Universidade Federal de Minas Gerais

Software Básico

Trabalho Prático 2

Montador

Lucas Furtini Veado – 2013007609

Edson Roteia Araujo Junior – 2014004174

1. **Introdução**

Este trabalho consiste na implementação de um montador de dois passos conforme descrito no livro Organização Estruturada de Computadores, Andrews S. Tanenbaum (5ª Ed). O conjunto de instruções é o mesmo do trabalho prático um pois o montador deve gerar um arquivo binário como saída e este arquivo será usado como entrada no emulador feito no primeiro trabalho.

Um montador de 2 passos realiza duas passagens no arquivo fonte contendo as instruções. A primeira passagem tem como objetivo principal montar a tabela de símbolos, que contém o valor de todos os símbolos. Um símbolo é um rótulo ou um valor no qual é atribuído um nome simbólico por meio de uma pseudo-instrução. Quando a segunda passagem começa, todos os valores dos símbolos são conhecidos, não havendo nenhuma referência antecipada e cada declaração pode ser lida, montada e produzida. A função da passagem dois é gerar o programa-objeto e possivelmente imprimir a tabela de símbolos.

1. **Projeto**

A estrutura da tabela de símbolos é uma lista encadeada contendo cada símbolo encontrado no arquivo fonte bem como o ILC (Instruction Location Counter – contador de localização de instrução) que possui o valor do tamanho de cada instrução até o ponto em que o símbolo foi encontrado.

Para esse projeto a tabela de símbolos também guarda o nome de cada variável definida pela pseudo-instrução DW (Define Word), porem ao final da passagem um ele atualiza o ILC das variáveis para a primeira posição da memória após esta ter recebido todas as instruções, ou seja, as variáveis são salvas a partir da primeira posição após a última instrução ser incluída na memória.

O programa foi dividido em 2 módulos mais o principal. São:

**- Montador:** módulo principal, responsável por inicializar a tabela de símbolos e chamar as funções de passagem um e dois do montador. Ele também chama a função de imprimir a tabela de símbolo caso um dos parâmetros de entrada seja ‘-v’.

**- Tradutor:** este é o módulo onde são implementadas as funções de passagem um e dois e também funções auxiliares necessárias. Essas funções auxiliares são responsáveis por traduzir o nome dos símbolos, instruções e operandos em números referentes aos números dados no trabalho passado (por exemplo, MOV = 1, AX = 2, etc.), e também em retornar o código de operandos (se é registrador - registrador, ou registrador – memória, etc.).

**- TabelaSimbolo:** neste módulo são definidas as estruturas da tabela de símbolo e da lista encadeada bem como funções que operam na tabela (insere, busca símbolo, imprime, etc.). É nela que é implementada a função de atualizar o valor das variáveis para a próxima posição de memória livre depois das instruções.

As instruções continuam sendo salvas na posição 0 da memória do emulador, como no TP1, porem modificamos o TP1 para ler os dois primeiros bytes do arquivo gerado pelo montador que representam a posição de início do IP do emulador.

1. Testes