

Matheus Luiz Massuda

Atividade03

(Fórum)

PROMPT PARA PESQUISA NA INTELIGÊNCIA ARTIFICIAL

Explique, detalhadamente, sobre entrada e saída de dados em algoritmos, esclareça: quais são as diferenças entre elas, qual a importância desses conceitos ao desenvolver um programa (na prática), cite exemplos práticos fáceis (apenas texto) e também mais complexos (em Java) para melhor entendimento. Responda como um professor de universidade especialista na área e use como referência livros aclamados sobre o tema. Ao explicar, ajuste a linguagem para um estudante de Desenvolvimento de Sistemas que ainda não é capaz de entender conceitos extremamente avançados.

RESUMO

Entrada de dado (Input): é o processo de receber informações externas para serem processadas pelo programa. As maneiras mais comuns de chegada desses tipos de dados são: teclado (via usuário), arquivo, banco de dados e até mesmo sensores (exemplo: robôs aspiradores).

Saída de dado (Output): é o processo de enviar informações processadas pelo programa para o mundo externo. As maneiras mais comuns são por meio de: exibição (impressão) na tela, gravação em arquivo ou até mesmo envio para uma impressora.

Ambos os tipos de dados são importantes pois permite **interação** homem (usuário) x máquina (computador, smartphone, etc.). Além disso após a entrada de dados, eles podem ser armazenados para uso futuro, sem necessidade de nova requisição e permitem comunicação **sistêmica** com outros dispositivos. A melhor maneira de entender seria uma calculadora simples em que um dado de

entrada seira o usuário digitando dois números e o sinal desejado da operação, após isso o programa interno da calculadora irá processar esses dados do tipo numérico e exibirá o resultado da operação na tela. Em resumo entrada e saída de dados é a base da interação entre programas e o mundo externo.

EXEMPLO EM PORTUGOL STUDIO

```
programa
{
    funcao inicio()
    {
        inteiro lado_a, lado_b, lado_c

        escreva ("Informe o primeiro lado do triângulo: ")
        leia (lado_a)

        escreva ("Informe o segundo lado do triângulo: ")
        leia (lado_b)

        escreva ("Informe o terceiro lado do triângulo: ")
        leia (lado_c)

        se (lado_a == lado_b e lado_a == lado_c)
        {
            // Se os três lados forem iguais é equilátero

            escreva ("\nEste triângulo é Equilátero\n")
        }
        senao
        {
            // Se chegou aqui é porque os três lados não são iguais
            // Basta ver se dois deles são iguais para saber se é isósceles

            se (lado_a == lado_b ou lado_b == lado_c ou lado_c == lado_a)
            {
                escreva ("\nEste triângulo é Isósceles\n")
            }
            senao
            {
                escreva ("\nEste triângulo é Escaleno\n")
            }
        }
    }
}
```

Esse exemplo, em Portugol Studio visa captar dados de **entrada** que correspondem aos lados de um triângulo e retornar como **saída** a classificação do tipo de triângulo após o processamento. É importante notar que não houve a **validação** dos dados que comprovem que as medidas dos lados fornecidas pelo usuário formam, de fato, um triângulo. Para isso, seria necessário acrescentar outra parte lógica no programa para conferir a possibilidade desses valores formarem um triângulo, previamente (a soma de dois lados deve, obrigatoriamente, ser maior que o valor do terceiro lado sozinho).

Nesse caso as **entradas** são os respectivos valores dos **lados do triângulo**, fornecida via teclado pelo usuário. Já as **saídas** são as respostas dos tipos de triângulo (equilátero, isósceles ou escaleno) depois do processamento do programa.

REFERÊNCIAS

ChatGPT. **Generative Pre-Trained Transformer** Disponível em: < <https://chatgpt.com/share/67c5aeb0-be50-8012-b535-6e6660b137a2> >. Acesso em: 3 mar. 2025.

Perplexity. **Inteligência Artificial de pesquisa e conversão baseado em IA com texto preditivo** Disponível em: < https://www.perplexity.ai/search/explique-detalhadamente-sobre-fRj4nzQIQheUIQbr52_slQ >. Acesso em: 3 mar. 2025.