



ft_irc

Internet Relay Chat

*Preâmbulo: Este projeto é sobre criar seu próprio servidor IRC.
Você usará um cliente IRC real para conectar-se ao seu servidor e testá-lo.
A Internet é governada por protocolos padrão sólidos que permitem que computadores
conectados interajam entre si.
É sempre benéfico entender esses protocolos.*

Versão: 9.1

Sumário

I	Introdução	2
II	Regras gerais	3
III	Instruções para IA	4
IV	Parte obrigatória	7
IV.1	Requisitos	9
IV.2	Para MacOS apenas	10
IV.3	Exemplo de teste	10
V	Parte bônus	11
VI	Entrega e avaliação por pares	12

Capítulo I

Introdução

Internet Relay Chat ou IRC é um protocolo de comunicação baseado em texto na Internet. Ele oferece mensagens em tempo real que podem ser públicas ou privadas. Os usuários podem trocar mensagens diretas e participar de canais de grupo.

Os clientes IRC se conectam aos servidores IRC para participar dos canais. Os servidores IRC estão conectados entre si para formar uma rede.

Capítulo II

Regras gerais

- Seu programa não deve travar em nenhuma circunstância (mesmo quando ficar sem memória) e não deve ser fechado inesperadamente.
Se isso acontecer, seu projeto será considerado não funcional e sua nota será 0.
- Você deve entregar um **Makefile** que irá compilar seus arquivos de origem. Ele não deve realizar a religação desnecessária.
- Seu **Makefile** deve conter pelo menos as regras:
`$(NAME)`, `all`, `clean`, `fclean` e `re`.
- Compile seu código com `c++` usando as flags `-Wall` `-Wextra` `-Werror`.
- Seu código deve estar em conformidade com o **padrão C++ 98**. Então, ele ainda deve compilar se você adicionar a flag `-std=c++98`.
- Tente sempre codificar usando recursos `C++` quando disponíveis (por exemplo, escolha `<cstring>` em vez de `<string.h>`). Você está autorizado a usar funções `C`, mas sempre prefira suas versões `C++` se possível.
- Qualquer biblioteca externa e bibliotecas **Boost** são proibidas.

Capítulo III

Instruções para IA

● Contexto

Durante sua jornada de aprendizado, a IA pode auxiliar em diversas tarefas. Dedique tempo para explorar as várias capacidades das ferramentas de IA e como elas podem apoiar seu trabalho. No entanto, sempre as utilize com cautela e avalie criticamente os resultados. Seja código, documentação, ideias ou explicações técnicas, você nunca pode ter certeza absoluta de que sua pergunta foi bem formulada ou que o conteúdo gerado é preciso. Seus colegas são um recurso valioso para ajudá-lo a evitar erros e pontos cegos.

● Mensagem principal

- 👉 Use a IA para reduzir tarefas repetitivas ou tediosas.
- 👉 Desenvolva habilidades de prompt — tanto para codificação quanto para outras tarefas — que beneficiarão sua carreira futura.
- 👉 Aprenda como os sistemas de IA funcionam para melhor antecipar e evitar riscos comuns, vieses e questões éticas.
- 👉 Continue desenvolvendo habilidades técnicas e interpessoais trabalhando com seus colegas.
- 👉 Use apenas conteúdo gerado por IA que você entenda completamente e pelo qual possa se responsabilizar.

● Regras para o aluno:

- Você deve dedicar tempo para explorar as ferramentas de IA e entender como elas funcionam, para que possa usá-las eticamente e reduzir potenciais vieses.
- Você deve refletir sobre seu problema antes de criar o prompt — isso ajuda você a escrever prompts mais claros, detalhados e relevantes, usando vocabulário preciso.

- Você deve desenvolver o hábito de verificar, revisar, questionar e testar sistematicamente tudo o que for gerado por IA.
- Você deve sempre buscar revisão por pares — não confie apenas em sua própria validação.

● Resultados da fase:

- Desenvolver habilidades de prompt tanto para uso geral quanto para áreas específicas.
- Aumentar sua produtividade com o uso eficaz de ferramentas de IA.
- Continuar fortalecendo o pensamento computacional, resolução de problemas, adaptabilidade e colaboração.

● Comentários e exemplos:

- Você encontrará regularmente situações — exames, avaliações e mais — onde você deve demonstrar compreensão real. Esteja preparado, continue desenvolvendo suas habilidades técnicas e interpessoais.
- Explicar seu raciocínio e debater com colegas frequentemente revela lacunas em sua compreensão. Priorize a aprendizagem colaborativa.
- As ferramentas de IA geralmente carecem do seu contexto específico e tendem a fornecer respostas genéricas. Seus colegas, que compartilham seu ambiente, podem oferecer insights mais relevantes e precisos.
- Onde a IA tende a gerar a resposta mais provável, seus colegas podem fornecer perspectivas alternativas e nuances valiosas. Conte com eles como um ponto de verificação de qualidade.

✓ Boa prática:

Pergunto à IA: “Como testo uma função de ordenação?” Ela me dá algumas ideias. Eu as testo e reviso os resultados com um colega. Nós refinamos a abordagem juntos.

✗ Má prática:

Peço à IA para escrever uma função inteira, copio e colo no meu projeto. Durante a avaliação por pares, não consigo explicar o que ela faz ou porquê. Perco credibilidade — e reprovo no meu projeto.

✓ Boa prática:

Uso a IA para ajudar a projetar um analisador sintático. Então, analiso a lógica com um colega. Detectamos dois erros e reescrevemos juntos — melhor, mais limpo e totalmente compreendido.

✗ Má prática:

Deixo o Copilot gerar meu código para uma parte importante do meu projeto. Ele compila, mas não consigo explicar como ele lida com pipes. Durante a avaliação, não consigo justificar e reprovo no meu projeto.

Capítulo IV

Parte obrigatória

Nome do programa	ircserv
Arquivos para entregar	Makefile, *.{h, hpp}, *.cpp, *.tpp, *.ipp, um arquivo de configuração opcional
Makefile	NAME, all, clean, fclean, re
Argumentos	port: A porta de escuta password: A senha de conexão
Funções externas autorizadas	Tudo em C++ 98. socket, close, setsockopt, getsockname, getprotobyname, gethostbyname, getaddrinfo, freeaddrinfo, bind, connect, listen, accept, htons, htonl, ntohs, ntohl, inet_addr, inet_ntoa, inet_ntop, send, recv, signal, sigaction, sigemptyset, sigfillset, sigaddset, sigdelset, sigismember, lseek, fstat, fcntl, poll (ou equivalente)
Libft autorizada	n/a
Descrição	Um servidor IRC em C++ 98

Você deve desenvolver um servidor IRC usando o padrão C++ 98.

Você **não deve** desenvolver um cliente IRC.

Você **não deve** implementar a comunicação de servidor para servidor.

Seu executável será executado da seguinte forma:

```
./ircserv <port> <password>
```

- **port:** O número da porta na qual seu servidor IRC estará escutando as conexões IRC de entrada.
- **password:** A senha de conexão. Será necessária por qualquer cliente IRC que tente se conectar ao seu servidor.



Embora `poll()` seja mencionado no subject e na régua de avaliação, você pode usar qualquer equivalente, como `select()`, `kqueue()` ou `epoll()`.

IV.1 Requisitos

- O servidor deve ser capaz de lidar com vários clientes simultaneamente sem travar.
- Forking é proibido. Todas as operações de E/S devem ser **não bloqueantes**.
- Apenas 1 `poll()` (ou equivalente) pode ser usado para lidar com todas essas operações (leitura, gravação, mas também escuta e assim por diante).



Como você precisa usar descritores de arquivo não bloqueantes, é possível usar funções `read/recv` ou `write/send` sem `poll()` (ou equivalente), e seu servidor não ficaria bloqueando. No entanto, isso consumiria mais recursos do sistema. Portanto, se você tentar ler/recv ou gravar/enviar em qualquer descritor de arquivo sem usar `poll()` (ou equivalente), sua nota será 0.

- Existem vários clientes IRC. Você deve escolher um deles como uma **referência**. Seu cliente de referência será usado durante o processo de avaliação.
- Seu cliente de referência deve ser capaz de se conectar ao seu servidor sem encontrar nenhum erro.
- A comunicação entre o cliente e o servidor deve ser feita via TCP/IP (v4 ou v6).
- Usar seu cliente de referência com seu servidor deve ser semelhante a usá-lo com qualquer servidor IRC oficial. No entanto, você só precisa implementar os seguintes recursos:
 - Você deve ser capaz de autenticar, definir um apelido, um nome de usuário, participar de um canal, enviar e receber mensagens privadas usando seu cliente de referência.
 - Todas as mensagens enviadas de um cliente para um canal devem ser encaminhadas para todos os outros clientes que se juntaram ao canal.
 - Você deve ter **operadores** e usuários regulares.
 - Então, você deve implementar os comandos que são específicos para **operadores de canal**:
 - * KICK - Ejetar um cliente do canal
 - * INVITE - Convidar um cliente para um canal
 - * TOPIC - Alterar ou visualizar o tópico do canal
 - * MODE - Alterar o modo do canal:

- i: Definir/remover canal somente para convidados
 - t: Definir/remover as restrições do comando TOPIC para operadores de canal
 - k: Definir/remover a chave do canal (senha)
 - o: Dar/tirar o privilégio de operador de canal
 - l: Definir/remover o limite de usuários para o canal
- Claro, espera-se que você escreva um código limpo.

IV.2 Para MacOS apenas



Como o MacOS não implementa `write()` da mesma forma que outros sistemas operacionais Unix, você tem permissão para usar `fcntl()`. Você deve usar descritores de arquivo no modo não bloqueante para obter um comportamento semelhante ao de outros sistemas operacionais Unix.



No entanto, você só tem permissão para usar `fcntl()` da seguinte forma:
`fcntl(fd, F_SETFL, O_NONBLOCK);`
Qualquer outra flag é proibida.

IV.3 Exemplo de teste

Verifique todos os possíveis erros e problemas, como recebimento de dados parciais, baixa largura de banda, etc.

Para garantir que seu servidor processe corretamente todos os dados enviados a ele, o seguinte teste simples usando `nc` pode ser realizado:

```
\$> nc -C 127.0.0.1 6667
com^Dman^Dd
\$>
```

Use `ctrl+D` para enviar o comando em várias partes: `'com'`, depois `'man'`, depois `'d\n'`.

Para processar um comando, você deve primeiro agregar os pacotes recebidos para reconstruí-lo.

Capítulo V

Parte bônus

Aqui estão recursos adicionais que você pode adicionar ao seu servidor IRC para fazê-lo ficar mais próximo a um servidor IRC real:

- Lidar com transferência de arquivos.
- Um bot.



A parte bônus só será avaliada se a parte obrigatória for PERFEITA. Perfeita significa que a parte obrigatória foi feita integralmente e funciona sem mau funcionamento. Se você não passou em TODOS os requisitos obrigatórios, sua parte bônus não será avaliada de forma alguma.

Capítulo VI

Entrega e avaliação por pares

Entregue seu projeto no seu repositório `Git` como de costume. Apenas o trabalho dentro do seu repositório será avaliado durante a defesa. Não hesite em verificar novamente os nomes de seus arquivos para garantir que estejam corretos.

Você é encorajado a criar programas de teste para seu projeto, mesmo que eles **não sejam enviados ou avaliados**. Esses testes podem ser especialmente úteis para testar seu servidor durante a defesa, mas também o de seus colegas se você tiver que avaliar outro `ft_irc` um dia. De fato, você é livre para usar quaisquer testes que precisar durante o processo de avaliação.



Seu cliente de referência será usado durante o processo de avaliação.

Durante a avaliação, uma breve **modificação do projeto** pode ser ocasionalmente solicitada. Isso pode envolver uma pequena mudança de comportamento, algumas linhas de código para escrever ou reescrever, ou um recurso fácil de adicionar.

Embora esta etapa possa **não ser aplicável a todos os projetos**, você deve estar preparado para ela se for mencionada nas diretrizes de avaliação.

Esta etapa visa verificar sua compreensão real de uma parte específica do projeto. A modificação pode ser realizada em qualquer ambiente de desenvolvimento que você escolher (por exemplo, sua configuração usual), e deve ser viável em poucos minutos — a menos que um prazo específico seja definido como parte da avaliação.

Você pode, por exemplo, ser solicitado a fazer uma pequena atualização em uma função ou script, modificar uma exibição ou ajustar uma estrutura de dados para armazenar novas informações, etc.

Os detalhes (escopo, alvo, etc.) serão especificados nas **diretrizes de avaliação** e podem

variar de uma avaliação para outra para o mesmo projeto.



16D85ACC441674FBA2DF65190663F432222F81AA0248081A7C1C1823F7A96F0B74495
15056E97427E5B22F07132659EC8D88B574BD62C94BB654D5835AAD889B014E078705
709F6E02