

**EA 869 – Turma U – 29/11/2011**  
**Prova 3.2 – Sem consulta – Duração: 100 minutos**

**NOME:**

**R.A.:**

**Questão 1. (1,0)** Explicar as diferenças entre **E/S Isolada** e **E/S mapeada em memória** com relação a:

- a) Barramentos de E/S e barramentos de memória
  
  
  
  
  
  
  
  
  
  
- b) Espaço de endereços e instruções de E/S
  
  
  
  
  
  
  
  
  
  
- c) Quantidade de endereços de E/S e tempo de execução de instruções de E/S.

**Questão 2. (1,5)** Explicar como funcionam os registradores de uma interface de E/S – Registrador de Estado, Registrador de Dados e Registrador de Controle – em uma operação de:

- a) Entrada de Dados
  
  
  
  
  
  
  
  
  
  
- b) Saída de Dados

**Questão 3. (1,0)** Quais são as diferenças entre a transferência serial assíncrona e a transferência serial síncrona?

**Questão 4. (3,5)** Fazer a montagem do programa listado abaixo, considerando que a palavra do processador tem 8 bits. A Tabela de Instruções de Máquina (TIM) e a Tabela de Pseudo-Instruções (TPI) são dadas abaixo.

- a) Mostre a Tabela de Símbolos; responder na próxima folha.  
b) Mostre o Código Objeto gerado indicando **todos** os endereços e seus conteúdos em hexadecimal; responder na próxima folha.  
c) Quantos **bytes** ocupa o código objeto gerado? **Resposta:**

```

PROGM:    EQU        1400H
ZERO:     EQU        0000H
SIZE:     EQU        0009H
          ORG        PROGM
START:    MVI        SIZE, R2
          MVI        DATA, R0
          MVI        ZERO, R4
LOOP:     MOVE       INPUT, (R0)+
          ADD        INPUT, R4
          DCR        R2
          JNZ        LOOP
          MOVE       R4, SUM
          HLT
          END        START
DATA:     EQU        0080H
          ORG        DATA
OUTPUT:   DS         SIZE
INPUT:    DWD        00A0H
SUM:      DS         08H
    
```

TIM			TPI	
Mnemônico	Código de Operação	Comprimento (bytes)	Mnemônico	Comprimento (bytes)
MVI R0	40H	2	ORG	-
MVI R2	42H	2	EQU	-
MVI R4	44H	2	DS	n
MOVE R0+	50H	3	DWD	2
ADD R4	64H	3	END	-
DCR R2	32H	1	ADR	2
JNZ	90H	3	GLOB	-
MOVE R4	84H	3		
HLT	99H	1		

**a) Tabela de Símbolos**

<b>Símbolo</b>	<b>Valor</b>

**b) Código objeto gerado**

<b>Endereço</b>	<b>Código</b>		<b>Endereço</b>	<b>Código</b>

**Questão 5. (1,0)**

a) Quais são as diferenças entre os esquemas de carregamento estático e de carregamento dinâmico?

b) Quais são as diferenças entre os esquemas de carregamento e ligação dinâmicos com resolução em tempo de carregamento e com resolução em tempo de execução?

**Questão 6. (3,0)** Um montador de ligação direta aplicado a dois arquivos em linguagem simbólica do processador Motorola 68K gerou os seguintes módulos objetos:

Módulo 1		Módulo 2	
0. 'MAIN'. 'SD'.0000.0001E		0. 'CALC'. 'SD'.0000.000C	
0. 'OUTPUT'. 'LD'.001C.0002		1.0000.02.4410	
1.0000.06.21AC00000018		1.0002.02.4316	
1.0006.06.4EBA00000000		1.0004.06.4AB000000000	
1.000C.06.440C0000001C		1.0004.02.4E75	
1.0012.04.20310016		2.0006.04. 'OUTPUT'	
1.0016.02.4E75		3.	
1.0018.04.00000020			
1.001C.02.0000			
2.0002.04. 'MAIN'			
2.0008.04. 'CALC'			
2.000E.04. 'MAIN'			
3.02.0000			

**Codificação usada no arquivo gerado pelo montador**

tipo=0 (ESD): tipo.simbolo.SD ou LD.end-relativo.compr-bytes  
 tipo=1 (TXT): tipo.end-relativo.compr-bytes.código-binário  
 tipo=2 (RLD): tipo.pos-relativa.compr-bytes.simbolo  
 tipo=3 (END): tipo.end-execução

Os módulos foram passados como argumentos na ordem: **Módulo 2 seguido de Módulo 1** – para um carregador de ligação direta; obteve-se o endereço inicial de carga (IPLA) **\$0900**.

- (a) Qual o conteúdo da Tabela de Símbolos Externos Globais (**GEST**) gerada pelo carregador?

Símbolo	Valor

- (b) O diagrama abaixo é um mapa de conteúdo da memória após o carregamento **sem** os ajustes de ligação e relocação. Indique neste mapa quais posições são ajustadas pelo carregador e qual o novo conteúdo destas posições.

	0	2	4	6	8	A	C	E
090-	4410	4316	4AB0	0000	0000	4E75	21AC	0000
091-	0018	4EBA	0000	0000	440C	0000	001C	2031
092-	0016	4E75	0000	0020	0000			

Posição	Novo conteúdo

- (c) Qual é o tamanho do código objeto em bytes e qual o endereço de início de execução após o carregamento do item b?

**Tamanho:**

**Endereço de início de execução:**

