – Algoritmos e Programação

Aula 06

Curso Técnico em Informática para Internet – Integrado ao Ensino Médio



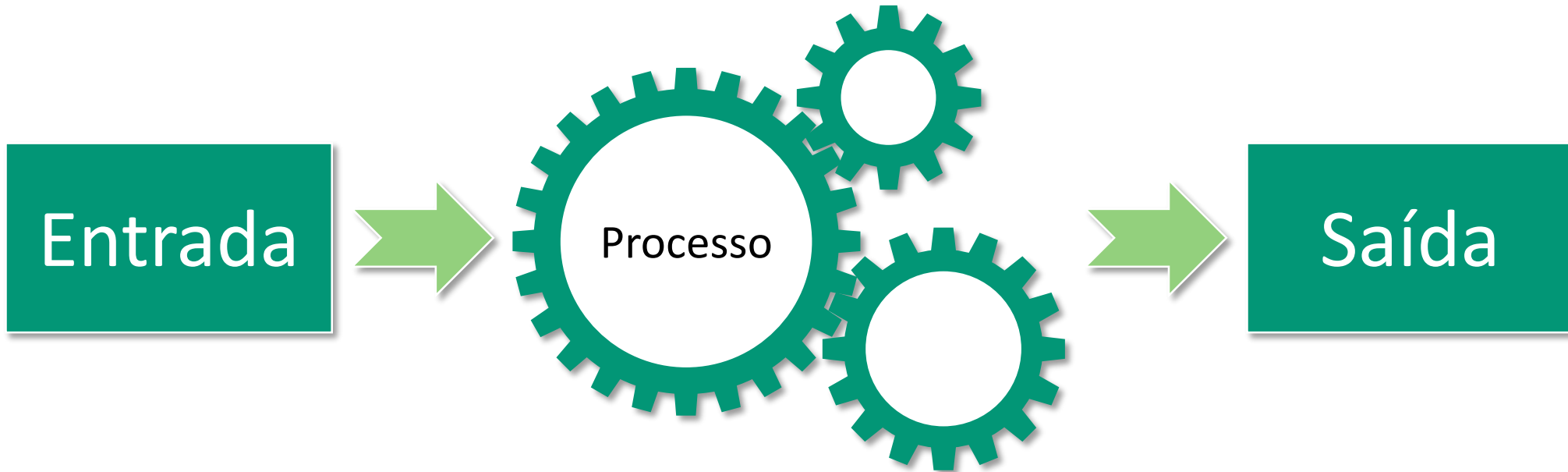
INSTITUTO FEDERAL
São Paulo
Câmpus São Carlos

Objetivos da aula

- Conhecer o conceito de algoritmo computacional, programa e linguagens de programação
- Conhecer os tipos de dados e variáveis
- Conhecer expressões aritméticas
- Conhecer comandos de entrada e saída
- Construir os primeiros programas com Scratch, utilizando variáveis, expressões aritmética e entrada e saída de dados

Algoritmo

- Conjunto de regras e procedimentos lógicos que, se seguidos, levam a solução de um problema em um número finitos de etapas
- Sequência de passos para a execução de uma tarefa



Algoritmos Computacionais

- O computador deve executar a tarefa
- Precisamos de uma linguagem de programação
- É preciso transformar a ideia em um programa

Algoritmo

- Conjunto de regras e procedimentos lógicos que, se seguidos, levam a solução de um problema em um número finito de etapas

Programa

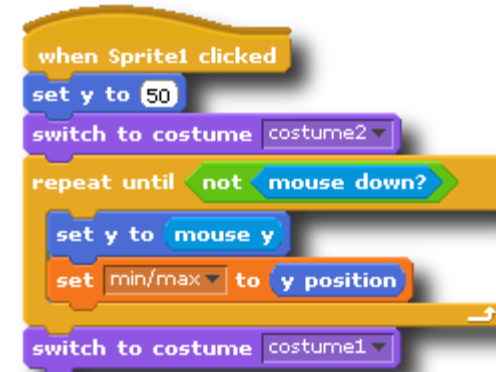
- Formalização de um algoritmo em uma linguagem de programação, segundo suas regras de sintaxe e semântica, de forma a permitir que o computador possa entender a sequência de ações

Linguagens de Programação

- Existem muitas linguagens de programação
 - Pascal
 - C
 - C++
 - Java Script
 - **Python**
 - Java
 - **Scratch**
- As linguagens de programação permitem descrever precisamente o algoritmo
- Elas possuem regras que devem ser conhecidas e obedecidas
 - Não é necessário conhecer todas as regras para começar a programar

Linguagens de Programação

- A maioria das linguagens de programação utilizam a notação textual para escrita de programas
- Como já apresentado, Scratch se utiliza de diagramas de bloco



Exemplo de algoritmo computacional em Scratch

EXEMPLO

- Algoritmo para calcular média de um aluno após 2 notas



Algoritmos, instruções, dados e informações

- Um algoritmo computacional deve ser uma sequência de instruções manipulando dados, que geram determinada informação
 - **Instruções:** comandos que definem integralmente uma operação a ser executada. Determinam a forma pela qual os dados devem ser tratados.
 - **Dados:** elementos conhecidos de um problema. Podem ser recolhidos/fornecidos por diversos meios e serão processados pelo computador através das instruções.
 - **Informação:** Um conjunto estruturado de dados, podendo transmitir conhecimento.

Comandos de entrada e saída

Comandos de entrada e saída

- Comandos de entrada e saída: Comandos que permitem a interação do usuário com o programa via os dispositivos de entrada e saída.

- Comando de entrada:

pergunte pergunta **e espere a resposta**

- Comando de saída:

diga Hello! **por 2 segundos**

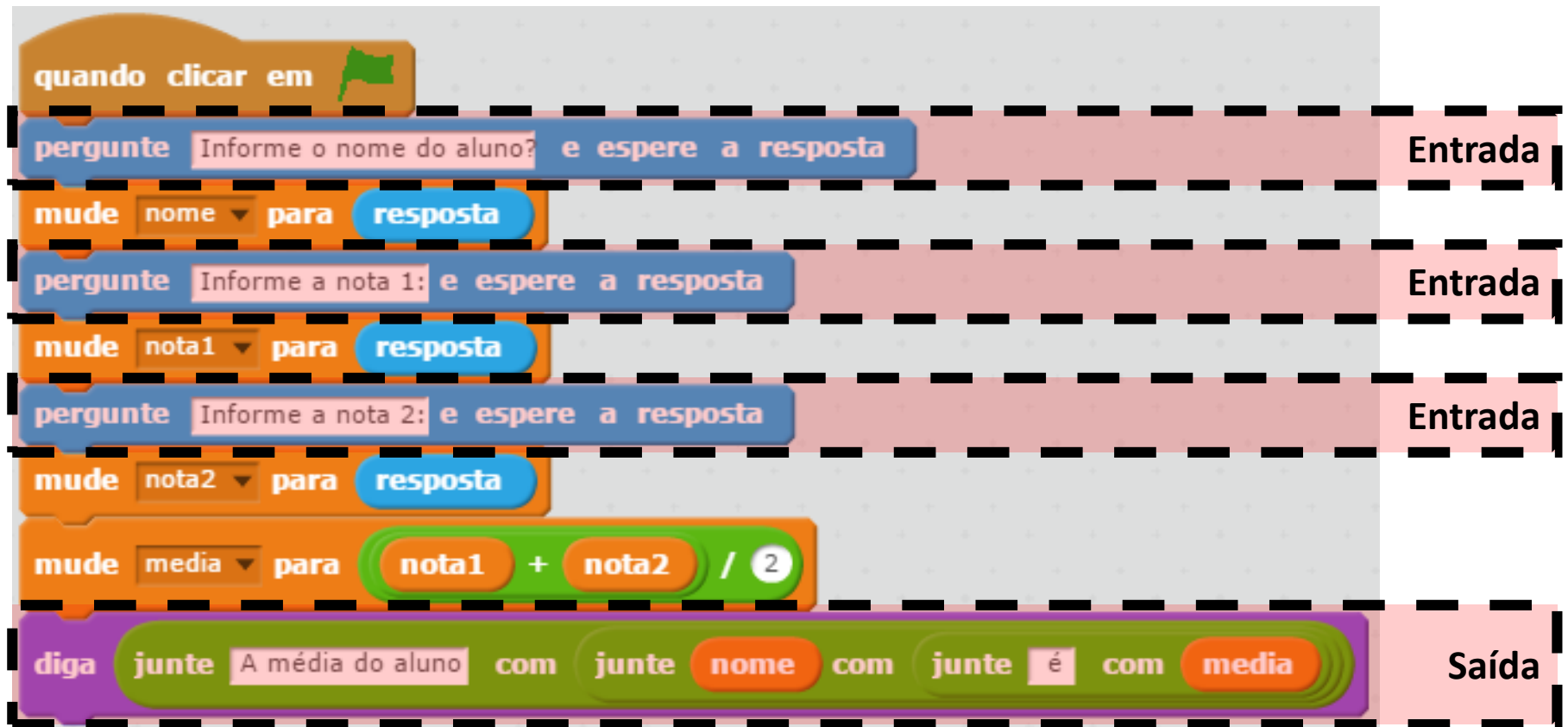
diga Hello!

pense Hmm... **por 2 segundos**

pense Hmm...

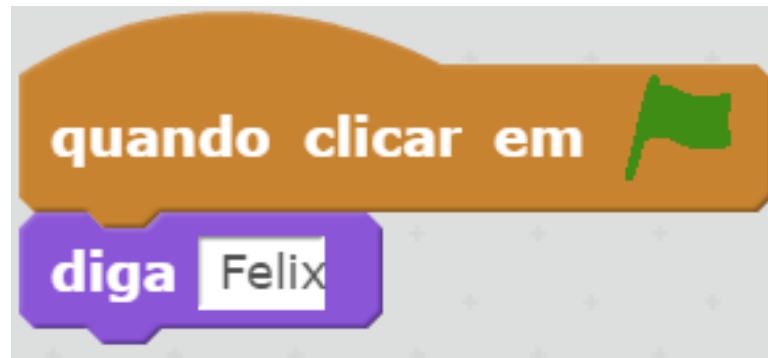
Comandos de entrada e saída

- Onde estão os comandos de entrada e saída no nosso exemplo?



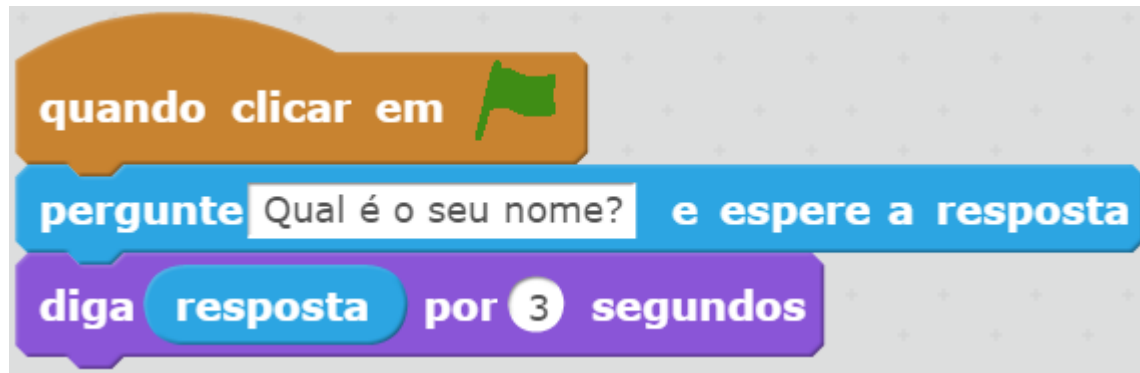
Atividade 1

- Criar um projeto que contenha um evento de clique na bandeira verde. Este evento deve fazer com que o ator diga o seu nome.



Atividade 2

- Criar um projeto que contenha um evento de clique na bandeira verde. Este evento deve disparar o seguinte conjunto de ações:
 1. Perguntar o nome do usuário
 2. Dizer o nome do usuário por 3 segundos



Atividade 3

- Alterar o programa anterior para dizer: “Olá, fulano” (fulano é a resposta da pergunta) por 3 segundos.



Tipos de dados

Dados e tipos de dados

- Um dado é uma informação que um algoritmo recebe ou manipula
- Exemplos de dados são:
 - Nomes
 - Datas
 - Valores (preços, notas, etc.)
 - Condições (verdadeiro e falso)
- Todo dado é de um certo tipo que define sua natureza (p. ex., um nome é diferente de um preço)
- Identificando seu uso, define-se as operações que podem ser realizadas com o dado
 - Por exemplo, podemos somar dois valores numéricos, mas não podemos somar um número e uma frase.

Dados e tipos de dados

- Os tipos de dados mais básicos em algoritmos são o numérico, o literal e o lógico.
- Um tipo de dado especifica:
 - A quantidade de bytes que deve ser reservada para uma constante ou variável.
 - Como o dado representado por esses bytes deve ser interpretado (o que significa a cadeia de bits).
 - Exemplo: Se o dado numérico é um inteiro ou um número real

Tipos de dados: Numérico

- Dados numéricos: podem ser basicamente de dois tipos de dados:
 1. **Inteiro:** representa um número inteiro.
 - Exemplos: -12; 0; 511.
 - Dados deste tipo podem ser usados para idade em anos, número de filhos etc.
 2. **Reais:** representa número real, ou seja, com casas decimais
 - Exemplos: 7,5; 12,75; 57,1.
 - Dados deste tipo podem ser usados para saldo bancário, altura, peso, temperatura, etc

Tipos de dados: Literal

- Dados literais: conjunto de símbolos que pode ser formado por números, letras e símbolos especiais. Tipos de dados literais:
 1. **Caractere:** símbolo pertencente ao conjunto de símbolos alfanuméricos:
 - Numéricos (0 a 9); Alfabéticos (A...Z, a...z); Especiais (*, +, %, #, @, !, etc)
 - Exemplo: 'M', 'F', 'x', '?'
 - Podem ser usados para armazenar, por exemplo, o sexo de uma pessoa: 'F' ou 'M'
 2. **String:** é uma cadeia de caracteres
 - Exemplos: "aula", "lógica de programação" , "14815-000"
 - Dados deste tipo podem ser usados para armazenar nomes, frases, endereços, etc.

Tipos de dados: Lógico

- Dados lógicos utilizados para representar dois valores únicos. Exemplos:
 - VERDADEIRO ou FALSO
 - TRUE ou FALSE
 - 1 ou 0
- São usados para expressar uma condição:
 - $4 > 5$ é falso
 - O cheque número 12331 já foi compensado? SIM ou NÃO?.
- Também conhecido como tipo booleano, em referência à álgebra de Boole
 - Notação muito comum

Tipos de Dados

- Quais tipos de dados são usados no nosso exemplo?

The image shows a Scratch script with the following blocks and annotations:

- quando clicar em** (green flag icon)
- pergunte** Informe o nome do aluno? e espere a resposta (blue block) → **String**
- mude** nome para resposta (orange block)
- pergunte** Informe a nota 1: e espere a resposta (blue block) → **Real**
- mude** nota1 para resposta (orange block)
- pergunte** Informe a nota 2: e espere a resposta (blue block) → **Real**
- mude** nota2 para resposta (orange block)
- mude** media para $\text{nota1} + \text{nota2} / 2$ (orange block) → **Real**
- diga** junte A média do aluno com junte nome com junte é com media (purple block)

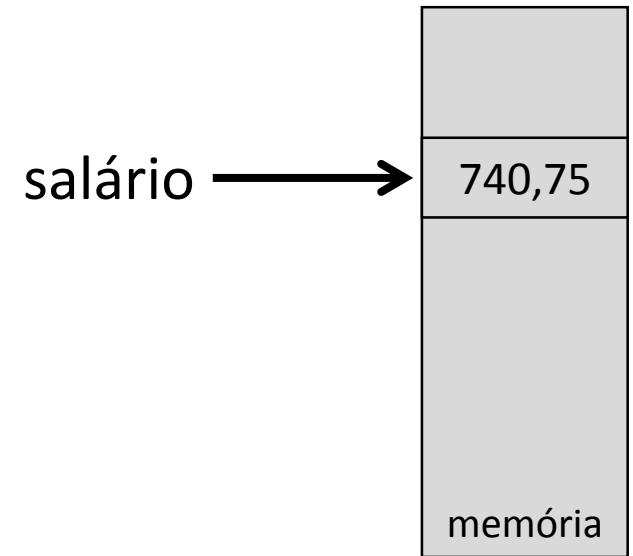
Atividade 4

- Determine o tipo primitivo de dado presente em:
 - O cartaz: “Desligue o celular na sala de aula!”
 - O carro vai de zero a cem km/h em 6,7 segundos
 - José Manoel tem 47 anos
 - A sentença acima é verdadeira
 - CPF
 - RG
 - Data de nascimento
 - Taxa de juros
 - Campeão da Copa do Mundo
 - Temperatura
 - Se o carro funciona ou não

Variáveis

Variáveis

- Ao desenvolver um programa, frequentemente é necessário armazenar dados referentes a ele, como um nome, um número ou mesmo o resultado de uma operação
- Para armazenar esses dados é necessário solicitar ao computador que ele reserve uma área da memória para uso. Para reservar essa memória devem ser criadas variáveis
- Uma variável tem a função de associar um nome a uma porção da memória onde um dado pode ser armazenado

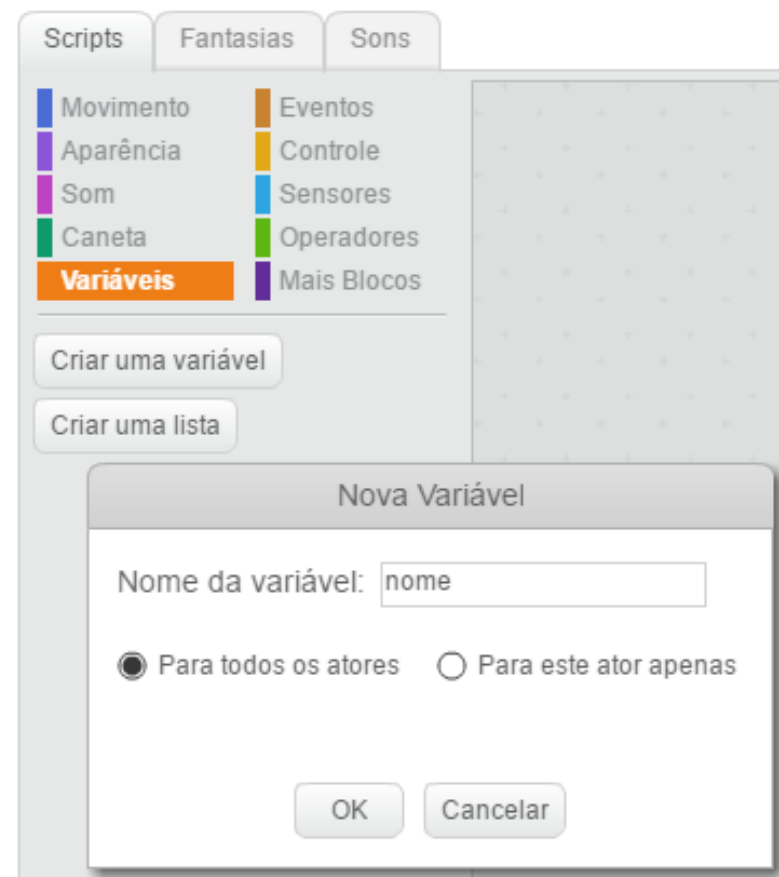


Variáveis – Identificadores

- Identificadores
 - Identificam uma variável por meio de um nome
 - Substituem endereços de memória
 - Toda variável possui um identificador
- Um identificador é formado por um ou mais caracteres alfanuméricos, sendo que o primeiro caractere deve ser, obrigatoriamente, uma letra
- Exemplo:
 - nota, a, x, w, X25, X_1 (identificadores válidos)
 - 5B, X-Y, E(13), A&B, A e B, (identificadores não válidos)

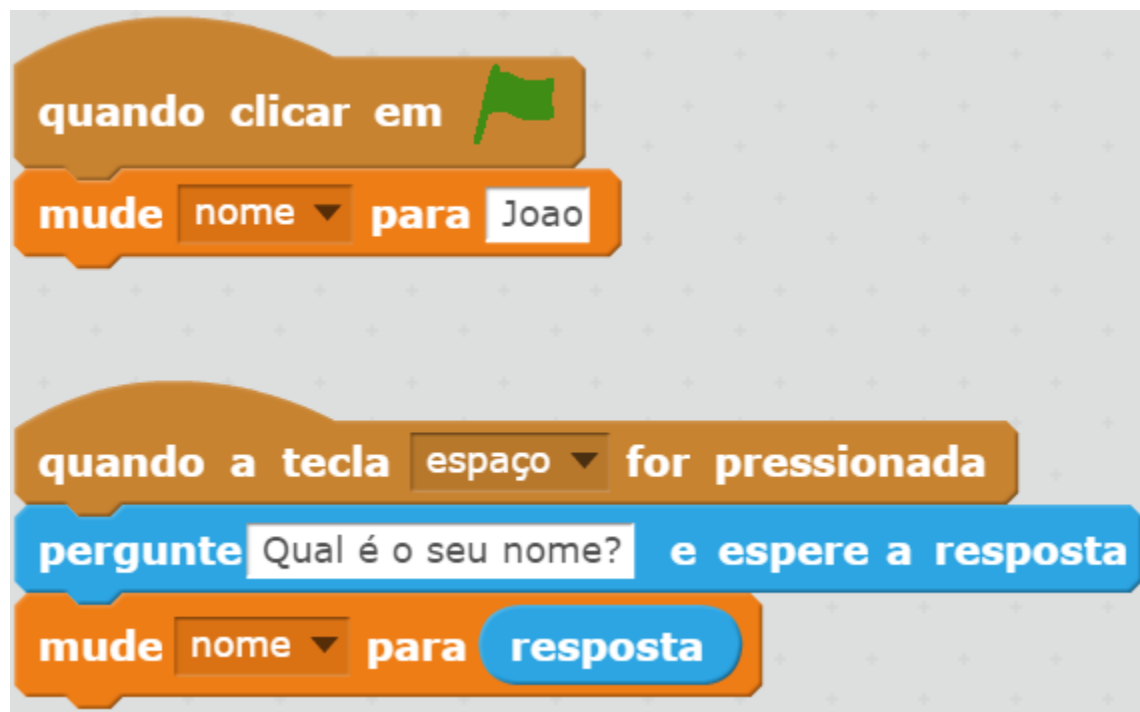
Variáveis – Criação

- Em Scratch, uma variável é criada por meio da instrução “Variáveis”

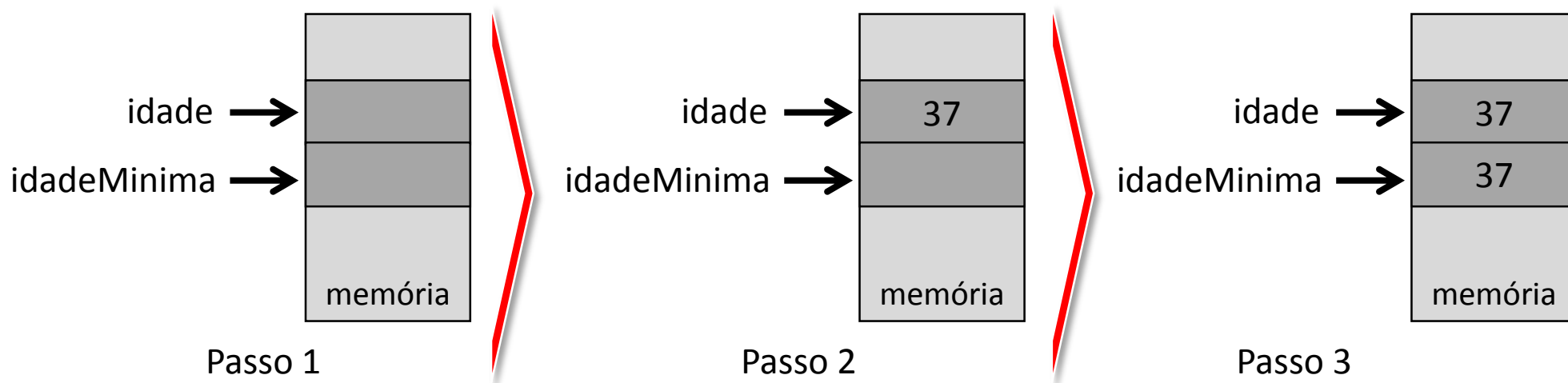


Variáveis – Atribuição de valor

- É possível atribuir um valor a variável ou ainda a armazenar a resposta de uma pergunta nela



Variáveis – Exemplo de operações



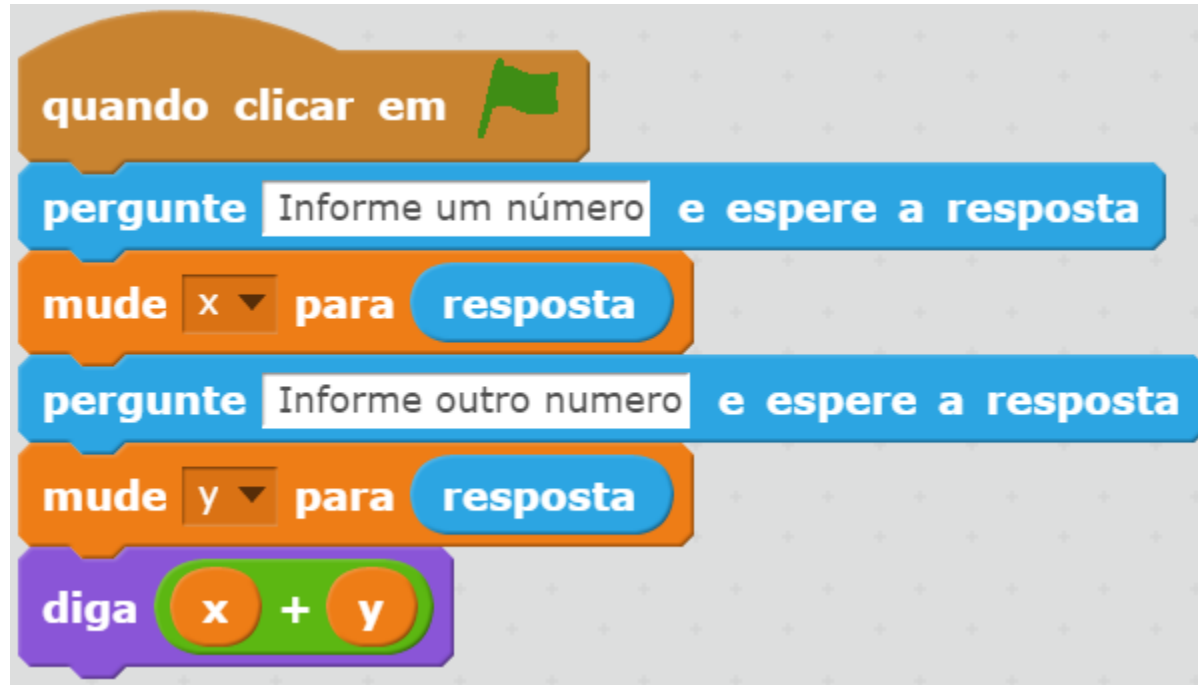
Variáveis

- Onde estão as variáveis no nosso exemplo?



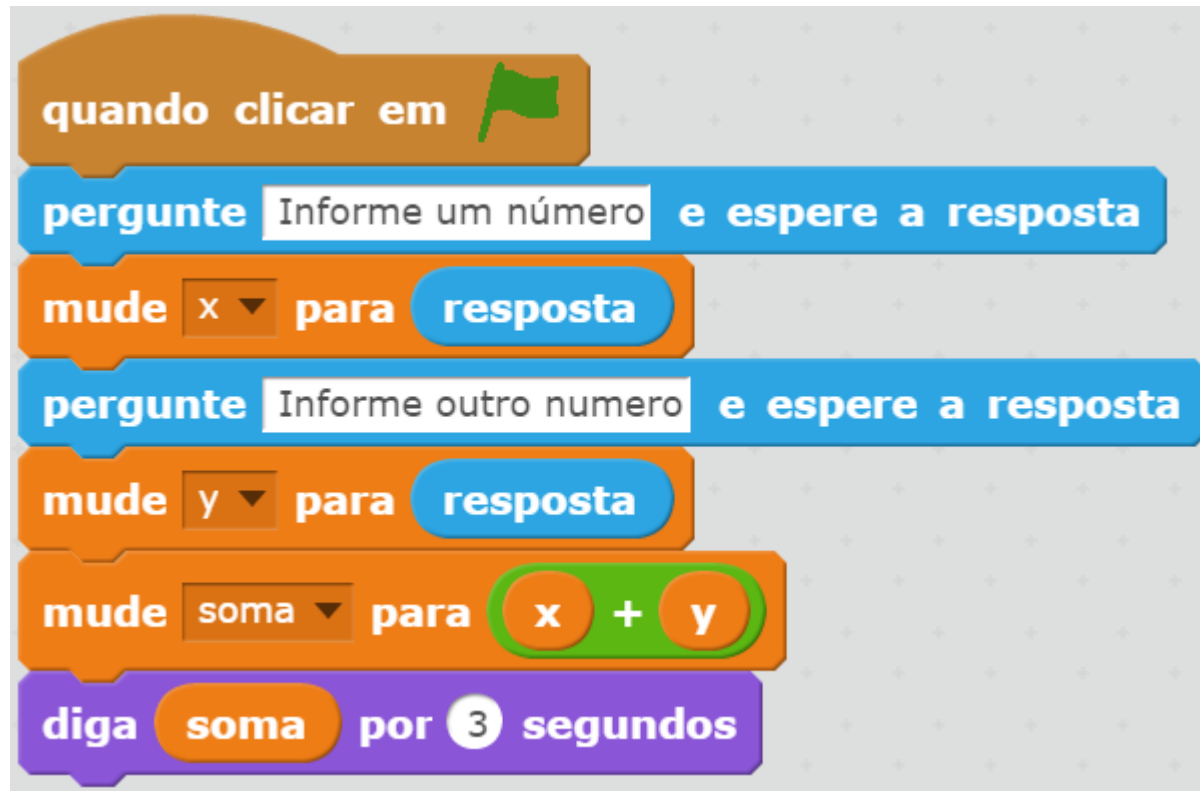
Atividade 5

- Criar um programa que pergunte dois números ao usuário e armazene em duas variáveis. Em seguida diga qual o resultado da soma dos dois.



Atividade 6

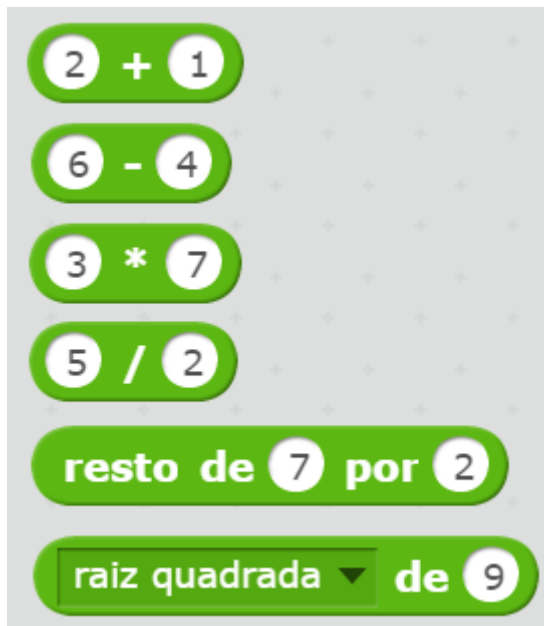
- Altere o programa anterior para armazenar o resultado da operação em uma variável e depois mostrá-lo por 3 segundos.



Expressões aritméticas

Expressões aritméticas

- Expressões aritméticas: são operações cujos operadores são aritméticos e cujos operandos variáveis do tipo numérico
- Operadores aritméticos:



Expressões aritméticas

- Precedências



Parênteses mais internos

Multiplicação e divisão

Soma e subtração

Atividade 7

- Quais os valores finais de x e y?



x 220

y 110

Expressões aritméticas

- Onde estão expressões aritméticas no nosso exemplo?



Exercícios



Exercício 1

- Faça um programa em Scratch que leia um número X do usuário e escreva ele na tela no seguinte formato: “O número escolhido foi X”

Exercício 2

- Faça um programa em Scratch que leia do usuário dois números. Calcule a multiplicação dos dois números e mostre o resultado.

Exercício 3

- Faça um programa em Scratch que leia do usuário um número e escreva:
 - Seu sucessor por 3 segundos
 - Seu antecessor por 2 segundos

Exercício 4

- Faça um algoritmo em Scratch que leia do usuário duas notas de prova. Calcule a média simples e mostre o resultado.

Exercício 5

- Usando o exercício anterior, inicie o programa solicitando o nome do aluno e use para exibir sua média.

Exercício 6

- Faça um programa que pergunte ao usuário o tamanho do lado de um quadrado. Após isso, faça com que o ator desenhe o quadrado na tela e diga a área dele.

Exercício 7

- Faça um programa que pergunte ao usuário dois números inteiros que representam os dois lados de um retângulo. Após isso, faça com que o ator desenhe o retângulo na tela e diga a área dele.

Exercício 8

- Faça um programa em Scratch que leia 2 notas de um aluno, onde a primeira nota possui peso um, a segunda possui peso dois. Calcule a média ponderada do aluno baseada nos pesos e exiba.

Exercício 9

- Faça um programa em Scratch que leia um número inteiro e diga ao usuário o resultado do resto da divisão desse número por 3.
 - Para pensar: como descobrir se os números são pares ou ímpares?

Exercício 10

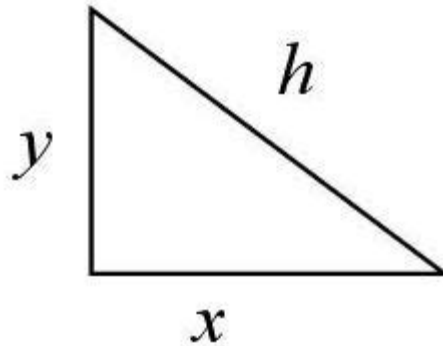
- O índice de massa corporal (IMC) é uma medida internacional usada para calcular se uma pessoa está no peso ideal. O IMC é determinado pela divisão da massa do indivíduo pelo quadrado de sua altura, em que a massa está em quilogramas e a altura em metros, ou seja:
 - $IMC = (massa/altura^2)$
- Faça um programa que leia a massa e altura da pessoa e calcule o IMC dela.

Exercício 11

- Calibrar os pneus do carro deve ser uma tarefa cotidiana de todos os motoristas. Para isto, os postos de gasolina possuem uma bomba de ar. A maioria das bombas atuais são eletrônicas, permitindo que o motorista indique a pressão desejada num teclado. Ao ser ligada ao pneu, a bomba primeiro lê a pressão atual e calcula a diferença de pressão entre a desejada e a lida. Com esta diferença ela esvazia ou enche o pneu para chegar na pressão correta. Sua ajuda foi requisitada para desenvolver o programa da próxima bomba da SBC - Sistemas de Bombas Computadorizadas. Faça um programa que, dada a pressão desejada digitada pelo motorista e a pressão do pneu lida pela bomba, indica a diferença entre a pressão desejada e a pressão lida.

Exercício 12

- Faça um programa que leia o valor dos catetos de um triângulo retângulo e calcule a hipotenusa, de acordo com o Teorema de Pitágoras.
- Escreva uma saída adequada para o seu algoritmo



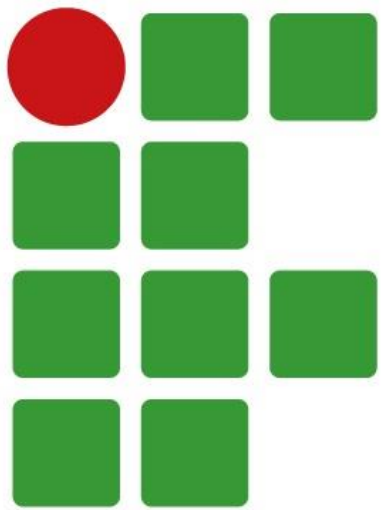
$$h^2 = x^2 + y^2$$

Exercício 13

- A Olimpíada Internacional de Informática (IOI, no original em inglês) é a mais prestigiosa competição de programação para alunos até o ensino médio; a cada ano, aproximadamente 300 competidores, de oitenta países, se reúnem em um país diferente para as provas da competição. Naturalmente, os competidores usam o tempo livre para acessar a Internet, programar e jogar em seus notebooks. No entanto, eles se depararam com um problema: o saguão do hotel só tem uma tomada. Felizmente, os quatro competidores da equipe brasileira da IOI trouxeram cada um uma régua de tomadas, permitindo assim ligar vários notebooks em uma só tomada do saguão. Eles sabem também que podem ligar uma régua em outra para aumentar ainda mais o número de tomadas disponíveis. No entanto, como as réguas têm muitas tomadas, eles pediram para você fazer um programa que, dado o número de tomadas em cada régua, determine o número máximo de notebooks que podem ser conectados a tomadas num mesmo instante.
- **Entrada:** Ler quatro números inteiros positivos T1, T2, T3, T4, indicando o número de tomadas de cada uma das quatro réguas.
- **Saída:** Seu programa deve informar um único número inteiro, indicando o número máximo de notebooks que podem ser conectados a tomadas num mesmo instante.
- Exemplos: 2 4 3 2 → 8 | 6 6 6 6 → 21

Referências

- BADGER, Michael. Scratch 2.0 Beginner's Guide – Second Edition
- A Hora do Código – <https://studio.code.org/>
- Scratch Brasil – <http://www.scratchbrasil.net.br/>
- Projeto Programando o Futuro – Introdução à Programação de Computadores – UNICAMP
- Olimpíada Brasileira de Informática
- Adaptação dos materiais didáticos dos professores
 - Adenilso Simão
 - Giampaolo Libralon
 - Rosely Sanches
 - Eduardo R. Hruschka
 - Pedro Cacique



INSTITUTO FEDERAL

São Paulo

Câmpus São Carlos