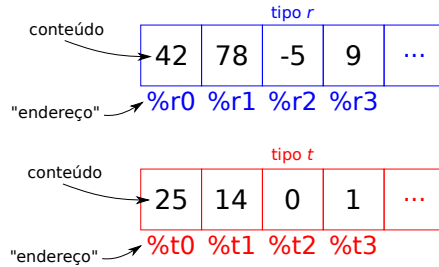


Compiladores

prof. Ricardo Oliveira

Documentação do *Assembly* Raposeitor (resumido)



Instruções de Entrada/Saída	
<code>printf str</code>	imprime frase
<code>printv op</code>	imprime valor
<code>read reg</code>	lê valor
Instrução de Atribuição	
<code>mov rdest, op</code>	$rdest \leftarrow op$
Instruções de Operações Aritméticas	
<code>add rdest, op1, op2</code>	$rdest \leftarrow op1 + op2$
<code>sub rdest, op1, op2</code>	$rdest \leftarrow op1 - op2$
<code>mult rdest, op1, op2</code>	$rdest \leftarrow op1 \times op2$
<code>div rdest, op1, op2</code>	$rdest \leftarrow op1 / op2$
<code>mod rdest, op1, op2</code>	$rdest \leftarrow op1 \% op2$
Instruções de Operações Lógicas	
<code>not rdest, op</code>	$rdest \leftarrow \text{“não” } op$
<code>or rdest, op1, op2</code>	$rdest \leftarrow op1 \text{ “ou” } op2$
<code>and rdest, op1, op2</code>	$rdest \leftarrow op1 \text{ “e” } op2$
Instruções de Comparação	
<code>equal rdest, op1, op2</code>	$rdest \leftarrow op1 = op2$
<code>diff rdest, op1, op2</code>	$rdest \leftarrow op1 \neq op2$
<code>less rdest, op1, op2</code>	$rdest \leftarrow op1 < op2$
<code>lesseq rdest, op1, op2</code>	$rdest \leftarrow op1 \leq op2$
<code>greater rdest, op1, op2</code>	$rdest \leftarrow op1 > op2$
<code>greatereq rdest, op1, op2</code>	$rdest \leftarrow op1 \geq op2$
Instruções de Acesso Indireto	
<code>load rdest, desl(base)</code>	$rdest \leftarrow \%r(base+desl)$
<code>store orig, desl(base)</code>	$\%r(base+desl) \leftarrow orig$
Instruções de Fluxo	
<code>label rotulo</code>	rotula
<code>jump rotulo</code>	<i>go-to</i> incondicional
<code>jf op, rotulo</code>	<i>jump if false</i>
<code>jt op, rotulo</code>	<i>jump if true</i>

Comentários com ;

Executar com os interpretadores dados (em C++ ou em Python3)