

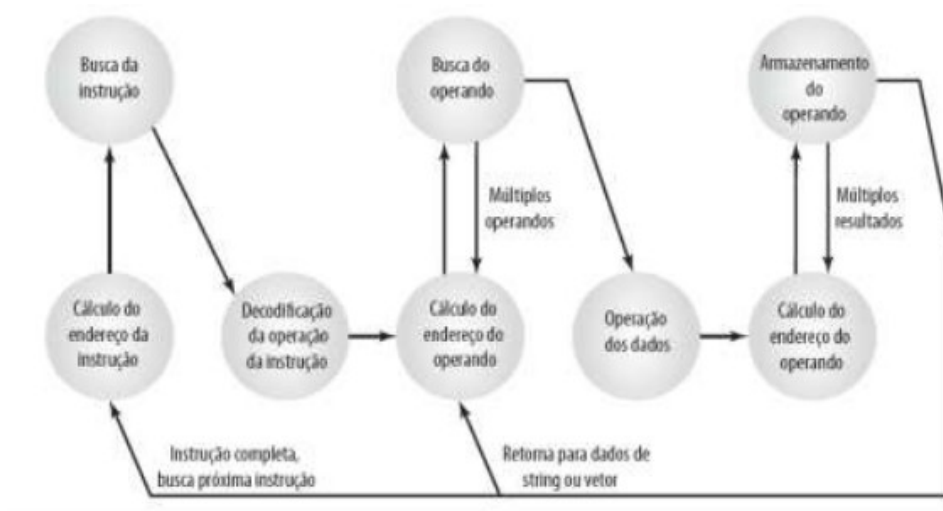
## Trabalho Prático 2º Bimestre

### Objetivo:

Aplicar os conhecimentos adquiridos durante o Bimestre para implementar um simulador de ciclos de instruções.

### Descrição:

O trabalho consiste em criar um programa que realize simulações das etapas descritas na Figura abaixo.



O programa deverá simular os registradores PC, IR, MAR, MBR e as flags de Zero e Negativo. O simulador **deve** possuir as seguintes instruções:

Código da Instrução	Operandos	Resultado
000001	#pos	$MBR \leftarrow \#pos$
000010	#pos #dado	$\#pos \leftarrow \#dado$
000011	#pos	$MBR \leftarrow MBR + \#pos$
000100	#pos	$MBR \leftarrow MBR - \#pos$
000101	#pos	$MBR \leftarrow MBR * \#pos$
000110	#pos	$MBR \leftarrow MBR / \#pos$
000111	#lin	JUMP to #lin
001000	#lin	JUMP IF Z to #lin
001001	#lin	JUMP IF N to #lin
001010	-	$MBR \leftarrow \text{raiz\_quadrada}(MBR)$
001011	-	$MBR \leftarrow - MBR$
001111	#pos	$\#pos \leftarrow MBR$
001100	-	NOP

Em um primeiro momento o programa deverá disponibilizar ao usuário a criação de um programa simbólico utilizando as instruções acima e uma representação da memória e registradores, que será utilizada para armazenar as instruções e os dados.

Com as instruções deve ser possível acompanhar a operação e a situação dos registradores em cada operação e ao final da execução.

## Exemplo:

Posição	Opcode	Operando1	Operando2	Operação
1	000010	251	5	Armazena 5 na posição 251
2	000010	252	10	Armazena 10 na posição 252
3	000010	253	15	Armazena 15 na posição 253
4	000001	251		MBR recebe o conteúdo da posição 251
5	000011	252		MBR recebe o conteúdo dele mesmo somado com o conteúdo da posição 252
6	000011	253		MBR recebe o conteúdo dele mesmo somado com o conteúdo da posição 253
7	001111	254		Posição 254 recebe o conteúdo de MBR
8	001100			Fim, no operation

No exemplo acima, as três primeiras instruções armazenam 5, 10 e 15, nas posições 251, 252 e 253 respectivamente. A quarta instrução carrega o registrador MBR com o conteúdo da posição 251 (5), em seguida a próxima instrução realiza uma soma entre o conteúdo do registrador MBR com o conteúdo da posição 252 (10), atribuindo o resultado (15) ao registrador MBR. A instrução na posição 6 realiza uma nova soma entre o conteúdo do MBR (15) e o conteúdo da posição 253 (15),

atribuindo o resultado (30) ao MBR. Por fim a instrução 7, salva o conteúdo do MBR (30) na posição 254 da memória, finalizando com a instrução 8 de nenhuma operação.

```
=====
=                                =
=          OPCOES:              =
=====
=          1 - INSERIR          =
=          2 - VER INSTRUcoes  =
=          3 - VER DADOS       =
=          4 - EXECUTAR        =
=          5 - LIMPAR          =
= 6 - DESATIVAR VISUALIZACAO EM LOTE=
=          7 - SAIR            =
=====
|
```

```
=====
=                                =
=          INSTRUcoes:         =
=====
=  COD   - OP1  - OP2  - RESULTADOS  =
= 000001 - #POS          - MBR <- #POS  =
= 000010 - #POS  #DADO - #POS <- #DADO  =
= 000011 - #POS          - MBR <- MBR + #POS  =
= 000100 - #POS          - MBR <- MBR - #POS  =
= 000101 - #POS          - MBR <- MBR * #POS  =
= 000110 - #POS          - MBR <- MBR / #POS  =
= 000111 - #LIN          - JUMP to #LIN  =
= 001000 - #LIN          - JUMP IF Z to #lin  =
= 001001 - #LIN          - JUMP IF N to #lin  =
= 001010 -              - MBR <- raiz_quadrada(MBR)=
= 001011 -              - MBR <- - MBR  =
= 001111 - #POS          - #POS <- MBR  =
= 001100 -              - NOP  =
=====
A INSTRUCAO SERA SEPARADA PARA MELHOR DIDATICA!
DIGITE A OPERACAO:
|
```

## **AVALIAÇÃO**

A avaliação do trabalho será a soma das seguintes notas:

- Código fonte e executável (0 a 6)
- Relatório (0 a 2)
- Apresentação para o professor (0 a 2)

## **ENTREGA**

Cada **aluno** deve entregar **três** arquivos, um **relatório técnico**, um **arquivo fonte do programa** e uma **documentação** com as **instruções** para **compilar** e **rodar** o programa.

### **O relatório técnico deve conter:**

- Introdução
- Objetivos
- Motivação e recursos utilizados durante a implementação
  - Estruturas de dados
  - Linguagem de programação
  - Bibliotecas
  - Entre outras questões relativas à implementação
- Resultados
- Conclusão
- Referências

**O código fonte deve estar comentado.**

**Trabalhos copiados serão zerados.**

**A não apresentação para o professor fará com que o trabalho seja zerado.**

## **DATA DE ENTREGA**

Envio dos arquivos via e-mail até dia **29/07/2024** para **guilherme.henrique@ies.unespar.edu.br**, ou e-mail com o link para download via Github, Google Drive, Dropbox, etc.

O assunto do e-mail **deve** ser obrigatoriamente **“Trabalho de Arquitetura – 2º Bimestre – C.C UNESPAR”**

Data da apresentação: **30/07/2024**