

Teste baseado na P1 de Programação de Sistemas – PCS3616

Ricardo Luis de Azevedo da Rocha

¹ 09/02/2017

1. **Questão conceitual** (2,5 pontos) Considere um simulador de CPU/Mem que deve implementar a troca de dados entre ambos. Em uma CPU os dados de um registrador são copiados por meio do acesso ao barramento para a memória e vice-versa. No ambiente simulado é possível acessar por meio de variáveis o conteúdo tanto da memória quanto dos registradores da CPU. Como proceder a simulação da transferência de dados entre ambos nesse caso? Suponha que o simulador foi escrito em uma linguagem que permite a alocação dinâmica de memória e que há rotinas já desenvolvidas para gerar uma área do tamanho desejado e inicializá-la e outra para devolver a a área ao sistema. São as rotinas `criaByte(<VALOR>)` e `retornaByte(<REGISTRADOR>)`, que retorna o objeto `Byte` contido no registrador especificado. Escreva um algoritmo, em linguagem de alto nível, para a sub-rotina `transfereDadoCPU-Mem()` que permite transferir o conteúdo do acumulador para a memória, definido pela variável de simulação dentro da CPU como `Accum`. Escreva seu algoritmo na forma descrita no exemplo abaixo:

```
function findMax(array, first, last) {
    if (first == last) return array[first];
    mid = first + (last - first)/2;
    a = findMax(array, first, mid);
    b = findMax(array, mid + 1, last);
    if (a < b) return b;
    return a;
}
```

2. **Linguagem de Máquina da MVN** (2,5 pontos) Desenvolva a seguinte primitiva, na “linguagem de máquina” da MVN, na forma de sub-rotina:

- `PUTLINEF`: escreve no dispositivo do tipo Monitor o conteúdo de um buffer.

Parâmetro: o endereço do buffer (`PL-END`) (os bytes cujos valores são 0000 indicam o final do buffer). Não é preciso tratar casos de erro.

Retorno (acumulador): não há.

Exemplo:

| Endereço | Dado |
|----------|------|
| 0002 | 4655 |
| 0004 | 4E46 |
| 0006 | 4121 |

`PL-END = 0002` => Saída: FUNFA!

Entrada (`PL-END`); Saída (Monitor)

Código do programa principal que chama `PUTLINEF`:

```
& /0000
MAIN      JP  START
PL-END    k /0000

STRING    K /4655      ; conteudo da string
          K /4E46
          K /4121
          K /0000      ; finalizador de string (NÃO DEVE SER ESCRITO)

START     LV  STRING    ; Param: endereço da string, é 0x0002
          MM  PL-END
          SC  PUTLINEF  ; Chama subrotina
END       HM  END      ; fim do programa

# MAIN
```

GABARITO

Q1:

```
Byte transfereDadoCPU-Mem() {  
    valor = retornaByte(Accum);  
    return criaByte(valor);  
}
```

Q2:

```
0014 0000 ; INICIO DE PUTLINEF  
0016 8002 ; CARREGA PL-END  
0018 402C ; ADICIONA OPERACAO  
001A 901C ; ARMAZENA EM 001C  
001C 8000 ; EXECUTA OPERACAO  
001E 102A ; TERMINA LACO SE ACUM = 0  
0020 E100 ; MOSTRA DADO NO MONITOR  
0022 801C ; CARREGA OPERACAO DE 001C  
0024 402E ; ADICIONA 0002 (PROXIMO)  
0026 901C ; ARMAZENA EM 001C  
0028 001C ; RETORNA AO INICIO DO LACO  
002A B014 ; RETORNO  
002C 8000 ; OPERACAO LOAD  
002E 0002 ; CONSTANTE 2
```