### 

Simple Programming Language - SPL

### Apresentado por:

* Felipe Mori Ferreira RA: 822129885
* Carlos armando Munhoz vilela RA: 822138213
* Caio André Porto de Oliveira RA: 822156202
* Giulio Enrico Miranda Maciotta RA: 822159355
* Gustavo Morais Cardoso RA: 822154343

### Sumário

Introdução\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_03

### Tipos de Dados\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_04

### Variáveis\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_05

### **Introdução**

### **SPL: Sua Primeira Aventura na Programação!**

### Você está dando os primeiros passos no fascinante mundo da programação? Que ótimo! Aprender a programar é como aprender uma nova língua, abrindo portas para um universo de possibilidades. E para te ajudar nessa jornada, vamos falar sobre a **SPL**.

### **O que é a SPL?**

### SPL é uma sigla que pode representar diversas linguagens de programação, dependendo do contexto. Para facilitar a sua compreensão, vamos considerar aqui a SPL como uma linguagem simplificada, especialmente projetada para iniciantes. Ela serve como uma espécie de ponte entre a lógica humana e a linguagem de máquina, tornando o processo de aprendizado mais intuitivo e divertido.

### **Por que aprender SPL?**

### **Simplicidade:** A SPL é construída com uma sintaxe clara e concisa, facilitando a compreensão dos conceitos básicos da programação.

### **Foco na lógica:** Ao invés de se preocupar com detalhes técnicos complexos, você poderá se concentrar em desenvolver a sua lógica de programação, que é a base para qualquer linguagem.

### **Preparo para outras linguagens:** Uma vez que você dominar os fundamentos da programação com a SPL, será mais fácil aprender outras linguagens mais complexas, como Python, Java ou C++.

Tipos de Dados

* Inteiro ----------------------------------------------------------------------- 10
* Real ------------------------------------------------------------------------- 0.2
* Caractere ------------------------------------------------------------------ ‘a’
* Literal ----------------------------------------------------------------------- “Uma frase “
* Lógico ---------------------------------------------------------------------- Verdadeiro

### 

### 

### Palavras Chaves e Suas Estruturas

### 0 1. If/Else (Novo nome: check/otherwise) - CONDICIONAL

* O check será usado como um substituto do if para verificar uma condição, e o otherwise substituirá o else.

### 2. While/Do (Novo nome: loop/until) - CONDICIONAL

* O loop será usado como o equivalente ao while para criar laços repetitivos, e o until substituirá o do como uma condição final de parada.

### 3. For (Novo nome: iterate) - REPETIÇÃO

* iterate substituirá o for e será utilizado para criar laços que percorrem intervalos de valores, listas ou estruturas.

### 4. Print (Novo nome: show) – VERIFICAÇÃO/VALIDAÇÃO

* A função show será usada para exibir valores na tela ou no console, como uma substituição para o print.

### 5. Console (Novo nome: log) - VERIFICAÇÃO/VALIDAÇÃO

* log será usado como uma interface para interagir com o console, permitindo não apenas a exibição de mensagens, mas também para fins de debug e registro de código.

Variáveis

A declaração de variáveis deve ser feita dentro de um bloco específico, que deve aparecer logo após a declaração do algoritmo. O bloco é iniciado pela palavra-chave “variaveis” (sim, sem acento) e termina com a palavra-chave “fim-de-variaveis”. Pelo menos uma variável deve ser declarada dentro do bloco (embora o bloco em si seja opcional) e apenas um bloco em escopo global deve ser declarado. Eis um exemplo para estudo:

algoritmo teste\_variaveis;

variaveis

x : inteiro;

nome : literal;

fim-variaveis

início

fim