

# REDES DE COMPUTADORES A

## Atividade 6

### Desenvolvimento da Atividade

A atividade deverá ser desenvolvida em **grupos de no máximo 4 integrantes**.

Para a realização desta atividade será necessário o uso de **dois** dispositivos Arduino Uno equipados com placas de rede Ethernet (Ethernet Shield) e um computador com Sistema Operacional Linux. Por isso recomenda-se que esta atividade seja realizada no Laboratório de Eletrônica (LABEL).

### Normas de conduta

A interação entre os grupos é estimulada, no entanto qualquer tentativa de plágio de trabalhos será punida com a **nota -Nmax nos trabalhos para todos os envolvidos**.

### Descrição da Atividade

**Objetivo:** Obter conhecimentos básicos para o desenvolvimento de software cliente e servidor em ambientes TCP/IP em sistemas embarcados.

Leia atentamente as informações a seguir e realize as atividades solicitadas:

- Estude os textos Projeto de Software de Clientes e Projeto e Implementação de Servidores;
- Estude os Capítulos 1, 2, 13 e 17 do Livro Arduino Básico;
- Implemente uma aplicação para monitoramento de temperatura de um ambiente remoto, como descrito a seguir;
- Elabore um relatório detalhado mostrando todo o trabalho realizado, como descrito a seguir.

Nesta atividade serão implementados um cliente TCP e um servidor TCP concorrente que permitirão o monitoramento de temperatura de um ambiente remoto utilizando sockets TCP.

O programa cliente TCP será executado em um dispositivo Arduino Uno equipado com uma placa de rede Ethernet (Ethernet Shield). Devem estar conectados a este dispositivo um led e um sensor de temperatura.

Ao ser iniciado, o Arduino deve obter um endereço IP dinamicamente (utilizando o protocolo DHCP) e inicializar o led e o sensor de temperatura adequadamente. Realizado o procedimento inicial, o dispositivo deve a cada 10 segundos medir a temperatura do ambiente e enviar esta informação ao programa servidor TCP em um endereço IP e uma porta pré-determinados, aguardando em seguida por uma resposta do servidor informando se o seu led deve ser ligado ou desligado.

Para a realização desta atividade devem ser iniciados pelo menos dois dispositivos Arduino com a configuração descrita anteriormente.

O programa servidor TCP deve ser executado pelo usuário em um computador com Sistema Operacional Linux, devendo informar o número da porta na qual o programa servidor aguardará por requisições, como no exemplo a seguir:

```
# ./servidor 5000
```

O servidor deve aguardar por requisições de conexão enviadas pelos clientes e quando um novo cliente se conectar deve exibir uma mensagem informando o endereço IP e a porta do cliente que solicitou a conexão.

Em seguida deve receber a temperatura enviada pelo cliente. Caso a temperatura informada por este cliente seja a maior temperatura registrada entre os clientes conectados, deve enviar a este cliente um comando para que ele acenda o seu led. Caso contrário, deve enviar a este cliente um comando para que ele apague o seu led. Em ambos os casos deve ser exibida uma mensagem na tela do servidor informando a identificação do respectivo cliente, a temperatura recebida e o comando enviado.

O servidor deve ser capaz de atender a vários clientes simultaneamente, através da criação de threads ou processos filhos. O servidor também deve evitar a existência de threads ou processos zumbi.

Além disso, cada cliente deve enviar todas as requisições através de uma única conexão TCP (conexão persistente).

## Detalhes da entrega

Cada integrante do grupo deverá postar em seu escaninho no AVA, na pasta **Atividade6**, o **código fonte de todos os programas implementados** e um **relatório em formato digital (.pdf)** contendo:

- detalhes do ambiente utilizado no desenvolvimento da atividade (contendo fotos do ambiente);
- detalhes de projeto e de implementação da aplicação, apresentando uma descrição de alto nível dos programas implementados (contendo textos explicativos e diagramas) e o formato das requisições e respostas enviadas;
- descrição do processo de compilação (contendo um screenshot que mostre os comandos utilizados na compilação e os resultados obtidos);
- descrição dos testes realizados (contendo screenshots da execução dos programas e os resultados obtidos, demonstrando que o programa realiza as tarefas solicitadas);
- os códigos fonte dos programas implementados.