



Home Page ► Meus cursos ► ACI ► Bibliografia ► ...bconjunto Linguagem de Máquina do Processador MIPS

## Navegação

Home Page

■ Minha página inicial

Páginas do site

Meu perfil

Meus cursos

ACI

Participantes

Bibliografia

? Texto de  
Arquitetura de  
Computadores 1

📅 Fórum de notícias

📁 ...bconjunto  
Linguagem de  
Máquina do  
Processador  
MIPS

Capítulo 1

Capítulo 2

Capítulo 3

Capítulo 4

Capítulo 5

## Configurações

Administração do curso

Minhas configurações de  
perfil

Fazer um programa, em linguagem C, que simule o comportamento de um tradutor de um subconjunto de instruções da linguagem de montagem (*assembly*) do processador MIPS. O subconjunto de instruções está mostrado na Tabela 1.

Tabela 1. Subconjunto de Instruções do Processador MIPS.

Instrução	Tipo	Significado	Opcode (em hexa)	Funct (em hexa)
add	R	$R[rd] = R[rs] + R[rt]$	0	20
sub	R	$R[rd] = R[rs] - R[rt]$	0	22
and	R	$R[rd] = R[rs] \& R[rt]$	0	24
or	R	$R[rd] = R[rs]   R[rt]$	0	25
sll	R	$R[rd] = R[rs] \ll \text{shamt}$	0	0
srl	R	$R[rd] = R[rs] \gg \text{shamt}$	0	2
slt	R	if ( $R[rs] < R[rt]$ ) $R[rd] = 1$ else $R[rd] = 0$	0	2a
jr	R	$PC = R[rs]$	0	8
j	J	$PC = ((PC+4)[31-28])   (\text{addr} \ll 2)$	2	N/A
jal	J	$R[31] = (PC+4)$ $PC = ((PC+4)[31-28])   (\text{addr} \ll 2)$	3	N/A
lw	I	$R[rt] = M[R[rs] + \text{SignExt}(\text{Imm})]$	23	N/A
sw	I	$M[R[rs] + \text{SignExt}(\text{Imm})] = R[rt]$	2b	N/A
addi	I	$R[rt] = R[rs] + \text{SignExt}(\text{Imm})$	8	N/A
bne	I	if ( $R[rs] \neq R[rt]$ ) $PC = PC+4 + (\text{SignExt}(\text{Imm}) \ll 2)$	5	N/A
beq	I	if ( $R[rs] == R[rt]$ ) $PC = PC+4 + (\text{SignExt}(\text{Imm}) \ll 2)$	4	N/A

O programa deve receber como entrada um arquivo contendo o código *assembly* de um programa. Ao final da tradução deve ser gerado um arquivo com o código binário associado ao programa de entrada. O código binário é uma cadeia de 0s e 1s que corresponde a cada instrução. Além disso, considere uma segunda opção da seguinte forma, através de inserção de um parâmetro na linha de comando indique que o arquivo de saída deve mostrar uma cadeia de 32 bits por linha. A Figura 1 ilustra o processo de tradução de um programa *assembly* para um programa em código binário, com a segunda opção.

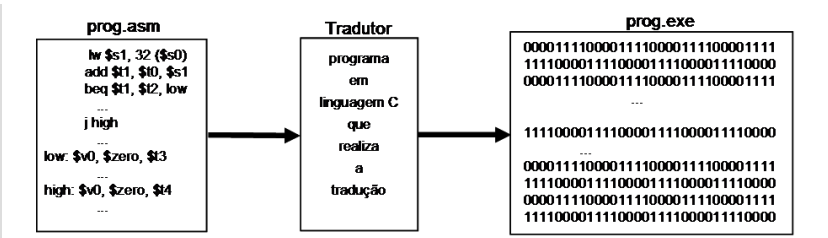


Figura 1. Processo de Tradução.

A Tabela 2 mostra o banco de registradores do processador MIPS, identificando cada um dos registradores e seus significados.

Tabela 2. Banco de Registradores do Processador MIPS.

Nome	Número do Registrador	Utilização	Preservado?
\$zero	0	Valor da constante 0	Não se aplica
\$at	1	Reservado para o montador	Não se aplica
\$v0-\$v1	2-3	Armazenar resultados e avaliar expressões	Não
\$a0-\$a3	4-7	Argumentos de funções	Sim
\$t0-\$t7	8-15	Temporários	Não
\$s0-\$s7	16-23	Salvos	Sim
\$t8-\$t9	24-25	Temporários	Não
\$k0-\$k1	26-27	Reservado para o SO	Não se aplica
\$gp	28	Global pointer	Sim
\$sp	29	Stack pointer	Sim
\$fp	30	Frame pointer	Sim
\$ra	31	Return address - endereço de retorno de procedimento	Sim

Pontos extras (podem ser acrescentados para aumentar a média, lembrando que o trabalho representa 1,0 na média) serão dados para os grupos que completarem a tarefa.

A entrega do trabalho deve ser feita pela plataforma moodle, em um arquivo.zip contendo:

- a) O código fonte do tradutor; com todos os arquivos associados;
- b) Os arquivos fonte e binário do programas em *assembly* usados para teste;
- c) Um arquivo .ppt com uma apresentação sobre o projeto.

A apresentação do projeto será feita em sala de aula e todos os alunos do grupo serão avaliados pela participação na apresentação e pelas respostas dadas ao professor, que arguirá cada aluno sobre detalhes do projeto. Alunos que aparentarem não conhecer o projeto terão uma drástica redução na nota final do trabalho.

Lembro que o tema é o mesmo para todos os grupos, e que todos são encorajados a conversarem sobre os desafios de implementação, preferencialmente, pelo fórum da disciplina. Entretanto, códigos idênticos, com modificações de variáveis, ou sofrendo dos mesmos tipos de *bugs* (erros) serão considerados para uma avaliação mais rigorosa para detecção de plágio.

O grupo que terminar o projeto é fortemente aconselhado a completar o montador para suportar um conjunto de instruções mais amplo e preparar uma interface gráfica, que auxiliará futuras implementações nas demais disciplinas da área.

Disponível a partir de:	quinta, 16 maio 2013, 00:20
Data de entrega:	domingo, 30 junho 2013, 00:20

### Esboço do documento

Nenhum arquivo enviado ainda

Enviar arquivos

Você acessou como Diego Rocha (Sair)

