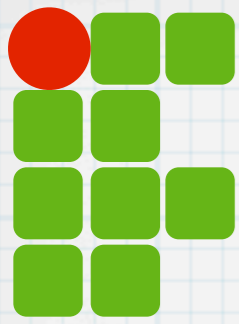


INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE

Programação de Computadores

E se?

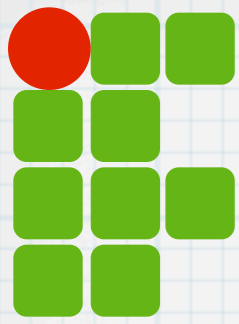
Copyright © 2013 IFRN



O que veremos hoje?

- * Fluxo de execução
- * Condição de processamento
- * Valores booleanos
- * A instrução/operação `if...then...else...`
- * Operadores relacionais (comparação)
 - * igual, diferente, maior, menor, maior-igual, menor-igual
- * Operadores lógicos
 - * ou, e e negação
- * Aninhamento de `if`
- * Exercícios

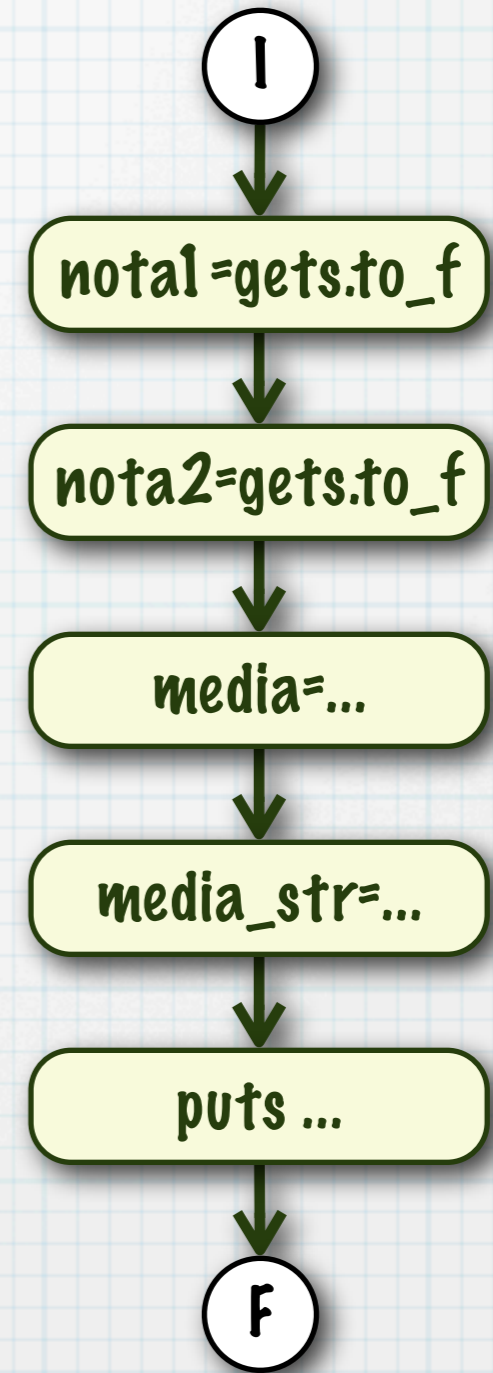


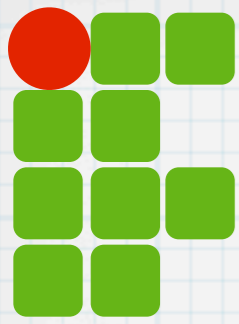


Fluxo de processamento

- * Um programa contém uma sequência de operações/instruções
- * Uma processada por vez
- * Todas são processadas
- * Ao final o programa termina
- * Considere o cálculo de uma média ponderada

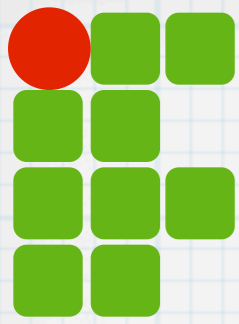
```
nota1 = gets.to_f
nota2 = gets.to_f
media = (nota1*2+nota2*3)/5
media_str = "%.1f" % media
puts media_str
```





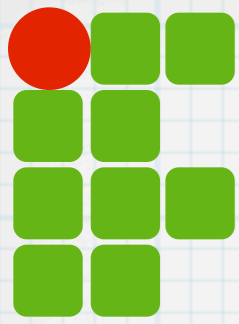
Condição para processamento

- * Uma (ou mais) operação(ões) só deve(m) ser processada(s) em determinados casos
 - * Condição
 - * Valor lógico (“booleano”)
 - * Verdadeiro (true) OU Falso (false)
- * Exemplo
 - * Informar se um aluno foi aprovado ou não
 - * Ler notas e calcular a média
 - * SE média igual ou maior do que 6.0: APROVADO
 - * SENÃO: NÃO APROVADO



Condição para processamento

- * **Mostrar que aluno foi aprovado**
 - * **APENAS** se a média foi atingida
 - * Se $media \geq 6.0$
 - * mostrar "ALUNO APROVADO"
- * **O que o programa mostra DEPENDE de valor(es) de variável(eis)**
 - * **Processamento condicional**
- * **As operação(ões)/instrução(ões) pode(m) ou não ser processada(s)**



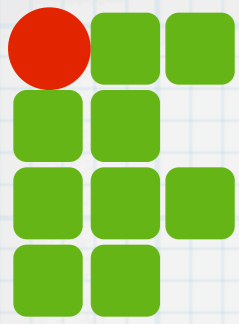
Fluxo de processamento

* Se..Senão

```
if (condição) then  
  instV1  
  ...  
else  
  instF1  
  ...  
end
```

Código processado apenas se a condição for verdadeira

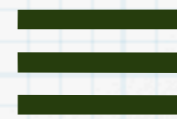
Código processado apenas se a condição for falsa



Fluxo de processamento

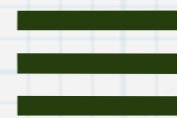
* Se..Senão

```
if (condição) then  
  instV1  
  ...  
else  
  instF1  
  ...  
end
```



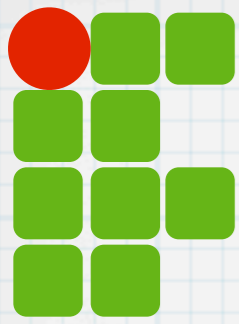
Se condição verdadeira

```
instV1  
...
```



Se condição falsa

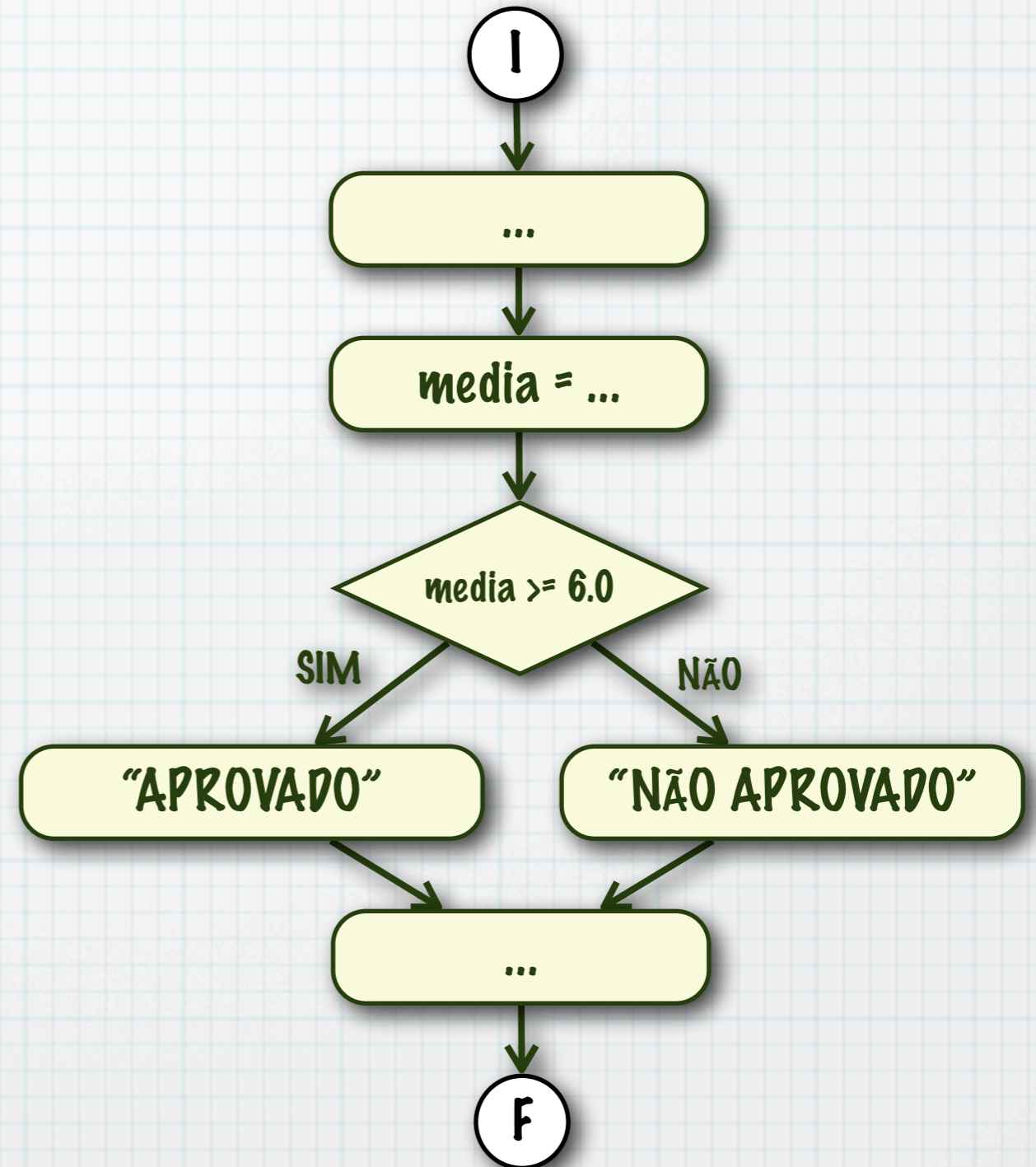
```
instF1  
...
```

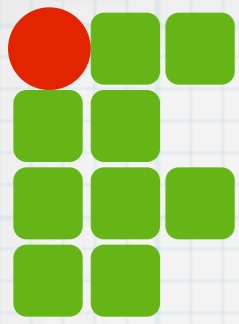


Fluxo de processamento

```
...  
if (media >= 6.0) then  
  puts "APROVADO"  
else  
  puts "NÃO APROVADO"  
end  
...
```

**Apenas UM
fluxo será
processado**

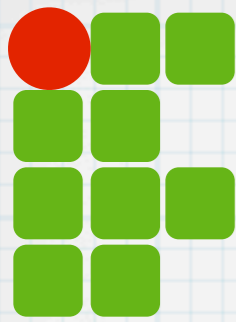




Exemplo

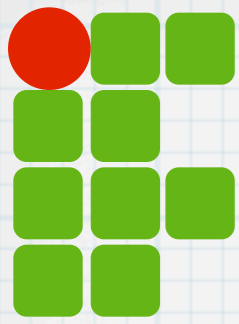
* Informar se um nome está em maiúsculo

```
nome = gets.chomp
if (nome.upcase == nome) then
  puts "O nome esta todo em maiusculo"
else
  puts "O nome nao esta todo em maiusculo"
end
```



Operadores relacionais

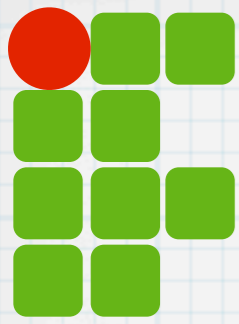
Operação	símbolo	Exemplo
Igual	<code>==</code>	<code>if (a == 10)</code>
Maior	<code>></code>	<code>if (vendas > meta)</code>
Menor	<code><</code>	<code>if (media < 6.0)</code>
Maior-igual	<code>>=</code>	<code>if (salario >= 600.00)</code>
Menor-igual	<code><=</code>	<code>if (x <= y)</code>
Diferente	<code>!=</code>	<code>if (a != b)</code>



Resultado do if

- * If como expressão
 - * Retorna um valor
 - * Pode ser usado em expressões

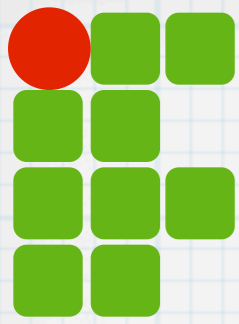
```
resultado = if (media >= 6.0) then "Aprovado" else "Reprovado" end
```



Resultado do if

- * If como expressão
 - * Retorna um valor
 - * Pode ser usado em expressões

```
resultado = if (media >= 6.0) then "Aprovado" else "Reprovado" end
```



Resultado do if

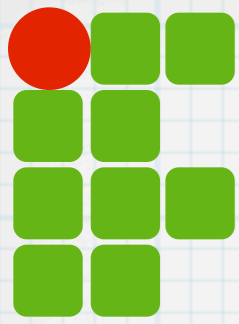
* If como expressão

- * Retorna um valor
- * Pode ser usado em expressões

Quando media for maior-igual a 6.0 ($media \geq 6.0$)

```
resultado = if (media >= 6.0) then "Aprovado" else "Reprovado" end
```

Quando media não for maior-igual a 6.0 ($\text{not}(media \geq 6.0)$)



Operação if

* Caso especial onde não existe o SENÃO (else)

```
if (condição) then
```

```
  inst
```

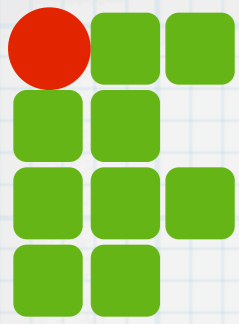
```
  inst
```

```
  inst
```

```
  . . .
```

```
end
```

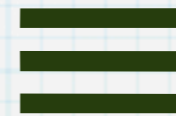
**Código
processado
apenas se a
condição for
verdadeira**



Operação if

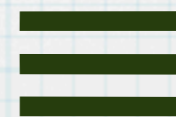
* Caso especial on de não existe o SENÃO (else)

```
if (condição) then  
  inst  
  inst  
  inst  
  ...  
end
```



Se condição verdadeira

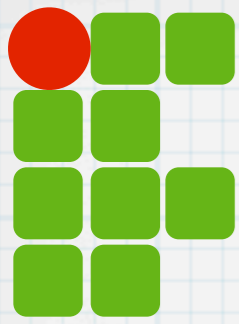
```
instV1  
...
```



Se condição falsa

```
nil
```

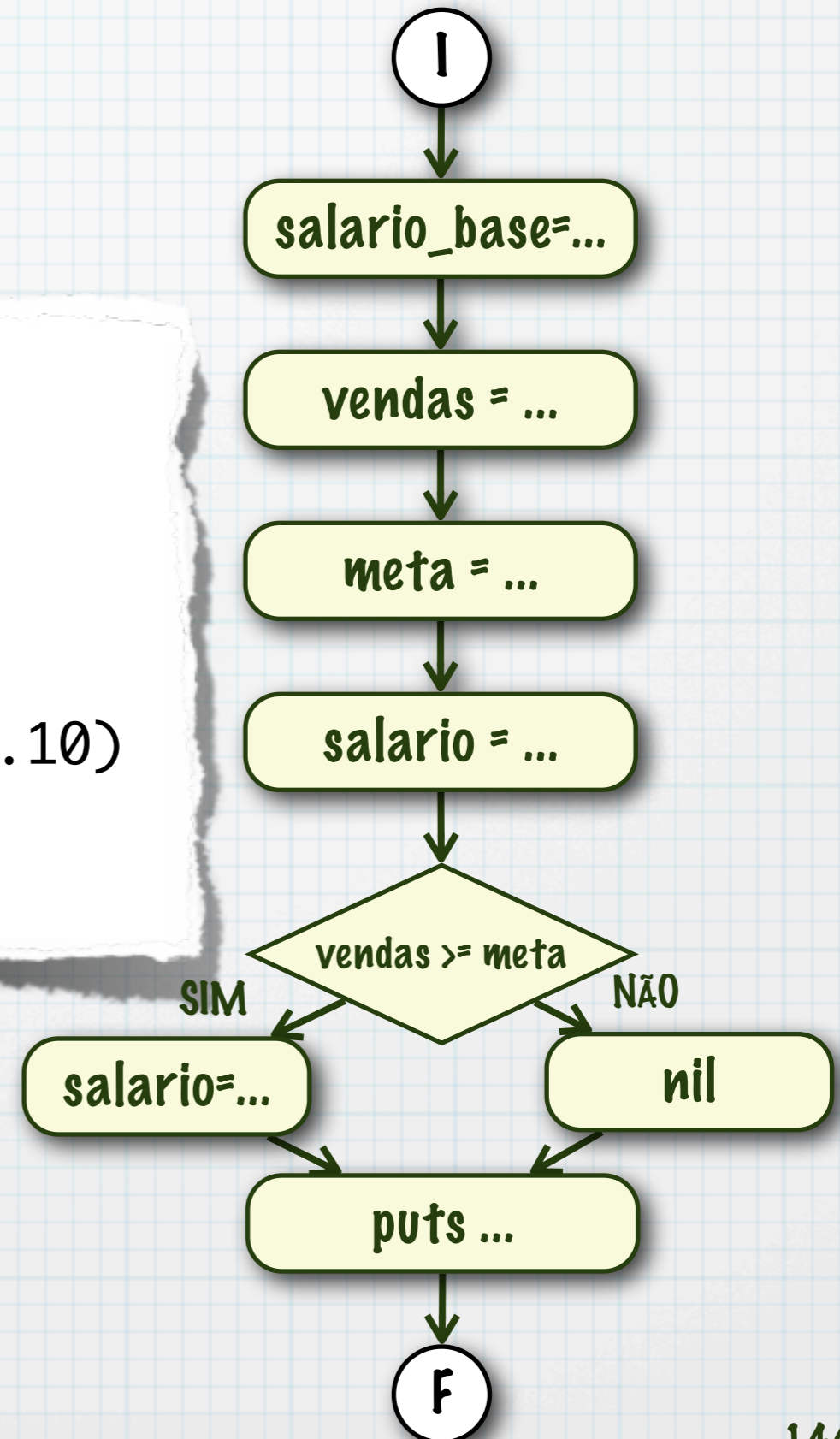
nil significa nada

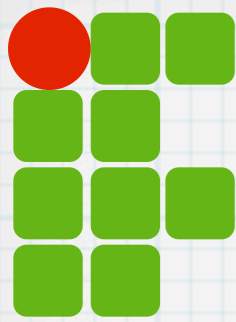


Operação if

* Cálculo do salário do mês

```
salario_base = gets.to_f
vendas = gets.to_f
meta = gets.to_f
salario = salario_base + vendas*0.05
if (vendas >= meta) then
  salario = salario + ((vendas-meta)*0.10)
end
puts "%.2f" % salario
```





Operação if

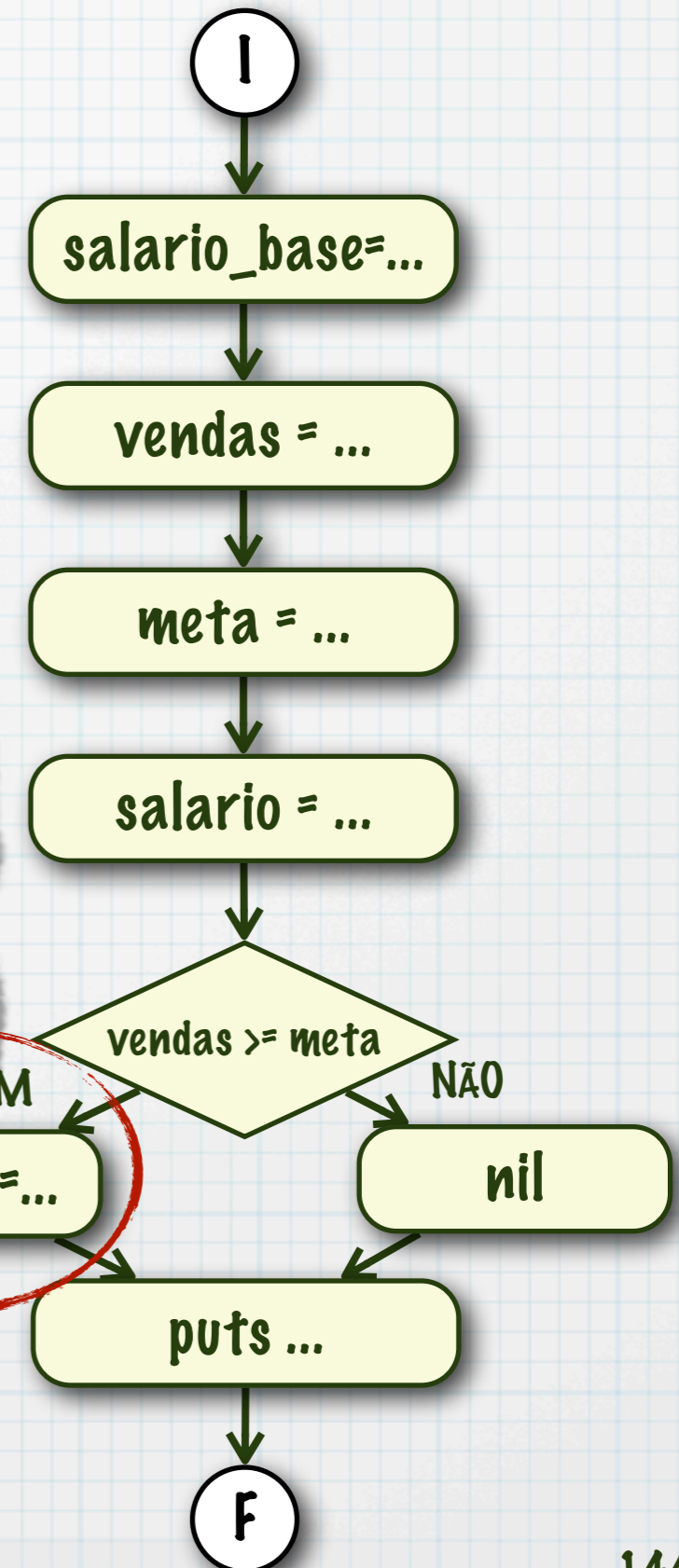
* Cálculo do salário do mês

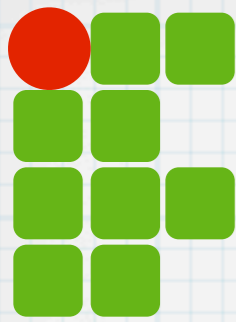
```
salario_base = gets.to_f  
vendas = gets.to_f  
meta = gets.to_f  
salario = salario_base + vendas*0.05
```

```
if (vendas >= meta) then  
  salario = salario + ((vendas-meta)*0.10)  
end
```

```
puts "%.2f" % salario
```

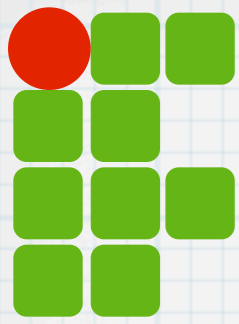
**processamento
condicional**





Operadores lógicos

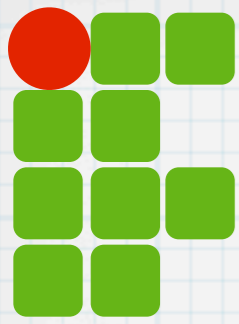
Operação	símbolo	Exemplo
E (and)	and	if (a == 10 and b < 20)
	&&	
Ou (or)	or	if (vendas > meta or sal < 1000.00)
Negação (not)	not	if (!(media == 6.0 and nota1 < 8.0))
	!	



Exemplo

* Informar se nota não é válida

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 < 0.0 or nota1 > 10.0) then
  puts "A nota 1 nao e valida!"
end
if (nota2 < 0.0 or nota2 > 10.0) then
  puts "A nota 2 nao e valida!"
end
...
```

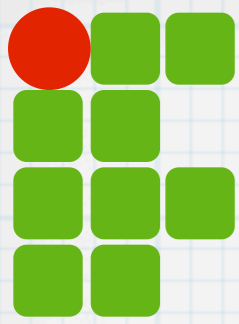


Exemplo

* Informar se nota não é válida

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 < 0.0 or nota1 > 10.0) then
  puts "A nota 1 nao e valida!"
end
if (nota2 < 0.0 or nota2 > 10.0) then
  puts "A nota 2 nao e valida!"
end
...
```

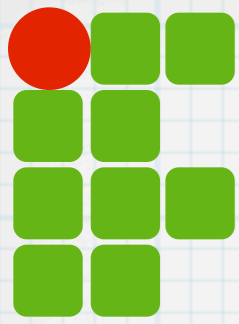
O **or** indica que, para a nota não ser válida, basta um dos dois ser verdadeiro



Exemplo

* Informar se nota é válida

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 >= 0.0 and nota1 <= 10.0) then
  puts "A nota 1 e valida!"
end
if (nota2 >= 0.0 and nota2 <= 10.0) then
  puts "A nota 2 e valida!"
end
```

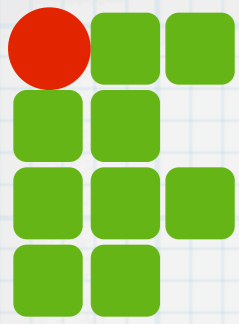


Exemplo

* Informar se nota é válida

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 >= 0.0 and nota1 <= 10.0) then
  puts "A nota 1 e valida!"
end
if (nota2 >= 0.0 and nota2 <= 10.0) then
  puts "A nota 2 e valida!"
end
```

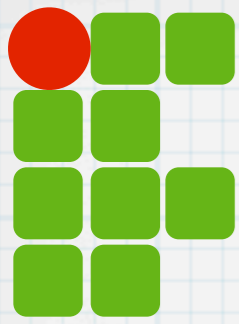
O **and** indica que,
para a nota ser
válida, os dois lados
devem ser verdadeiro



Exemplo

* Ainda se nota não é válida

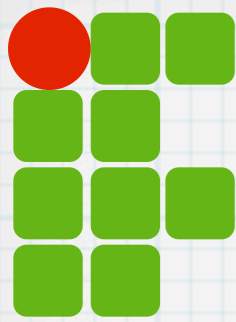
```
nota1 = gets.to_f
nota2 = gets.to_f
if (not(nota1 >= 0.0 and nota1 <= 10.0)) then
  puts "A nota 1 nao e valida!"
end
if (not(nota2 >= 0.0 and nota2 <= 10.0)) then
  puts "A nota 2 nao e valida!"
end
```



Exemplo

* Ainda se nota não é válida

```
nota1 = gets.to_f
nota2 = gets.to_f
if (not(nota1 >= 0.0 and nota1 <= 10.0)) then
  puts "A nota 1 nao e valida!"
end
if (not(nota2 >= 0.0 and nota2 <= 10.0)) then
  puts "A nota 2 nao e valida!"
end
```

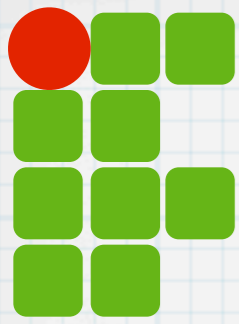



Exemplo

* Cálculo do salário

- * Se atingir meta bônus de 10% sobre o que passou
- * Se salário total maior que R\$ 1200,00 retenção de imposto de 15% sobre o salário base

```
salario_base = gets.to_f
vendas = gets.to_f
meta = gets.to_f
salario = salario_base + vendas*0.05
if (vendas >= meta) then
  salario = salario + ((venda-meta)*0.10)
end
if (salario > 1200.00)
  salario = salario - (salario_base * 0.15)
end
puts "%.2f" % salario
```



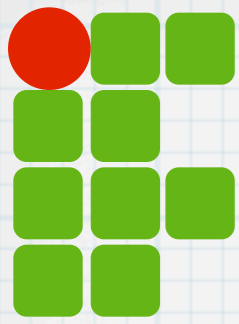
Exemplo - Cálculo de comissão

```
salario_base = gets.to_f
vendas = gets.to_f
meta = gets.to_f
bonus = if (vendas >= meta) then (venda-meta)*0.10 else 0 end
salario_bruto = salario_base + vendas*0.05 + bonus
imposto = if (salario_bruto > 1200.00) then (salario_base * 0.15) else 0 end
salario = salario_bruto - imposto
puts "%.2f" % salario
```

if como
expressão

if como
controle
de fluxo

```
salario_base = gets.to_f
vendas = gets.to_f
meta = gets.to_f
salario = salario_base + vendas*0.05
if (vendas >= meta) then
  salario = salario + ((venda-meta)*0.10)
end
if (salario > 1200.00)
  salario = salario - (salario_base * 0.15)
end
puts "%.2f" % salario
```

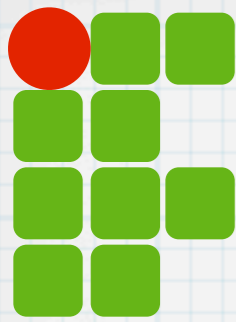


Múltiplos fluxos

* Senão_Se (elsif)

- * Várias condições
- * Apenas UM bloco será executado
- * Primeira condição que for VERDADEIRA

```
if (condição) then
  inst
  ...
elsif (condição) then
  inst
  ...
elsif (condição) then
  inst
  ...
else
  inst
  ...
end
```



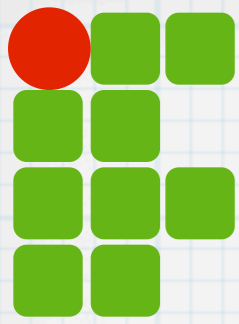
Múltiplos fluxos

* Senão_Se (elsif)

- * Várias condições
- * Apenas UM bloco será executado
- * Primeira condição que for VERDADEIRA

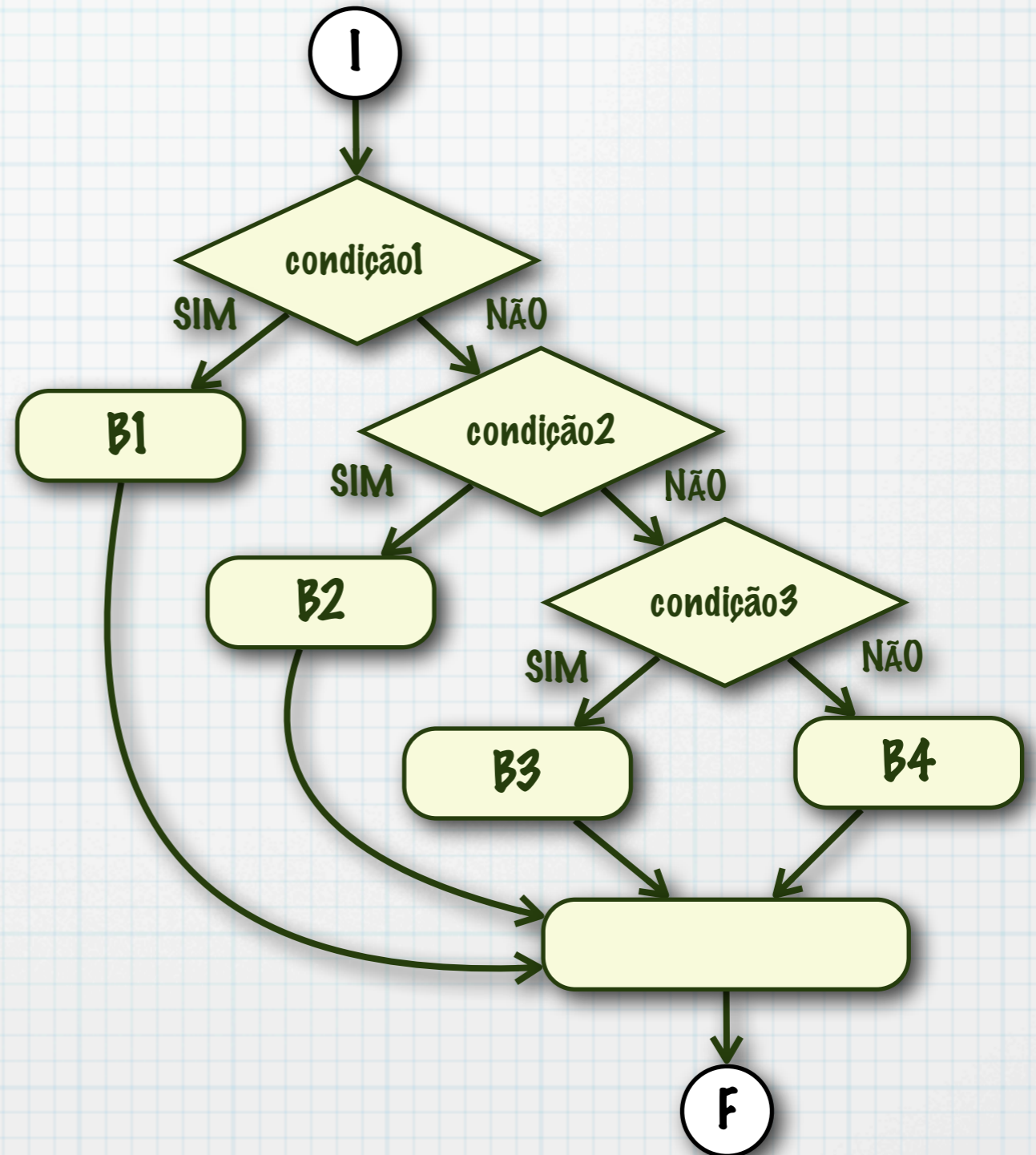
Apenas UM
entre B1, B2, B3
e B4 irá
executar

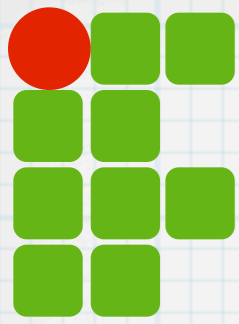
```
if (condição) then
  inst B1
  ...
elsif (condição) then
  inst B2
  ...
elsif (condição) then
  inst B3
  ...
else
  inst B4
  ...
end
```



Múltiplos fluxos

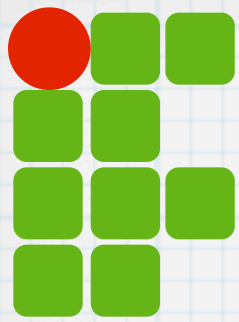
```
if (condição) then
  inst
  ...
elsif (condição) then
  inst
  ...
elsif (condição) then
  inst
  ...
else
  inst
  ...
end
```





Exemplo - Quantidade de raízes

```
...  
delta = b * b - 4 * a * c  
if (delta == 0) then  
    numero_raizes_reais = 1  
elseif (delta > 0) then  
    numero_raizes_reais = 2  
else  
    numero_raizes_reais = 0  
end  
...
```



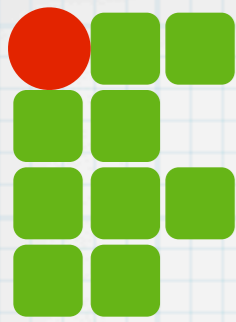
Exemplo - Quantidade de raízes

```
...  
delta = b * b - 4 * a * c  
if (delta == 0) then  
  numero_raizes_reais = 1  
elsif (delta > 0) then  
  numero_raizes_reais = 2  
else  
  numero_raizes_reais = 0  
end  
...
```

if como
controle
de fluxo

if como
expressão

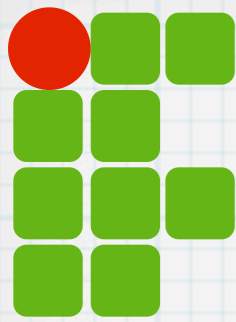
```
...  
delta = b * b - 4 * a * c  
numero_raizes_reais = if (delta == 0) then 1  
  elsif (delta > 0) then 2  
  else 0  
  end  
...
```



Exemplo

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 < 0 or nota1 > 10) then
  puts "Primeira nota nao e valida"
  exit
end
if (nota2 < 0 or nota2 > 10) then
  puts "Segunda nota nao e valida"
  exit
end
media = (nota1*2 + nota2*3)/5
resultado = "sem estado"
if (media < 2.0) then
  resultado = "reprovado"
elsif (media < 6.0) then
  resultado = "em recuperacao"
else
  resultado = "aprovado"
end

puts "O aluno esta " + resultado
puts "A media foi #{media}"
```

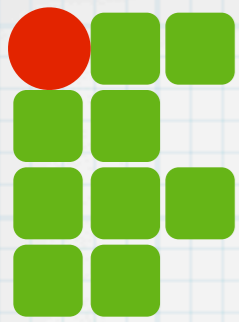



Exemplo

A operação **exit**
sai do programa
sem executar o que
vem depois

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 < 0 or nota1 > 10) then
  puts "Primeira nota nao e valida"
  exit
end
if (nota2 < 0 or nota2 > 10) then
  puts "Segunda nota nao e valida"
  exit
end
media = (nota1*2 + nota2*3)/5
resultado = "sem estado"
if (media < 2.0) then
  resultado = "reprovado"
elsif (media < 6.0) then
  resultado = "em recuperacao"
else
  resultado = "aprovado"
end

puts "O aluno esta " + resultado
puts "A media foi #{media}"
```

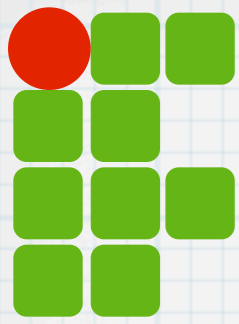


Exemplo

A operação **exit** sai do programa sem executar o que vem depois

DICA: tentem colocar as saídas (**puts** e **print**) em pontos específicos do programa

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 < 0 or nota1 > 10) then
  puts "Primeira nota nao e valida"
  exit
end
if (nota2 < 0 or nota2 > 10) then
  puts "Segunda nota nao e valida"
  exit
end
media = (nota1*2 + nota2*3)/5
resultado = "sem estado"
if (media < 2.0) then
  resultado = "reprovado"
elsif (media < 6.0) then
  resultado = "em recuperacao"
else
  resultado = "aprovado"
end
puts "O aluno esta " + resultado
puts "A media foi #{media}"
```



Recuo de texto

* Indentação, endentação, indentação

* do inglês indent

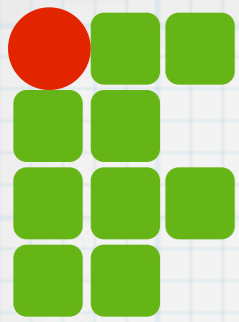
* Espaços colocados no início da linha para identificar dependência com linhas anteriores

* Facilita visualização do código

* Elementos hierarquisados

* O que “pertence” ao if?

```
if (condição) then
  inst
  inst
  inst
  ...
end
```



Recuo de texto

* Indentação, endentação, indentação

* do inglês indent

* Espaços colocados no início da linha para identificar dependência com linhas anteriores

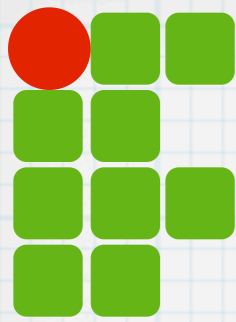
* Facilita visualização do código

* Elementos hierarquisados

* O que "pertence" ao if?

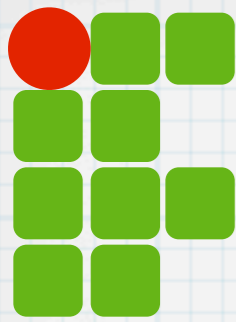
Indentação: visualmente percebe-se que as operações são dependentes do `if`

```
if (condição) then
  inst
  inst
  inst
  ...
end
```



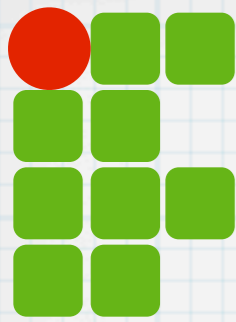
Exemplo

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 >= 0.0 and nota1 <= 10.0 and
    nota2 >= 0.0 and nota2 <= 10.0) then
media = (nota1*2 + nota2*3)/5
resultado = "sem estado"
if (media < 2.0) then
resultado = "reprovado"
elsif (media < 6.0) then
resultado = "em recuperacao"
else
resultado = "aprovado"
end
puts "O aluno esta " + resultado
puts "A media foi #{media}"
else
if (nota1 < 0.0 or nota1 > 10.0) then
puts "Primeira nota nao e valida"
end
if (nota2 < 0.0 or nota2 > 10.0) then
puts "Segunda nota nao e valida"
end
end
end
```



Exemplo

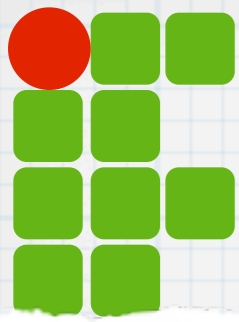
```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 >= 0.0 and nota1 <= 10.0 and
    nota2 >= 0.0 and nota2 <= 10.0) then
  media = (nota1*2 + nota2*3)/5
  resultado = "sem estado"
  if (media < 2.0) then
    resultado = "reprovado"
  elsif (media < 6.0) then
    resultado = "em recuperacao"
  else
    resultado = "aprovado"
  end
  puts "O aluno esta " + resultado
  puts "A media foi #{media}"
else
  if (nota1 < 0.0 or nota1 > 10.0) then
    puts "Primeira nota nao e valida"
  end
  if (nota2 < 0.0 or nota2 > 10.0) then
    puts "Segunda nota nao e valida"
  end
end
end
```



Exemplo

Indentação

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 >= 0.0 and nota1 <= 10.0 and
    nota2 >= 0.0 and nota2 <= 10.0) then
  media = (nota1*2 + nota2*3)/5
  resultado = "sem estado"
  if (media < 2.0) then
    resultado = "reprovado"
  elsif (media < 6.0) then
    resultado = "em recuperacao"
  else
    resultado = "aprovado"
  end
  puts "O aluno esta " + resultado
  puts "A media foi #{media}"
else
  if (nota1 < 0.0 or nota1 > 10.0) then
    puts "Primeira nota nao e valida"
  end
  if (nota2 < 0.0 or nota2 > 10.0) then
    puts "Segunda nota nao e valida"
  end
end
end
```



Exemplo- Situação do aluno

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 >= 0 and nota1 <= 10 and
    nota2 >= 0 and nota2 <= 10) then
  media = (nota1*2 + nota2*3)/5
  Resultado =
    if (media < 2.0) then "reprovado"
    elsif (media < 6.0) then "em recuperacao"
    else "aprovado" end
  puts "O aluno esta " + Resultado
  puts "A media foi #{media}"
elsif (nota1 < 0 or nota1 > 10) then
  puts "Primeira nota nao e valida"
else
  puts "Segunda nota nao e valida"
end
```

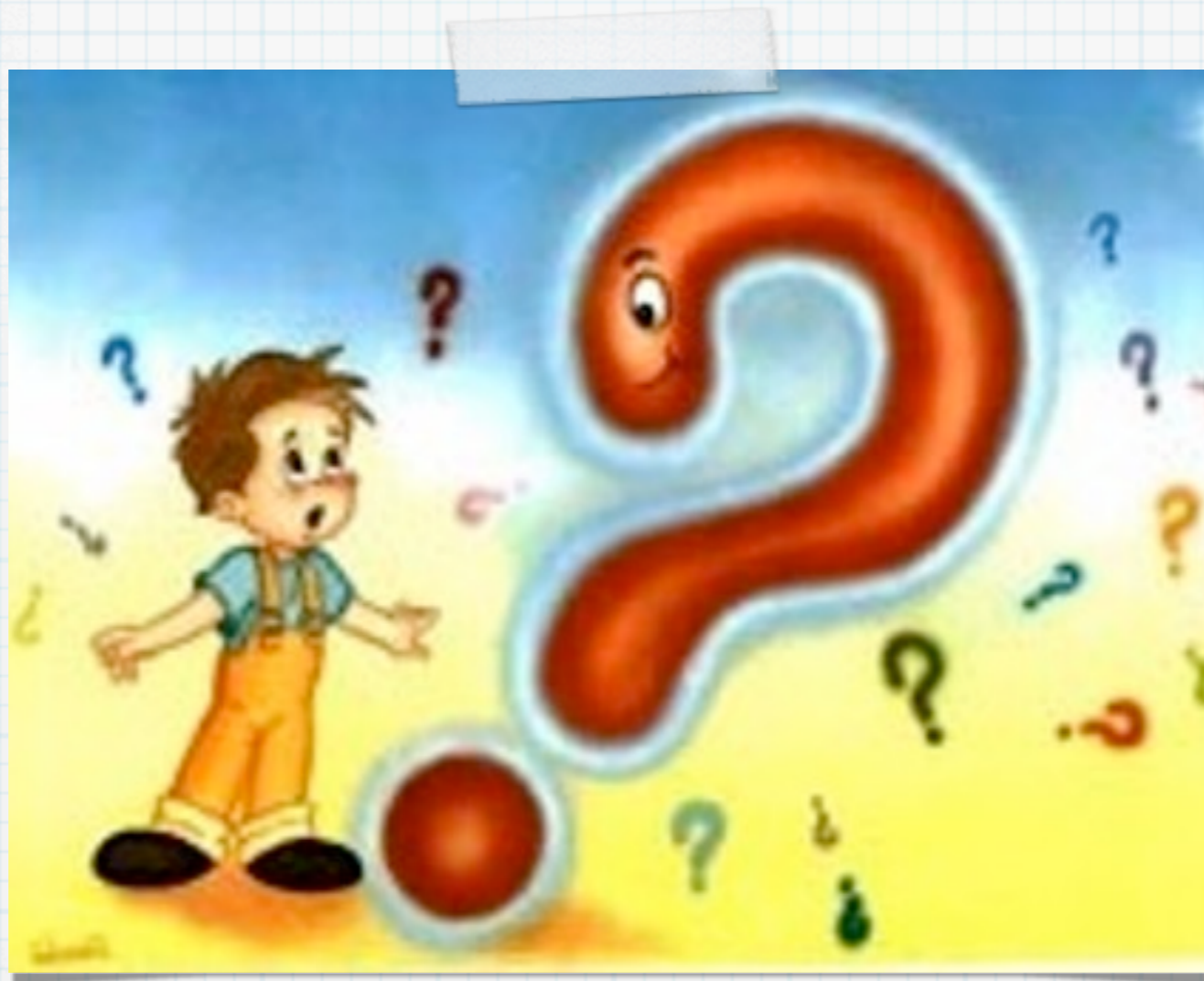
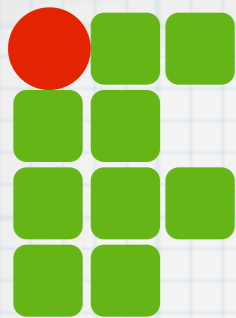
↑

if como
expressão

→

if como
controle
de fluxo

```
nota1 = gets.to_f
nota2 = gets.to_f
if (nota1 >= 0.0 and nota1 <= 10.0 and
    nota2 >= 0.0 and nota2 <= 10.0) then
  media = (nota1*2 + nota2*3)/5
  resultado = "sem estado"
  if (media < 2.0) then
    resultado = "reprovado"
  elsif (media < 6.0) then
    resultado = "em recuperacao"
  else
    resultado = "aprovado"
  end
  puts "O aluno esta " + resultado
  puts "A media foi #{media}"
else
  if (nota1 < 0.0 or nota1 > 10.0) then
    puts "Primeira nota nao e valida"
  end
  if (nota2 < 0.0 or nota2 > 10.0) then
    puts "Segunda nota nao e valida"
  end
end
```

Dúvidas?