

# TOY

A written programming language

André De Marco Toyama

Engenharia da Computação - INSPER

# Proposta

- Criar uma linguagem de programação em que as operações são escritas por extenso. Podendo ter a leitura similar a um texto. Assim como um compilador para executar programas de extensão “.toy”.

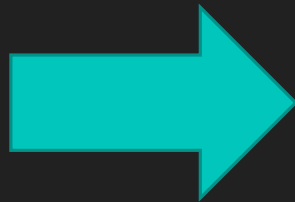
# Proposta

## Exemplo:

- $X = 1 + 2$

- If  $(x > 10)$

- $Y = 2/4 * 10$



- X is equal to 1 plus 2

- If x is greater than 10

- Y is equal to 2 divided by 4 times 10

# Recursos da Linguagem

## ○ Operadores

○ + : Plus

○ - : Minus

○ \* : Times

○ / : Divided by

○ = : is Equal to

○ > : is greater than

○ < : is less than

○ And

○ Or

○ Not

# Recursos da Linguagem

- **Tipos**

- Booleanos
- Inteiros

- **Impressão**

- Print valor

- **Input**

- variable is equal to Input

- **Declaração de variável**

- Integer x
- Boolean y

- **Atribuição**

- X is equal to 10
- Y is equal to True

# Recursos da Linguagem

## ○ Condicional

```
If x is greater than 10 {  
    y is equal to true  
  
}  
else {  
    y is equal to false  
  
}
```

## ○ Repetição

```
while x is less than 1000 {  
    x is equal to x times 2  
  
}
```

# Recursos da Linguagem

- Funções sem retorno

- Declaração

- ```
void soma( integer x, integer y)
```

- ```
    print x plus y
```

- ```
end void
```

- Chamada

- ```
soma( 5, 3)
```

# Recursos da Linguagem

## ○ Funções com retorno

### ○ Declaração

```
function soma( integer x, integer y) as integer
```

```
    soma is equal to x plus y
```

```
end function
```

### ○ Chamada

Variable is equal to soma( 5, 3)

Para retornar um valor, ele deve ser atribuído a uma variável com o nome da função, sem necessidade de ser declarada previamente



# Melhorias

- Implementar outros tipos de dados
- Implementar novos operadores
  - $\geq$
  - $\leq$
  - $+=$
  - $-=$
  - $\ast=$
  - $/=$
  - $\%$