

Quádruplas

Operações	addr1	addr2	addr3	addr4	Descrição
opADD	add	\$r1	\$r2	\$r3	soma entre r1 e r2 ($r3 = r1 + r2$)
opSUB	sub	\$r1	\$r2	\$r3	subtração entre r1 e r2 ($r3 = r1 - r2$)
opMULT	mult	\$r1	\$r2	\$r3	multiplicação entre r1 e r2 ($r3 = r1 * r2$)
opDIV	div	\$r1	\$r2	\$r3	divisão entre r1 e r2 ($r3 = r1 / r2$)
opLT	bgeq	\$r1	\$r2	L0	if $r1 < r2$ não desvia para L0
opLEQUAL	bgt	\$r1	\$r2	L0	if $r1 \leq r2$ não desvia para L0
opGT	bleq	\$r1	\$r2	L0	if $r1 > r2$ não desvia para L0
opGREQUAL	blt	\$r1	\$r2	L0	if $r1 \geq r2$ não desvia para L0
opIGL	bne	\$r1	\$r2	L0	if $r1 == r2$ não desvia para L0
opDIF	beq	\$r1	\$r2	L0	if $r1 != r2$ não desvia para L0
opASSIGN	atrib	\$r1	\$r2	-	atribui r2 a r1 ($r1=r2$)
opALLOC	alloc	nome var	memLoc	Escopo	aloca var na posição memLoc da memória
opIMMED	immed	\$r1	imediato	-	coloca o imediato em um reg ($r1 = \text{imediato}$)
opLOAD	load	\$r1	nome var	memLoc	coloca um valor da mem em um reg ($r1 = \text{memLoc}$)
opSTORE	store	nome var	memLoc/posVec	\$r1	coloca um valor de um reg na mem ($\text{memLoc} = r1$)
opVEC	vec	\$r1	nome var	posVec	atribui um vetor a um reg ($r1 = \text{var}[\text{posVec}]$)
opGOTO	goto	L0	-	-	pula para a posição da label L0 (j L0)
opRET	ret	\$r1	-	-	indica o reg que armazena o retorno da função
opFUN	fun	nome fun	memLoc	-	aloca fun na posição memLoc da memória
opEND	end	nome fun	-	-	indica o fim da função
opPARAM	param	\$r1	-	-	indica que r1 é um parametro
opCALL	call	\$r1	nome fun	qtd param	indica que r1 vai armazenar o retorno da função e possui qtd parametros
opARG	arg	nome arg	memLoc	nome fun	aloca o arg na posição memLoc da memória
opLAB	lab	L0	-	-	indica uma label (L0)
opHLT	hit	-	-	-	indica o fim do programa