**Linguagem de Programação I**

**Relatório Aula 22**

Amanda – 22.00721-0

b. O retorno da execução da classe Client.java foi o seguinte erro: “Erro ao iniciar o cliente: Connection refused: connect”, além do título “Console Cliente” e “Cliente Finalizado!”. O erro quer dizer que não foi possível conectar com o servidor, que não está aberto para receber mensagens. Isso pode ocorrer por alguns motivos como a porta estar indisponível, o IP incorreto ou o servidor não estar rodando. Neste caso, sabe-se que o servidor não foi aberto para receber mensagens.

Uma imagem contendo Texto

Descrição gerada automaticamente

c. O retorno da execução da classe Server.java exibe na tela “Servidor iniciado em 127.0.0.1:3334” além do título “Console do Servidor” e a mensagem “Aguardando conexão de um cliente!”, indicando que o servidor está aberto para receber mensagens do cliente. Porém como nenhum cliente está conectado, nada mais será feito até que algum objeto se conecte.

Tela preta com letras brancas

Descrição gerada automaticamente

d. Após executar o servidor e conectar um cliente, recebe-se a mensagem “Cliente 127.0.0.1:3334 conectado!” e “Aguardando mensagem para enviar ao servidor!” no prompt do cliente, além de “Digite uma mensagem para o servidor ou <sair> para finalizar:” que aguarda a mensagem que será enviada e se repete até que o cliente decida parar. Já no console do servidor estão presentes as mensagens “Servidor iniciado em 127.0.0.1:3334” e “Cliente /127.0.0.1:64255 conectado!” após o cliente se conectar. O servidor aguarda o cliente mandar alguma mensagem. Quando a menagem é enviada, ela chega ao servidor que exibe “Mensagem recebida do cliente /127.0.0.1:64255:” com a mensagem enviada até que o cliente opte por sair. Por fim, após a mensagem de parada, é exibido “Cliente finalizado!” no console do cliente, porém o servidor não foi finalizado.

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

e. Sem encerrar o servidor, as mensagens passadas se mantêm. Além disso, o servidor notifica a entrada de um novo cliente exibindo “Cliente /127.0.0.1:64743 conectado!” e logo em seguida são exibidas as mensagens enviadas por esse novo cliente. No console do cliente, é iniciado um novo cliente diferente do passado que enviará novas mensagens até que o usuário opte por sair – o que não será o caso dessa vez, conforme pedido no enunciado.

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

f. Ao conectar um segundo cliente no mesmo servidor já iniciado, o novo cliente é reconhecido e as mensagens enviadas por ele são aceitas normalmente. Por usar threads, o servidor consegue processar mais de um cliente. Enquanto isso, a primeira instância segue ativa caso alguma mensagem queria ser enviada.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Servidor

Texto

Descrição gerada automaticamente

Primeira instância

Texto

Descrição gerada automaticamente

Segunda instância

g. Ao conectar um terceiro cliente no mesmo servidor já iniciado, o novo cliente é reconhecido e as mensagens enviadas por ele são aceitas normalmente. Por usar threads, o servidor consegue processar mais de um cliente. Enquanto isso, a primeira instância e a segunda instância seguem ativas caso alguma mensagem queria ser enviada.

Texto

Descrição gerada automaticamente

Servidor

Texto

Descrição gerada automaticamente

Primeira instância

Texto

Descrição gerada automaticamente

Segunda instância

Texto

Descrição gerada automaticamente

Terceira instância

h. Sem encerrar nenhuma instância, ao mandar mais mensagens por cada uma delas, todas os envios são aceitos pelo servidor, que exibe quem enviou e o que foi enviado.

Texto

Descrição gerada automaticamente

Servidor

i. Encerrando cada um dos clientes na ordem instanciada – do primeiro para o último -, todos os clientes se encerram e exibem a mensagem de cliente encerrado, enquanto o servidor se mantém ativo para a conexão de novos clientes.



Mesma mensagem foi apresentada para os 3 clientes

j. Ao tentar iniciar um servidor sem o anterior ter sido finalizado, é gerado um erro pois a porta já está ocupada.

