UNIVERSIDADE FEDERAL DE MINAS GERAIS

Trabalho Prático 3

Redes de Computadores

Elise Guimarães de Araújo

24/6/2015

Introdução

Esse relatório discorre acerca da implementação de um servidor capaz de suportar múltiplas conexões em duas portas, uma delas operando sob o protocolo HTTP, e de um cliente capaz de testar a validade da conjectura de Beal para intervalos finitos de números naturais. O protocolo de transporte escolhido para tais implementações foi o TCP, o que dispensa a necessidade de dedicar parte das aplicações à garantia de integridade das mensagens.

Implementação

Os códigos foram inteiramente desenvolvidos em linguagem C, utilizando as bibliotecas de sockets.

Denomine-se "servidor de testes" a parte do programa - codificado no arquivo server.c - responsável por distribuir intervalos da conjectura de Beal para que clientes busquem contra-exemplos, e "servidor de http" a parte responsável por externar uma pagina em html (por uma porta diferente daquela utilizada pelo servidor de testes).

Três itens chave resumem essa implementação: suporte a múltiplas conexões, suporte a http e busca por contra-exemplos da conjectura de Beal.

O método de threading foi escolhido para dar suporte a múltiplas conexões. Funções da biblioteca pthread foram empregadas para criar uma thread a cada novo cliente que se conecta ao servidor de testes, criando assim uma componente do processo principal que é gerenciada independentemente. Uma thread também foi utilizada para processar paralelamente o servidor de http. Este, por sua vez, utiliza forks - método escolhido devido à simplicidade de implementação - para suportar mútiplos clientes (browsers, no caso).

O servidor de http funciona com base nas funçoes send e recv. O suporte ao protocolo de hipertexto foi dado formatando-se manualmente as strings para que se atingisse o comportamento esperado. Uma sessão http tem início quando o cliente inicia uma requisição estabelecendo uma conexão TCP para a porta escolhida no servidor (e.g. digitando no browser "127.0.0.1:32001"). O servidor então aguarda a mensagem de requisição do cliente (com um recv de buffer tamanho 1000, no caso), para em seguida retornar a linha de estado "HTTP/1.1 200 OK" e subsequentemente as mensagens com o conteudo em HTML.

A busca por contra exemplos é feita com o servidor passando a cada cliente intervalos de bases e expoentes a serem analisados. O cliente então itera por esses intervalos - testando da base mínima à máxima e do expoente mínimo ao máximo - buscando, primeiramente, valores de A^x + B^y que resultem em potências perfeitas (i.e. C^y). A cada potência perfeita encontrada, o cliente utiliza o método Euclidiano para encontrar os fatores primos comuns às três bases (chamando a funçao commonFactor3(A,B,C)). Caso não haja um fator primo comum, um resultado foi encontrado. Ao fim das iterações, o cliente envia ao server, primeiramente, o número de resultados encontrados e, em seguida, uma mensagem de tamanho fixo para cada resultado, contendo os valores de A,B,C,x,y e z que compõem a exceçao à conjectura de Beal. O servidor de testes tem comportamento complementar, aguardando, na thread designada para lidar com a conexão de determinado cliente de teste, o número de resultados encontrados e, caso esse resultado seja diferente de zero, executando um recv com tamanho de buffer fixo para armazenar cada um dos valores que compõem a exceção.

Metodologia

Os programas foram desenvolvidos, compilados e executados no sistema operacional Ubuntu versão 12.04 (Precise Pangolin). As especificações do sistema relevantes são: processador Intel Core i7-4800MQ 2.70GHz, 8GB de RAM, adaptadores de rede Killer Wireless-N 1202 e Realtek PCIe Gigabit Ethernet. Os testes foram realizados em uma única máquina.

Não houve preocupação com medição de tempo, de forma que a metologia de testes adotada foi exclusivamente qualitativa.

A sequência de testes consistiu de manter o servidor sendo executado e realizar múltiplas conexões simultâneas da parte dos clientes de teste, ao mesmo tempo que o browser se conectava pela porta escolhida para http. Consideram-se características de uma execução de sucesso:

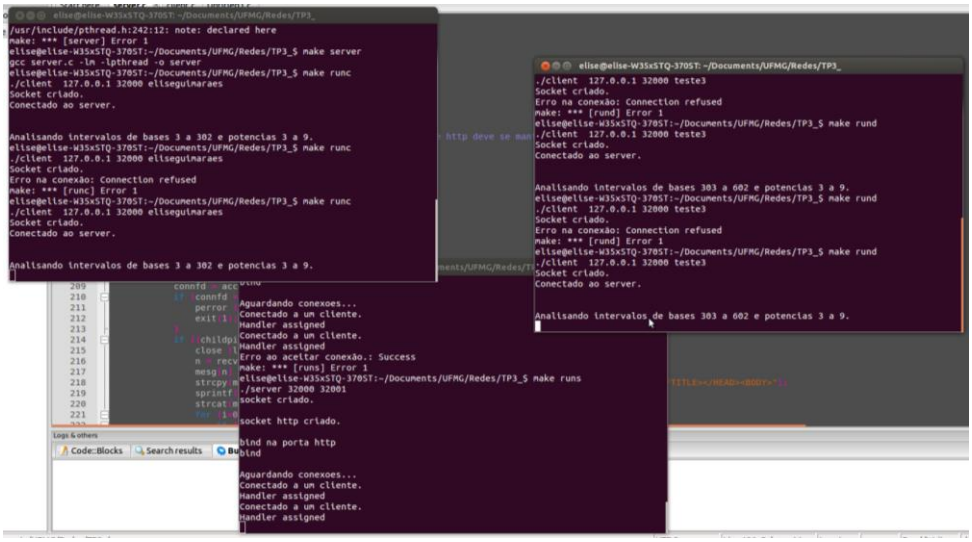
- O servidor de testes suportar no mínimo dois clientes conectados simultaneamente.
- O servidor de testes distribuir corretamente os intervalos a serem analisados.
- Os clientes de teste analisarem corretamente os intervalos submetidos.
- O browser conseguir se conectar ao servidor e exibir corretamente a saída.
- Atualizações no browser atualizarem os resultados conforme o progresso da execução.

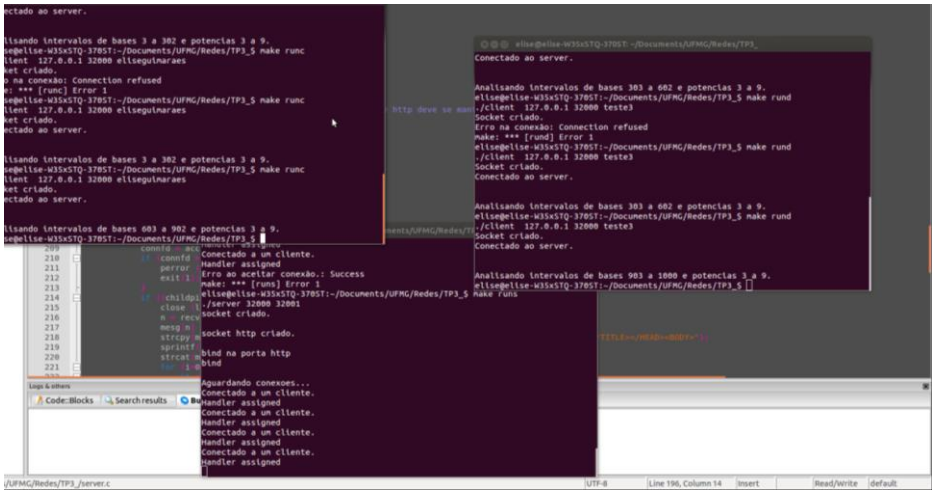
A sessão de resultados apresenta provas de conceito.

Resultados

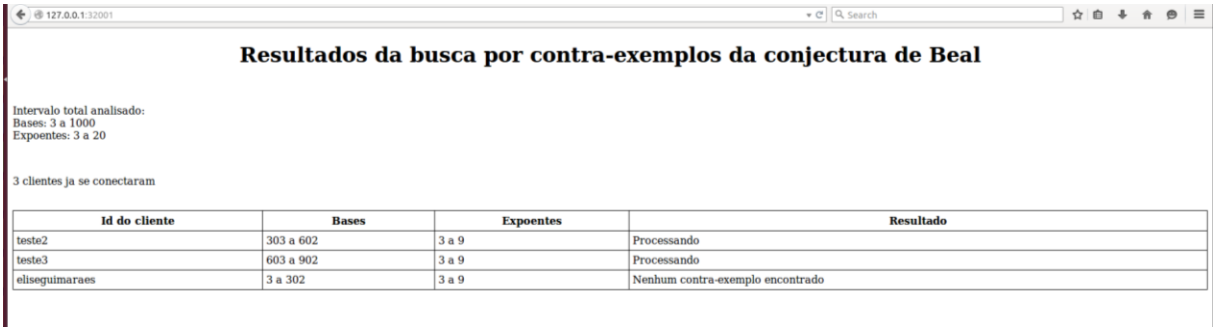
As imagens a seguir explicitam o funcionamento do programa.

Dois clientes executando testes simultaneamente:





Página em HTML:



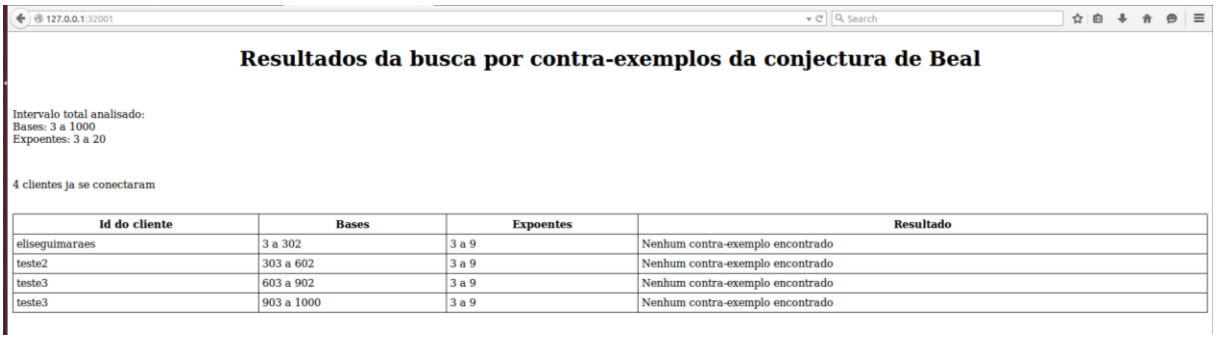
Resultados da busca por contra-exemplos da conjectura de Beal

Intervalo total analisado:
Bases: 3 a 1000
Expoentes: 3 a 20

3 clientes ja se conectaram

Id do cliente	Bases	Expoentes	Resultado
teste2	303 a 602	3 a 9	Processando
teste3	603 a 902	3 a 9	Processando
eliseguimaraes	3 a 302	3 a 9	Nenhum contra-exemplo encontrado

Página após uma atualização:



Resultados da busca por contra-exemplos da conjectura de Beal

Intervalo total analisado:
Bases: 3 a 1000
Expoentes: 3 a 20

4 clientes ja se conectaram

Id do cliente	Bases	Expoentes	Resultado
eliseguimaraes	3 a 302	3 a 9	Nenhum contra-exemplo encontrado
teste2	303 a 602	3 a 9	Nenhum contra-exemplo encontrado
teste3	603 a 902	3 a 9	Nenhum contra-exemplo encontrado
teste3	903 a 1000	3 a 9	Nenhum contra-exemplo encontrado

Análise

Conforme explicitado na sessão de resultados, obteve-se sucesso em todos os critérios esperados. O servidor suportou perfeitamente conexões múltiplas tanto de clientes de teste quanto de clientes http (browsers), distribuindo corretamente os intervalos, que por sua vez foram analisados corretamente - e de maneira relativamente rápida - pelos clientes. O programa codificado em client.c tem uma linha que pode,

caso necessário, imprimir na tela todos os resultados de potências perfeitas encontrados (inclusive os com fatores primos comuns). A linha se encontra comentada a fim de se manter uma saída limpa.

Ademais, é importante ressaltar que nenhuma exceção á conjectura de Beal foi encontrada, caracterizando completo fracasso caso se considere o ganho de um milhão de dólares o objetivo primário desse trabalho prático.

Conclusão

Esse relatório tratou da implementação de um servidor com suporte a múltiplas conexões em duas portas. Obteve-se sucesso em todos os critérios de funcionamento observados.

Esse trabalho prático possibilitou aprendizado acerca de métodos de programação paralela, do protocolo HTTP - cujas requisições e respostas foram feitas quase que manualmente, utilizando-se apenas as funções send e recv -, da conjectura de Beal - e dos teoremas de Fermat, consequentemente -, do protocolo de transporte TCP - cujas vantagens se destacaram consideravelmente após a implementação do TP2, em que não havia tais comodidades -, de programação de sockets e da linguagem C em geral. Além das vantagens supracitadas, é válido mencionar que foram desenvolvidas habilidades de administração e manutenção de software, uma vez que as diferentes funcionalidades exigidas demandaram consideráveis modularização e organização no desenvolvimento.