**Ciência da Computação**

**Fundação Universidade Federal De Rondônia – UNIR**

**ESTRUTURA DE DADOS II**

**Porto Velho – RO – Brasil**

Thales Junior De Souza Gomes - RA:201611117 [Thales3238Junior@gmail.com](mailto:Thales3238Junior@gmail.com)

Gieldson Alves de Oliveira - RA:201611923 [Gioliveira920@gmail.com](mailto:Gioliveira920@gmail.com)

**Relatório de utilização**

**Problema proposto:**

Em uma certa cidade há **N** intersecções ligadas por ruas de mão única e ruas com mão dupla de direção. É uma cidade moderna, de forma que muitas ruas atravessam túneis ou têm viadutos. Evidentemente é necessário que se possa viajar entre quaisquer duas intersecções, isto é, dadas duas intersecções **V** e **W**, deve ser possível viajar de **V** para **W** e de **W** para **V**.

O programa consiste em ler a descrição do sistema de tráfego de uma cidade que está no arquivo.txt para teste, para determinar se o requisito de conexidade é satisfeito ou não.

**Utilização:**

O arquivo.txt contém dois números inteiros **N** e **M**, separados por um espaço em branco, indicando respectivamente o número de intersecções (2 ≤ **N** ≤ 2000) e o número de ruas (2 ≤ **M** ≤ **N**(**N**−1)/2). A descrição de cada uma das M ruas consiste de três inteiros **V**, **W** e **P**, separados por um espaço em branco, onde **V** e **W** são identificadores distintos de intersecções (1 ≤ **V**, **W** ≤ **N** , **V** ≠ **W** ) e **P** pode ser 1 ou 2; se **P** = 1 então a rua é de mão única, e vai de **V** para **W**, se **P** = 2 então a rua é de mão dupla, liga **V** e **W**. Não pode existir duas ruas ligando as mesmas intersecções.

Ao executar o programa o resultado será igual a **1** se o requisito de conexidade estiver satisfeito, ou o resultado será igual a **0** se o requisito de conexidade não estiver satisfeito.

**Exemplos dos Grafos**

**Legenda:**

**1**

**: P =** 1 Rua de mão única.

**: P =** 2 Rua de mão dupla.

**2**

**Exemplo 1:**

**N:** 4 **M:** 5

**V:** 1 **W:** 2 **P:** 1

**V:** 1 **W:** 3 **P:** 2

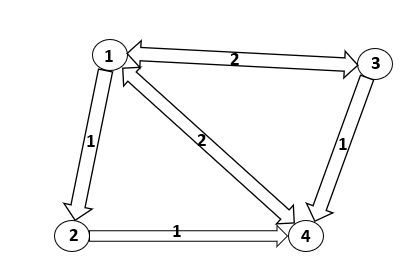
**V:** 2 **W:** 4 **P:** 1

**V:** 3 **W:** 4 **P:** 1

**V:** 4 **W:** 1 **P:** 2

**Resultado:**

**Requisito de conexidade satisfeito.**

****

**-------------------------------------------------------------------------------------------------------**

**Exemplo 2:**

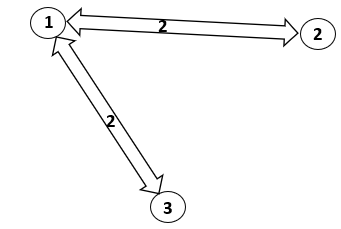
**N:** 3 **M:** 2

**V:** 1 **W:** 2 **P:** 2

**V:** 1 **W:** 3 **P:** 2

**Resultado:**

**Requisito de conexidade satisfeito.**

****

**-------------------------------------------------------------------------------------------------------**

**Exemplo 3:**

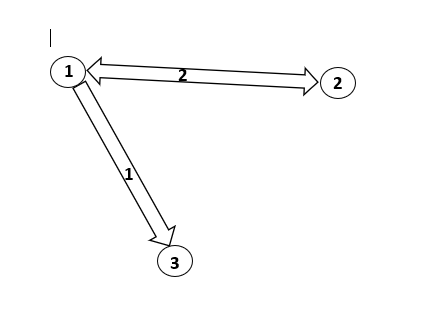
**N:** 3 **M:** 2

**V:** 1 **W:** 2 **P:** 2

**V:** 1 **W:** 3 **P:** 1

**Resultado:**

**Requisito de conexidade não satisfeito.**

****

**-------------------------------------------------------------------------------------------------------**

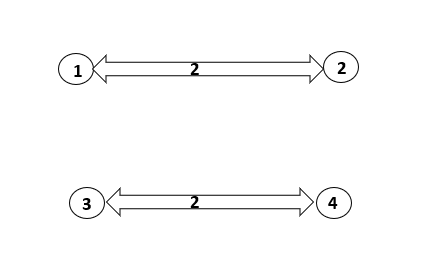
**Exemplo 4:**

**N:** 4 **M:** 2

**V:** 1 **W:** 2 **P:** 2

**V:** 3 **W:** 4 **P:** 2

**Resultado:**

**Requisito de conexidade não satisfeito.**

**-------------------------------------------------------------------------------------------------------**

**Conclusão:**

Para que o **requisito de conexidade** esteja satisfeito, o grafo não deve conter fonte, sumidouro ou faltar conexão que permita sair de um vértice (intersecção) qualquer e visitar todos os outros vértices (intersecções) presentes no grafo.