

IDJL - Raphael's Language

# Problemas encontrados

A ideia principal na criação desta linguagem foi a dificuldade visual encontrada em editores de texto simples, como o GNU Nano e o Vim. Nestes editores, não há nenhum tipo de cor por padrão, o que dificulta muito quando estamos procurando por coisas muito específicas, como um if ou uma declaração de variável.

Para linguagens que não possuem tipagem, como o python, a identificação pode ficar ainda pior

# Exemplo

```
function fibonacci(n as integer) as integer
dim flag as boolean
flag = false
if n = 0 then
fibonacci = 1
flag = true
end if
if n = 1 then
fibonacci = 1
flag = true
end if
if flag = false then
fibonacci = fibonacci(n-2) + fibonacci(n-1)
end if
end function
sub Main()
print fibonacci(5)
dim x as integer
x = 0
while x < 3 then
x = x + 1
print x
wend
end sub
```

# Proposta

Para tornar principais pontos do código mais legível, a proposta é marcar o início de linhas com alguns marcadores para facilitar os 'Ctrl+f's mais rápidos:

@ - Criação e modificação de uma variável

# - Criação de uma cadeia condicional

\$ - Criação de um loop

^ - Criação de uma função

# Resultado

```
^ function fibonacci(n as integer) as integer
@ dim flag as boolean
flag = false
# if n = 0 then
fibonacci = 1
flag = true
end if
# if n = 1 then
fibonacci = 1
flag = true
end if
# if flag = false then
fibonacci = fibonacci(n-2) + fibonacci(n-1)
end if
end function
^ sub Main()
print fibonacci(5)
@ dim x as integer
x = 0
$ while x < 3 then
x = x + 1
print x
wend
end sub
```

# Aplicações

A idéia no exemplo mostrado é de que, utilizando um editor "cru", procurar por pontos principais do código seja mais rápido, principalmente em códigos muito grandes. No exemplo anterior, para procurar todas os ifs que dependem da variável **flag**, basta procurar por 'Ctrl+F' (ou equivalente no seu editor) e '# if flag'.