# UNIVERSIDADE FEDERAL DE UBERLÂNDIA
# PPGCC – Tópicos Especiais em Sistemas de Computação 1: Algoritmos Distribuídos PGC307C

Caio Thomás Oliveira
Rodrigo Moreira

Relatório do trabalho que implementa o algoritmo *Clockwise Leader Election*.

O projeto foi desenvolvido em *HTML*, *CSS* e *Javascript* e consiste em simular o algoritmo do livro utilizando animação providas pelas bibliotecas *Jquery* e *Jquery Path*. A primeira biblioteca fornece muitas funcionalidades para objetos de *tags* de *HTML* para realizar efeitos, validações e alteração em tempo de execução. O *Jquery Path* fornece uma *API* para realizar animações de acordo com um *plot*, ou seja, um linha onde o objeto poderá ser animado. A escolha dessas linguagens se justifica pelo objetivo principal do trabalho que é simular o algoritmo de forma intuitiva e visual.

O pseudocódigo extraído livro *"Distributed Algorithtms: an intuitive approach"* ajuda montar um algoritmo para a linguagem *Javascript* seguindo a mesma ideia. A Figura 1 mostra o pseudocódigo utilizado para implementação do algorítimo.

**Algorithm 8** Clockwise

1: **Each node** $v$ executes the following code:
2: $v$ sends a message with its identifier (for simplicity also $v$) to its clockwise neighbor. {If node $v$ already received a message $w$ with $w > v$, then node $v$ can skip this step; if node $v$ receives its first message $w$ with $w < v$, then node $v$ will immediately send $v$.}
3: **if** $v$ receives a message $w$ with $w > v$ **then**
4: $v$ forwards $w$ to its clockwise neighbor
5: $v$ decides not to be the leader, if it has not done so already.
6: **else if** $v$ receives its own identifier $v$ **then**
7: $v$ decides to be the leader
8: **end if**

Figura 1: Pseudocódigo *Clockwise Algorithm*.

Inicialmente, tem-se uma classe Node que contém atributos:

- Marca: identificador do node, um nome qualquer.
- Valor: elemento inteiro para definir quem será o líder.
- Mensagem: quando um nó vizinho envia uma mensagem, ela é armazenada neste atributo.
- *Leader*: bit para marcar se é o líder ou não. Este atributo é utilizado apenas para auxiliar na animação.

Os objetos são criados e armazenados em um vetor. Para cada nó é criado uma função, isto ocorre devido ao fato de que para cada nó são chamadas animações exclusivas. Com o *JavaScript* é possível executar algum código em intervalos de tempo especificados. Foi utilizado a temporização de eventos *setInterval*, cujo objetivo é executar determinada função, em intervalos de tempo especificados. Com isso, são criados temporizadores para cada nó, para que sejam executadas as funções. Foi utilizado este *setInterval* devido ao fato de que o *Javascript* não possui *Threads* para ficar ouvindo se chegou alguma mensagem. A tabela e o relatório é atualizado constantemente de acordo com a execução da aplicação.

O usuário clica em qualquer nó para iniciar o algoritmo. Quando clicado, uma mensagem é disparada e envida para o próximo nó. O nó fica "ouvindo", ou seja, verifica no atributo mensagem se existe um elemento. Se tiver, e a mensagem for igual ao valor que ele possui, ele é o líder e muda para cor verde. A cor azul mostra que o nó recebeu uma mensagem e está processando. O nó repassa o seu valor quando a mensagem recebida for menor. Se for maior, repassa o valor da mensagem recebida pelo nó anterior.

Apesar do projeto e o algoritmo serem simples, o desenvolvimento desta aplicação poderá auxiliar novos alunos a entenderem a execução de forma fácil e visual. Além do mais, a escolha das tecnologias *web* permite que ela seja acessada de forma rápida e sem instalação.

A Figura 2 ilustra o exemplo da execução do algorítimo. Na Figura tem-se uma tabela que apresenta as trocas de mensagens, abaixo da tabela encontra-se o *log* que mostra os detalhes das trocas de mensagens entre os nós. Animação visual é ilustrada no anel à direita da Figura. Após a execução, o nó na coloração verde é o líder. Ademais as mensagens continuam no anel de forma intermitente. Caso haja influência de um agente Bizantino, por exemplo, adulteração do conteúdo da mensagem, há possibilidade de haver eleição de outro líder.

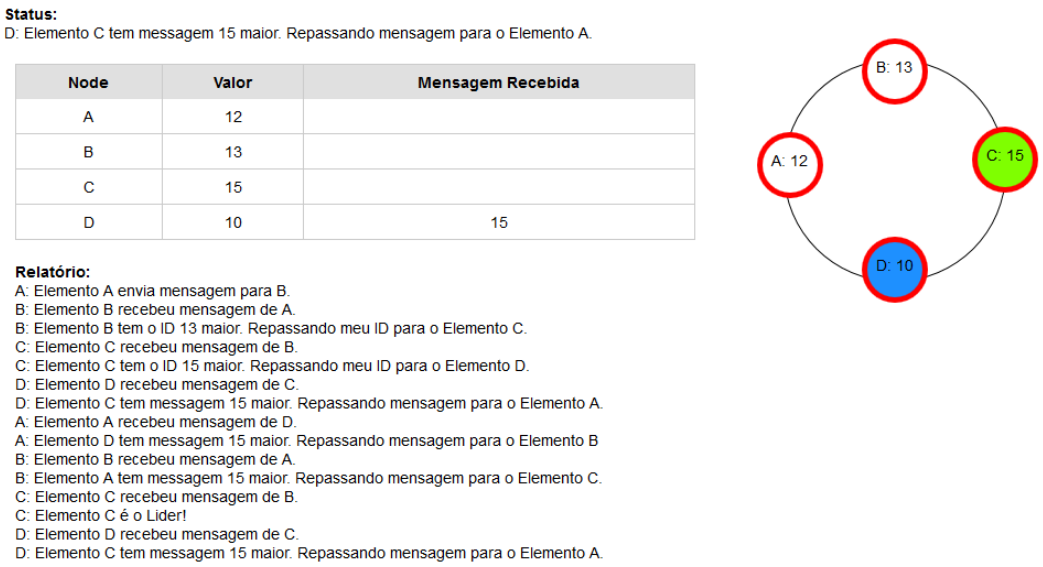


Figura 2: Exemplo execução do algorítimo.