



# Fundamentos de Arquitetura de Computadores

## Trabalho 02

Prof. Tiago Alves

### Operações Lógicas

#### *Introdução*

A disciplina de Fundamentos de Arquitetura de Computadores trata de diversos tópicos que nos ajudam a compreender como sistemas eletrônicos de computação são construídos. Esse tipo de conhecimento ajudará profissionais de áreas afetas a tecnologias de informação e comunicação a aplicarem, adequadamente, um computador digital na realização de tarefas que, devido à sua natureza, serão melhores conduzidas por um sistema automatizado.

Além de identificar a conveniência da aplicação dos computadores digitais, a disciplina ajudará a desenvolver competências necessárias para a solução de problemas em sistemas computacionais em operação, principalmente problemas decorrentes de análise de desempenho.

#### *Objetivos*

- 1) Exercitar conceitos da linguagem de montagem (assembly) MIPS.
- 2) Interagir com ferramentas de desenvolvimento para criação, gerenciamento, depuração e testes de projeto de aplicações.

#### *Contextualização*

Um bit de paridade, ou um bit de conferência, é um bit adicionado a uma string de um código binário de forma que a string composta possua um número total de bits 1 par (paridade par) ou ímpar (paridade ímpar). Bits de paridade são uma das formas mais simples de códigos de detecção de erro e foram utilizados nos primórdios da Internet na proteção dos dados transmitidos pelos modems, dados que trafegavam em links de comunicação seriais, geralmente providos pelo par trançado das linhas telefônicas.

Há duas variantes para o cálculo de bits de paridade: a paridade par e a paridade ímpar. No caso da paridade par, para um conjunto de bits, as ocorrências dos bits cujo valor é 1 são contadas. Se a contagem total resultar em um número ímpar, o bit de paridade é ajustado para 1, fazendo com que a quantidade de ocorrências de 1 no conjunto (incluindo o bit de paridade) seja um número par. Se a contagem de 1s em determinado conjunto de bits já for par, o bit de paridade é ajustado para 0.

O bit de paridade pode ser colocado na posição mais significativa da palavra resultante, conforme exemplificado na Tabela 1.



*Tabela 1: Cálculo de Bit de Paridade para palavras de 7 bits.*

7 bits of data	(count of 1-bits)	8 bits including parity	
		even	odd
0000000	0	<b>0</b> 0000000 = 0	<b>1</b> 0000000
1010001	3	<b>1</b> 1010001 = 209	<b>0</b> 1010001
1101001	4	<b>0</b> 1101001 = 105	<b>1</b> 1101001
1111111	7	<b>1</b> 1111111 = 255	<b>0</b> 1111111

## Referências Teóricas

David A. Patterson; John Hennessy, Organização e Projeto de Computadores, Campus, 3ª Edição, 2005.

## Material Necessário

- Computador com sistema operacional programável
- Ambiente de simulação para arquitetura MIPS: MARS ou SPIM.

## Roteiro

- 1) Revisão de conceitos básicos da arquitetura MIPS.

Colete o material acompanhante do roteiro do trabalho a partir do Moodle da disciplina e estude os conceitos básicos da arquitetura MIPS.

- 2) Realizar as implementações solicitadas no questionário do trabalho.

## Implementações e Questões para Estudo

- 1) Implemente um programa em *assembly* MIPS que atenda aos seguintes comandos:
  - Leia um número inteiro menor que 128. Por exemplo:  
127
  - Processe o número lido calculando o valor de bit necessário à ser adicionado para que a quantidade de bits 1 seja um número par. Para a entrada acima, a saída é:  
bit-paridade: 1  
saida: 255
  - Outros exemplos:  
128



```
=====
entrada incorreta

0
=====
bit-paridade: 0
saida: 0

1
=====
bit-paridade: 1
saida: 129
```

## *Instruções e Recomendações*

A submissão das respostas aos problemas dos trabalhos deverá ser feita no MOJ e, também, através do Moodle da disciplina.

Para o Moodle, a solução para Trabalho 02 deverá ser entregue em um pacote ZIP. A dupla de alunos deverá nomear o pacote ZIP da seguinte forma: nome\_sobrenome\_matricula\_nome\_sobrenome\_matricula\_**trab02.zip**.

Entre os artefatos esperados, listam-se:

- códigos-fonte das soluções dos problemas;
- documentação mínima da aplicação:
  - o qual sistema operacional foi usado na construção do sistema;
  - o qual ambiente de desenvolvimento foi usado;
  - o quais são as telas (instruções de uso)
  - o quais são as limitações conhecidas

Não devem ser submetidos executáveis.

Códigos-fonte com erros de compilação/montagem serão desconsiderados (anulados).

Os trabalhos poderão ser realizados em duplas; a identificação de cópia ou plágio irá provocar anulação de todos os artefatos em recorrência.