

Primeiro Exercício-Programa (EP 1)

MAC329 – Álgebra Booleana e Aplicações

Marcelo S. Reis ¹

Junior Barrera ¹

¹ Instituto de Matemática e Estatística, Universidade de São Paulo.

{msreis,jb}@ime.usp.br

7 de março de 2013

1 Introdução

O computador HIPO (de “computador hipotético”) é uma máquina virtual desenvolvida para fins educacionais [1]. HIPO foi criado para ensinar os princípios de funcionamento de uma CPU, para isso fazendo uso de uma arquitetura e de uma linguagem de máquina bastante simples.

A descrição completa do computador HIPO, incluindo a sua estrutura, conjunto de instruções e binário MS-DOS da implementação em Pascal da máquina virtual encontram-se disponíveis na página oficial do mesmo:

<http://www.ime.usp.br/~vwsetzer/hipo/hipo-descr.html>

1.1 Objetivos

O principal objetivo deste Exercício-Programa (EP) é o desenvolvimento de um montador (*assembler*) para o computador HIPO. O montador deverá ser implementado em linguagem C e terá como entrada um arquivo contendo as instruções em mnemônicos de um dado programa HIPO e como saída um arquivo contendo o programa em “código de máquina HIPO”. Os formatos dos arquivos serão detalhados na próxima seção.

Além disso, um pequeno programa para ser executado no HIPO também deverá ser elaborado; o enunciado do mesmo será descrito na seção 3.

2 O montador

2.1 Entrada

O montador deverá ter como entrada um programa com as instruções utilizando mnemônicos HIPO, conforme são descritos no artigo da máquina virtual, na seção 3 [1]. Segue abaixo um pequeno exemplo de um programa nesse formato:

```
; apos o ponto e virgula temos comentarios
;
00 {LDA} 10      ; soma valores
01 {ADD} 11
02 {STA} 12      ; prepara a impressao
03 {PRN} 12
04 {STP}         ; fim do programa
;
10 -7770         ; termos da soma
11 +7777
```

Observem que a primeira coluna indica o endereço da memória onde se localiza um par operação-endereço ou então um dado (i.e., um número inteiro entre -9999 e $+9999$). As linhas do programa sempre devem estar ordenadas em ordem crescente no índice da posição de memória; em qualquer linha, tudo o que vem a partir de um ponto e vírgula denota comentário e deve ser ignorado pelo montador.

2.2 Saída

O montador deverá produzir um arquivo de saída no formato de “código de máquina HIPO”, conforme é apresentado nos exemplos `prog1.hip` e `prog2.hip` na página oficial da máquina virtual [1]. Por exemplo, para o código com mnemônicos HIPO mostrado na seção anterior, o montador deverá produzir a seguinte saída:

```
+1110
+2111
+1212
+4112
+7000
```

```
-7770
+7777
```

Observem que, no formato `.hip`, a linha i corresponde a posição de memória $i - 1$; dessa forma, para podermos armazenar o número -7770 na posição de memória 10, precisamos colocar esse valor na linha 11 do arquivo de saída.

2.3 Chamada do montador

Suponhamos que o arquivo contendo o código do montador chama-se `EP1.c`. Utilizando o compilador `gcc` para gerar um binário, digamos `EP1.bin`, o mesmo deverá ser executado na linha de comando com o arquivo de entrada como argumento. Por exemplo, se executarmos o programa com o seguinte comando:

```
bash$ ./EP1.bin teste1.asm
```

caso o código do arquivo `.asm` esteja correto do ponto de vista sintático, então o programa deverá produzir o arquivo `teste1.hip`; caso contrário (e.g., posição de memória inválida na primeira coluna, instrução inexistente, etc.), deverá ser impresso na saída padrão a seguinte mensagem:

```
Erro de sintaxe.
```

Neste último caso, nenhum arquivo de saída deverá ser gerado.

3 Um pequeno programa HIPO

Escreva um programa HIPO que inicie lendo um inteiro k (através da instrução `INN`). Se $k \leq 0$ ou $k > 10$, então o programa deverá encerrar a sua execução; caso contrário, o mesmo

deverá ler k inteiros e imprimir na saída (através da instrução PRN) o valor do maior inteiro, o valor do menor inteiro e a média aritmética (quociente de uma divisão inteira), nessa ordem.

O programa deverá ser escrito utilizando mnemônicos HIPO, de acordo com o formato apresentado na seção anterior.

4 Entrega do EP e observações

4.1 Observações importantes

- Este EP deverá ser feito em grupos de 4 pessoas; em casos excepcionais e com autorização do professor / monitor, pode-se aceitar grupos com menos integrantes.
- O EP será corrigido de forma automatizada; portanto, é muito importante que o código siga **exatamente** os formatos de entrada e de saída.
- Utilize a máquina virtual disponível na página oficial do HIPO para testar a corretude de seu montador e de seu programa.
- Documente adequadamente o código-fonte, através de comentários. Isto vale tanto para o código do montador quanto para o do programa-exemplo.
- O montador deve ser compilável sem erros ou avisos, quando utilizado o compilador `gcc` com as opções `-ansi -pedantic -Wall`. Portanto, façam uso dessas opções para testar a corretude do código.
- Recomenda-se que as dúvidas sejam discutidas no fórum da disciplina no Moodle.

4.2 Data de entrega

Este EP deverá ser entregue até o dia **23 de março (sábado)**, através do fórum da disciplina no Moodle. Após esse dia, o sistema não permitirá a entrega deste EP.

Deverá ser entregue um zip ou tar.gz, contendo o arquivo `.c` do código do montador descrito na seção 2, além de um arquivo `.asm` do programa da seção 3. Apenas um membro do grupo deverá subir o arquivo; portanto, lembrem-se de colocar, em um comentário no início dos códigos, nome completo e número USP de todos os integrantes do grupo.

Referências

- [1] Valdemar W. Setzer. The HIPO computer – a tool for teaching basic computer principles through machine language. <http://www.ime.usp.br/~vwsetzer/hipo/hipo-descr.html>, 2000.