

LINGUAGEM DO BOLEIRO

Luís Souza

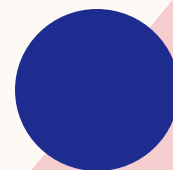
ITENS

Motivação

Características

Curiosidades

Exemplo



MOTIVAÇÃO

Dado que estamos em ano de Copa do Mundo, nada mais justo que criar uma linguagem que homenageie os "boleiros" do Brasil. Pensando nisso, foi criada a Linguagem do Boleiro, que utiliza-se primariamente de termos futebolísticos informais brasileiros. Além disso, foi um teste para atestar a modularidade do meu compilador desenvolvido ao longo do curso, já que eu busquei checar quão fácil seria portar a linguagem do compilador para outra com termos diferentes.

CARACTERÍSTICAS

A linguagem é capaz de realizar operações aritméticas e booleanas, além de criar e setar valores de variáveis e funções. Além disso, o código só pode ser lido se estiver contido em uma função, e a função que é automaticamente lida pelo código terá nome de "Main". Ela também pode imprimir, realizar loops e operações condicionais, além de pegar input do usuário. Os tipos possíveis são inteiros e strings.

CURIOSIDADES

A linguagem, por mais que seus termos todos estejam em português, não é acessível a todos os brasileiros, já que muitos dos termos contidos nela só farão sentido para os entendedores de futebol.

EXEMPLO

- O exemplo abaixo contém todos os possíveis comandos da linguagem, e representa bem o que ela é capaz ou não de fazer.
- Nele, o usuário é pedido dois inputs. Para os inputs 1 e 1, o código irá dar de output: 3 (após o primeiro input) e 3, 2, 1, 0 (para o segundo input).

```
regulamento soma(x: i32, y: i32) -> i32{
    recua x + y;
}

regulamento Main(){
    // v2.3 testing
    jogador x_1: i32;
    x_1 = 2;
    x_1 = soma(1, x_1);

    x_1 = faz_teu_nome();
    resenha ((x_1 > 1) && !(x_1 < 1)) {
        x_1 = 3;
    }
    ramelou {
        {
            x_1 = (-20+30)*4*3/40;;;;; // teste de comentario
        }
    }
    explana(x_1);
    x_1 = faz_teu_nome();
    resenha ((x_1 > 1) && !(x_1 < 1))
        x_1 = 3;
    ramelou
        x_1 = (-20+30)*12/40;;;;;

    explana(x_1);
    enquanto ((x_1 > 1) || (x_1 == 1)) {x_1 = x_1 - 1;explana(x_1);}
```