CSC 470 Report for Project: 03

Full Name: Sam Blamo

Section: 01

1. Introduction

a. Goal

The primary objective of this project was to develop a network-based, multiplayer Battleship game using C++ programming. The game aimed to replicate the classic Battleship experience, where players place ships on a grid and take turns guessing their opponent's ship locations. The challenge was to create a functional and interactive game environment over a network, using client-server architecture.

b. Problems

The primary challenge of this project was to manage complex game states and synchronize them across multiple clients in a networked environment. Implementing a robust and efficient communication protocol between the server and clients was essential. Additionally, handling concurrency and ensuring consistent game state amidst simultaneous player actions added to the complexity. The difficulty of the task lay in the intricacies of socket

programming, multithreading, and the logic required to manage a multiplayer game environment.

2. Approach

The approach taken was to split the game into two main components: the server and the client. The server was responsible for maintaining the game state, managing client connections, and relaying messages between clients. The client was designed to provide a user interface for the player to interact with the game, send commands to the server, and receive updates.

This design was expected to work well in a local network environment where latency is minimal. The client-server model was chosen for its scalability and ease of managing game states centrally.

3. Methodology

a. Implementation Pieces

The project was divided into several key components:

1. Grid Management (grid.h): Implementing the game board for each player.
2. Network Communication (network_packet.h, server.cpp, test_client.cpp): Handling the exchange of data between the server and clients.

b. Grid Management

Implementations: The grid was implemented as a class with methods to initialize, update, and display the grid.

Choice: A class-based implementation was chosen for its encapsulation and ease of managing the game state.

c. Network Communication

Server (server.cpp): Implemented using socket programming.

- Choice: Single-threaded server with select() for handling multiple clients

Client (test_client.cpp): Implemented with multithreading to handle user input and network communication simultaneously.

- Choice: Multithreading was necessary to keep the user interface responsive while waiting for network messages

d. Difficulties and Choices

Turns Feature: Implementing turns in a networked environment was challenging. The solution involved tracking player turns on the server and synchronizing this information with clients.

Tracking Number of Players: To manage the game flow based on the number of connected players, the server needed to dynamically track active connections and update clients accordingly.

-

4. Results

a. Testing Objectives

Turn Management: Ensuring that turns were correctly managed and synchronized across clients.

Player Connection Management: Verifying that the server correctly handled varying numbers of players and their states.

b. Experiments

Varying Number of Players: Tested with different numbers of clients connected to the server to observe behavior and stability.

Turn Sequence Testing: Tested the turn-taking mechanism under normal and edge-case scenarios.

c. Outcomes

The results indicated that the game managed turns effectively and handled multiple players with stability. However, issues arose in edge cases, such as abrupt client disconnections.

5. Discussion

a. Opinions

The approach was largely successful in creating a functional multiplayer Battleship game. However, there is room for improvement in handling edge cases and enhancing the user interface.

b. Areas for improvement

Improved Turn Management: Implementing a more robust system to handle unexpected scenarios like client disconnections.

User Interface: Enhancing the client interface for a more engaging user experience.

c. Possible Features

Implementing additional game features like chat or advanced game statistics.

d. Learning Experience

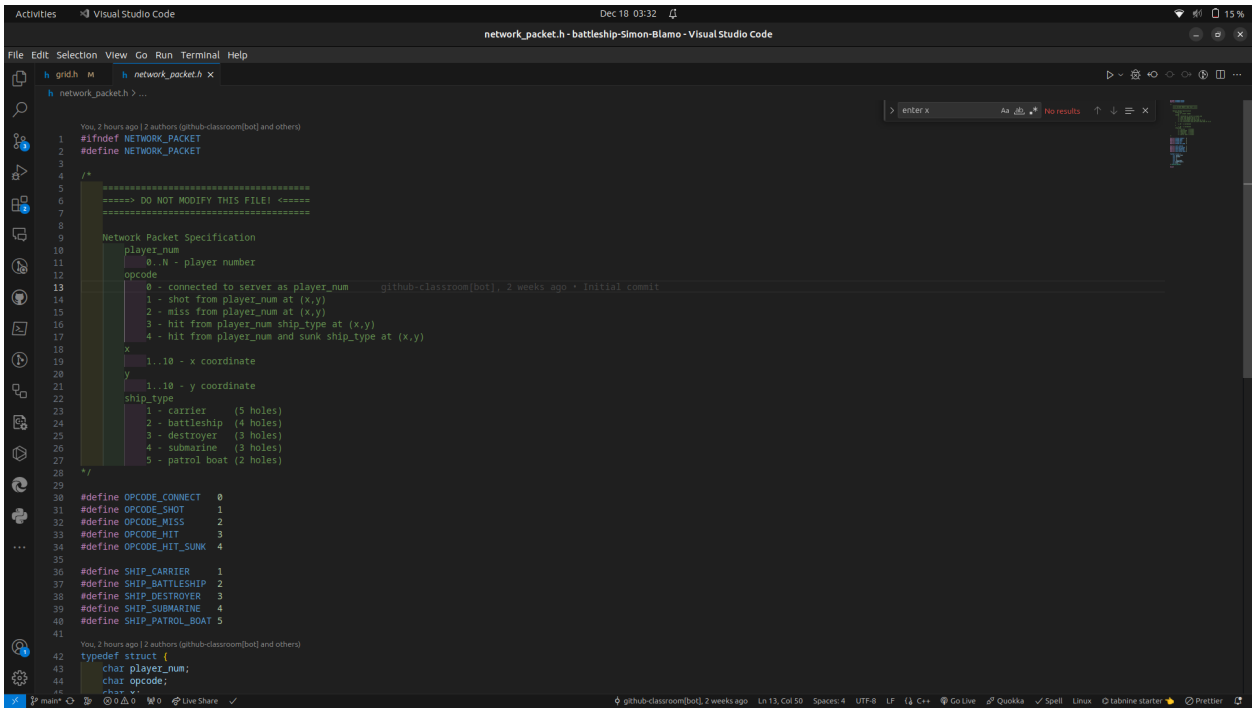
Although I was familiar with socket programming, this the first time using it to make a multiplayer game on a network. With this experience, I've learned the complex nature of real-time gaming.

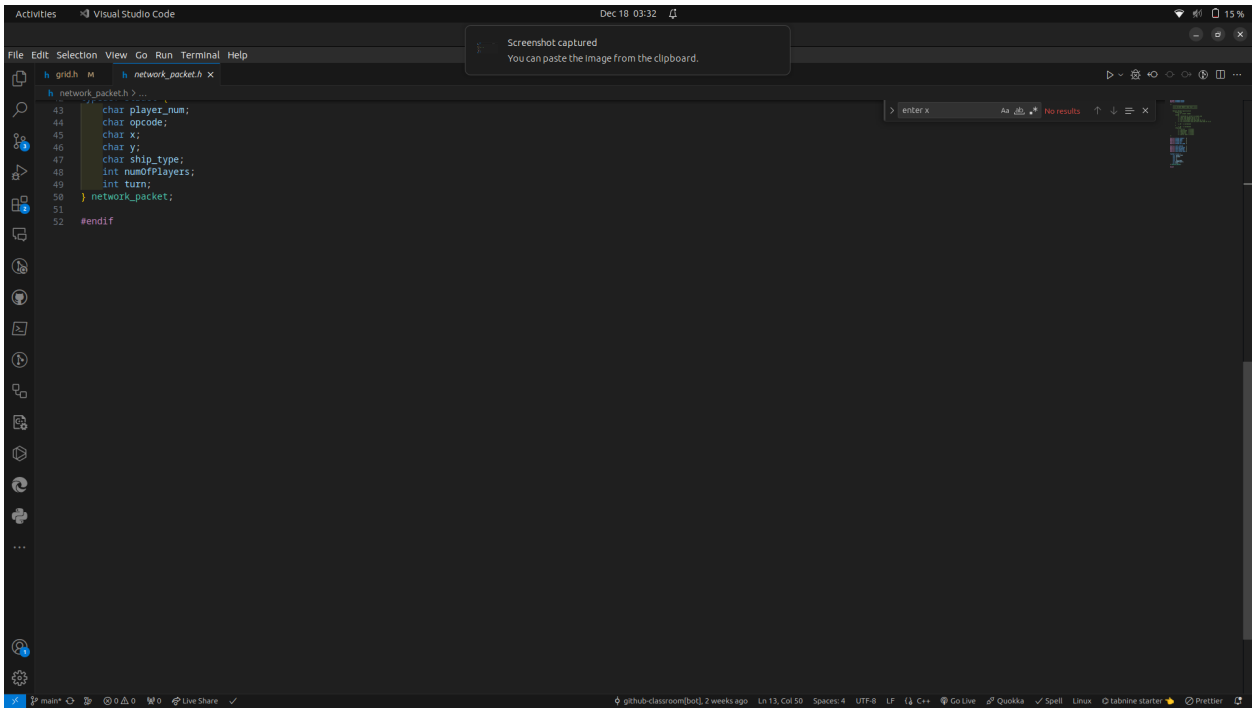
6. Conclusion

This Battleship game project demonstrates a successful implementation of a classic game in a networked environment using C++. The challenges faced, particularly in turn management and tracking player numbers, were significant learning experiences. The project underscores the importance of robust design and testing in networked applications. Future work can expand on this foundation to create a more feature-rich and scalable multiplayer game.

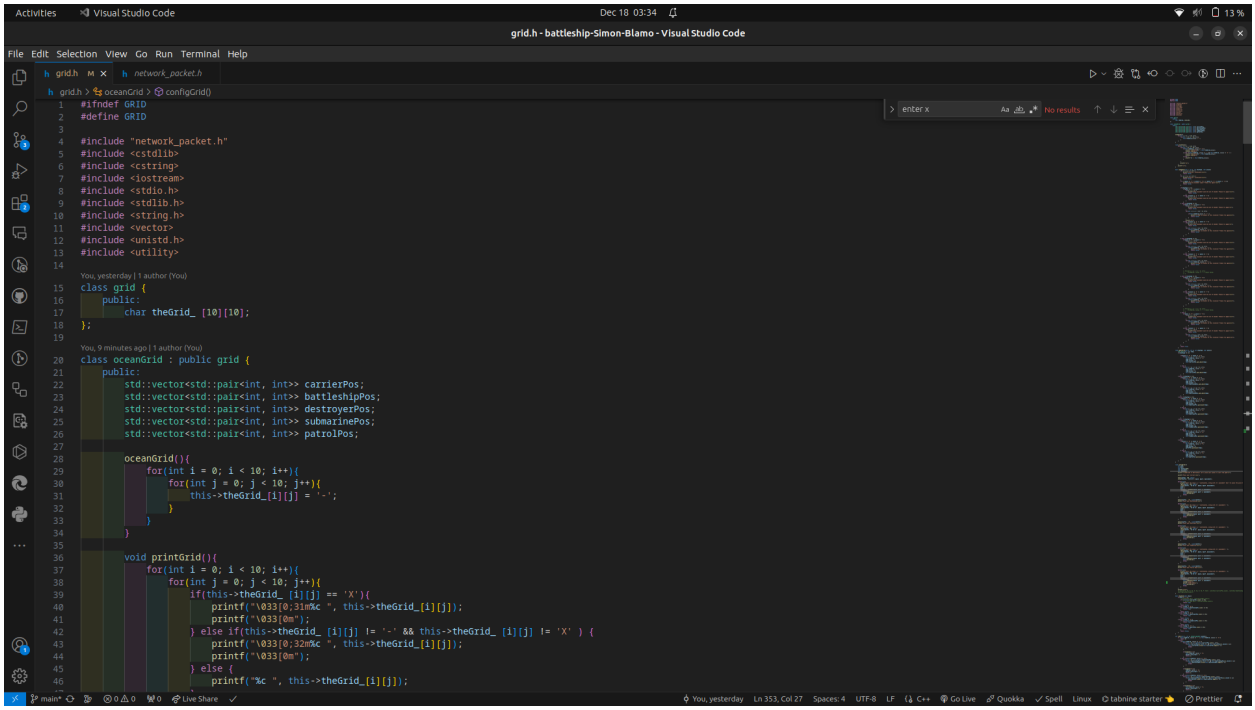
7. Source Code

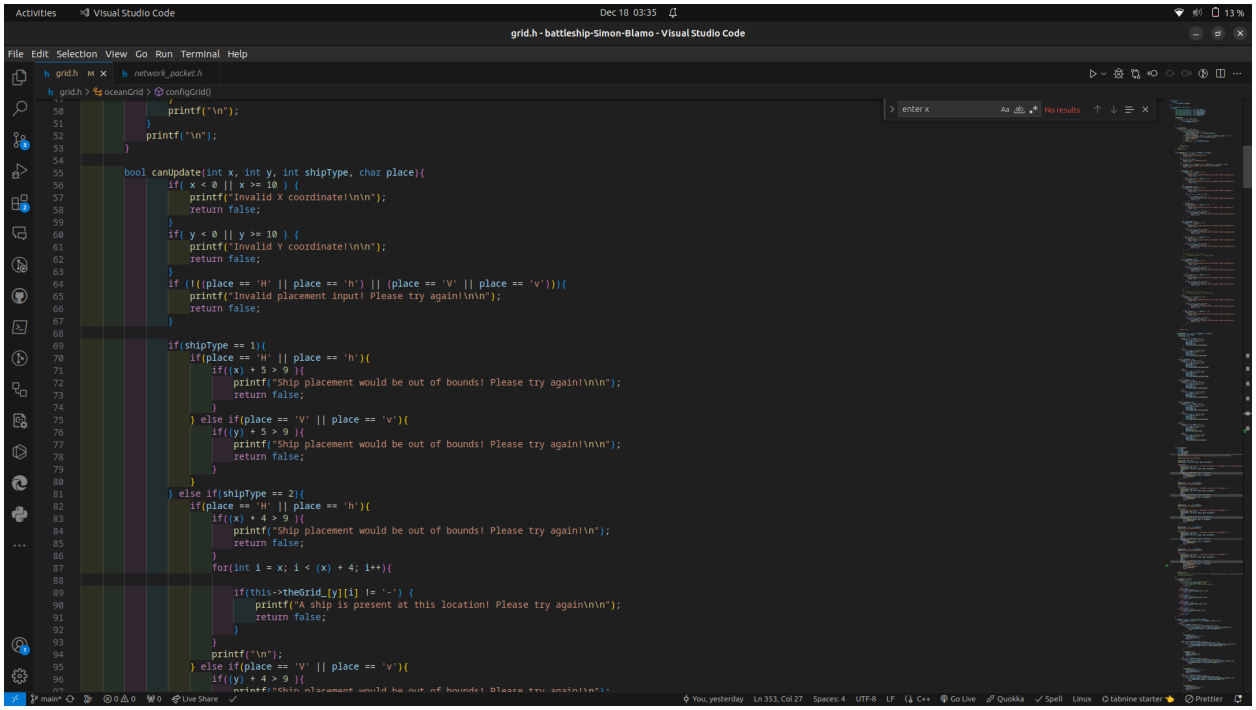
network_packet.h

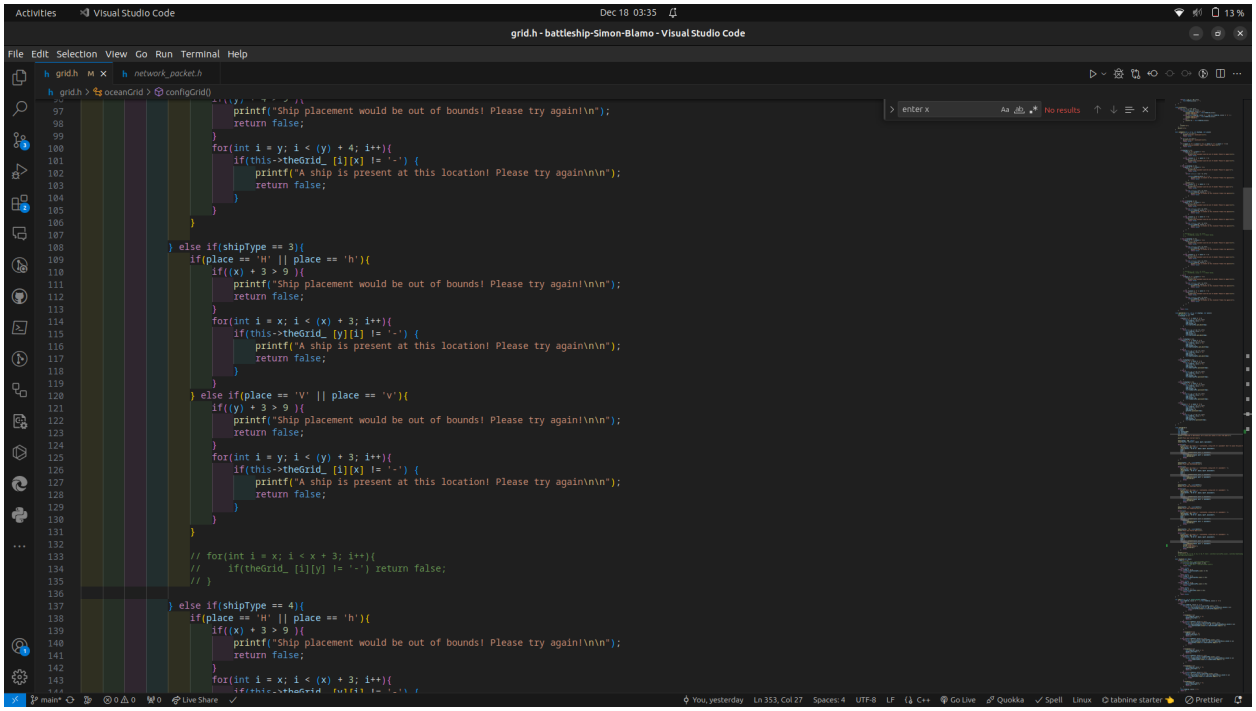

Activities
Visual Studio Code
Dec 18 03:32
network_packet.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
network_packet.h > ...
enter x
No results
You, 2 hours ago | 2 authors (github-classroom[bot] and others)
#ifndef NETWORK_PACKET
#define NETWORK_PACKET
/*
=====> DO NOT MODIFY THIS FILE! <=====
Network Packet Specification
player_num
0..N - player number
opcode
0 - connected to server as player_num
1 - shot from player_num at (x,y)
2 - miss from player_num at (x,y)
3 - hit from player_num ship_type at (x,y)
4 - hit from player_num and sunk ship_type at (x,y)
x
1..10 - x coordinate
y
1..10 - y coordinate
ship_type
1 - carrier (5 holes)
2 - battleship (4 holes)
3 - destroyer (3 holes)
4 - submarine (3 holes)
5 - patrol boat (2 holes)
*/
#define OPCODE_CONNECT 0
#define OPCODE_SHOT 1
#define OPCODE_MISS 2
#define OPCODE_HIT 3
#define OPCODE_HIT_SUNK 4
#define SHIP_CARRIER 1
#define SHIP_BATTLESHIP 2
#define SHIP_DESTROYER 3
#define SHIP_SUBMARINE 4
#define SHIP_PATROL_BOAT 5
You, 2 hours ago | 2 authors (github-classroom[bot] and others)
typedef struct {
char player_num;
char opcode;
github-classroom[bot], 2 weeks ago • Initial commit
main*
Live Share
github-classroom[bot], 2 weeks ago
Ln 13, Col 50
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

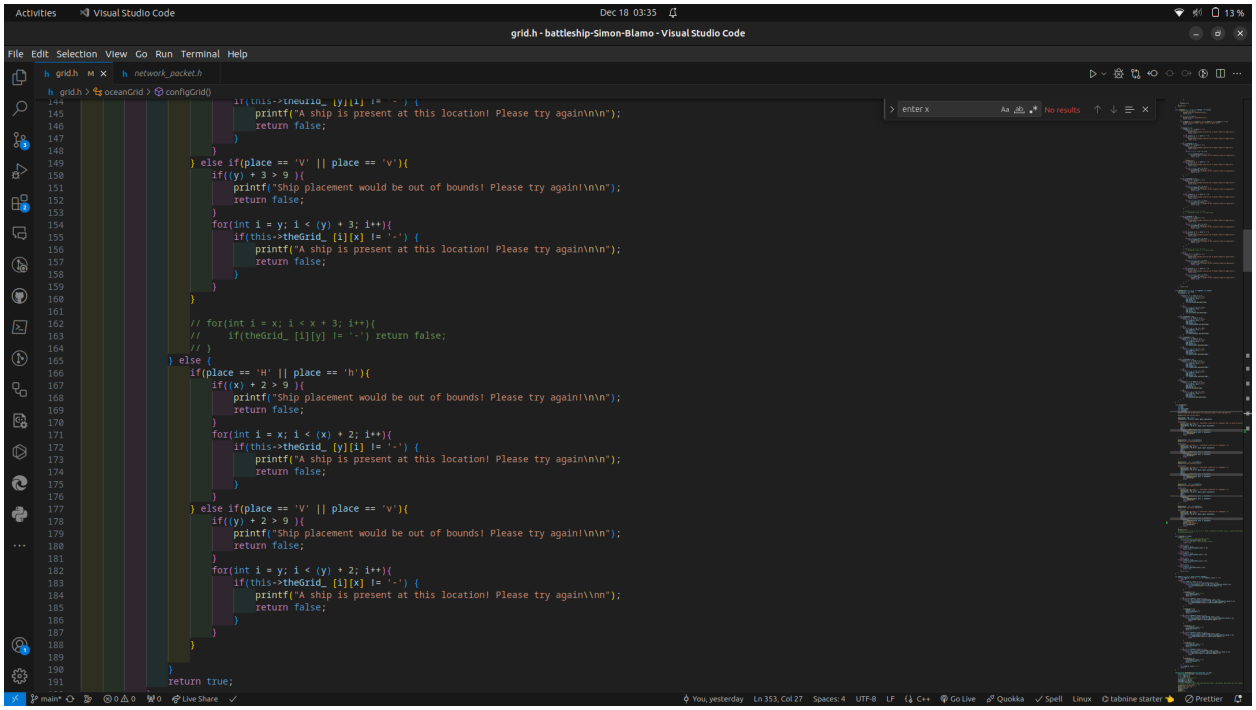

Activities
Visual Studio Code
Dec 18 03:32
Screenshot captured
You can paste the image from the clipboard.
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
network_packet.h > ...
enter x
No results
char player_num;
char opcode;
char x;
char y;
char ship_type;
int numOfPlayers;
int turn;
} network_packet;
#endif
main*
Live Share
github-classroom[bot], 2 weeks ago
Ln 13, Col 50
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

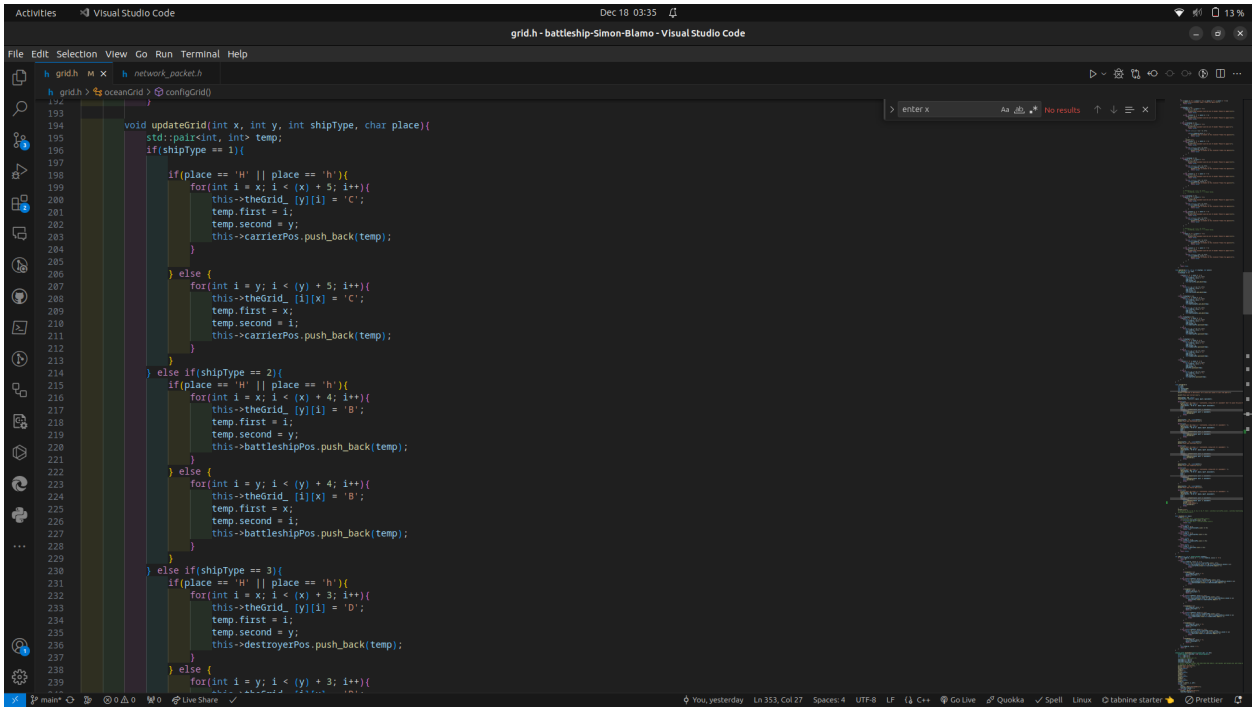
grid.h

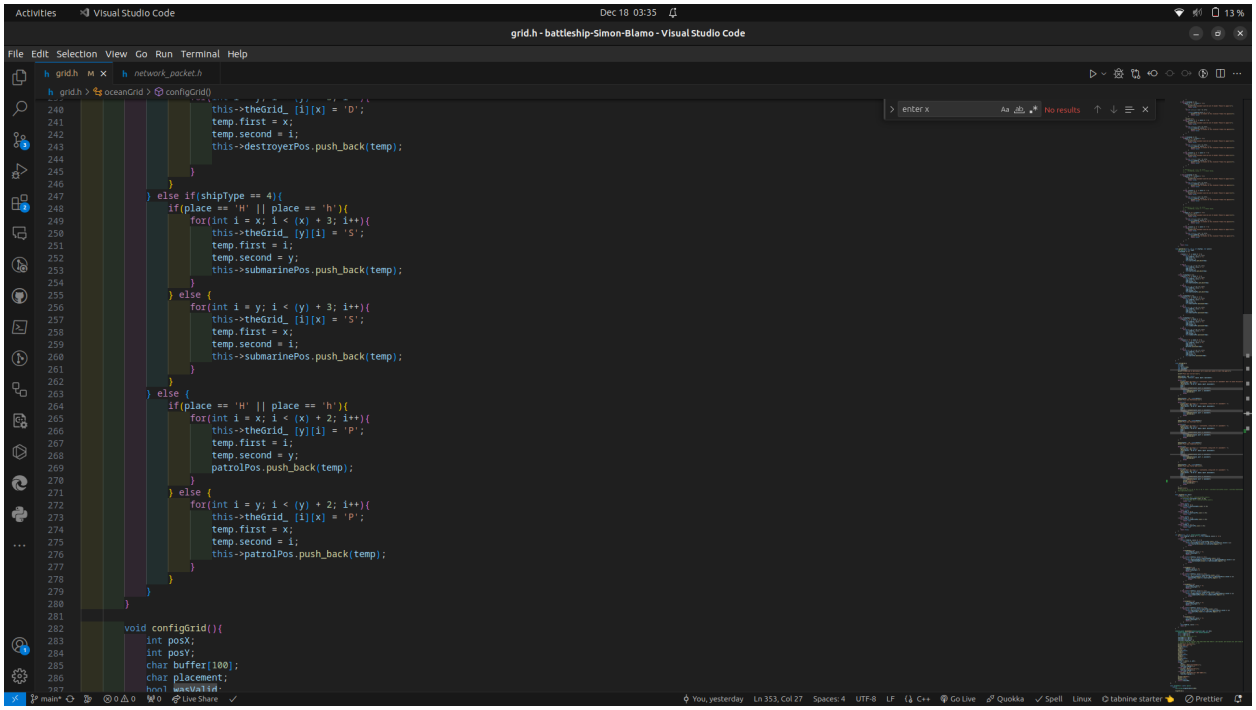

Activities
Visual Studio Code
Dec 18 03:34
13 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
#ifndef GRID
#define GRID
#include "network_packet.h"
#include <cstdlib>
#include <cstring>
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <vector>
#include <unistd.h>
#include <utility>
You, yesterday | 1 author (You)
class grid {
public:
char theGrid_ [10][10];
};
You, 9 minutes ago | 1 author (You)
class oceanGrid : public grid {
public:
std::vector<std::pair<int, int>> carrierPos;
std::vector<std::pair<int, int>> battleshipPos;
std::vector<std::pair<int, int>> destroyerPos;
std::vector<std::pair<int, int>> submarinePos;
std::vector<std::pair<int, int>> patrolPos;
oceanGrid(){
for(int i = 0; i < 10; i++){
for(int j = 0; j < 10; j++){
this->theGrid_[i][j] = '-';
}
}
}
void printGrid(){
for(int i = 0; i < 10; i++){
for(int j = 0; j < 10; j++){
if(this->theGrid_ [i][j] == 'X'){
printf("\033[0;31m%c ", this->theGrid_[i][j]);
printf("\033[0m");
} else if(this->theGrid_ [i][j] != '-' && this->theGrid_ [i][j] != 'X') {
printf("\033[0;32m%c ", this->theGrid_[i][j]);
printf("\033[0m");
} else {
printf("%c ", this->theGrid_[i][j]);
main*
0 0 0
Live Share
You, yesterday
Ln 353, Col 27
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

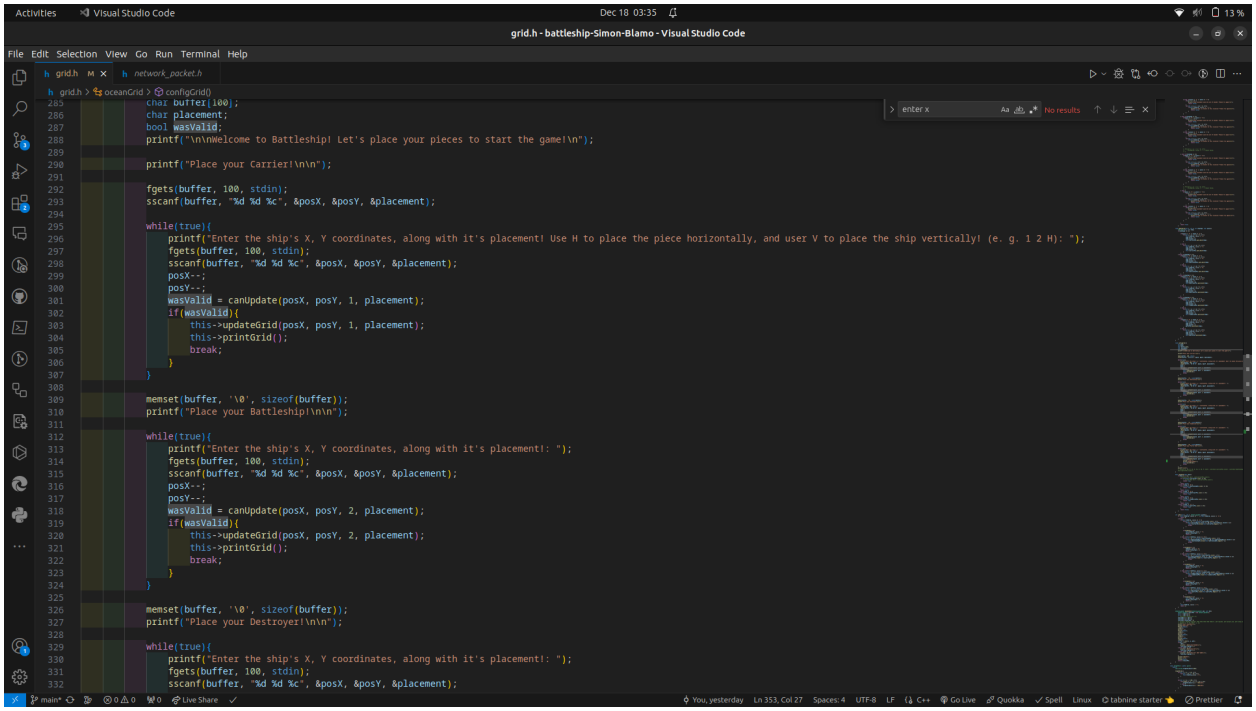

Activities
Visual Studio Code
Dec 18 03:35
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
printf("\n");
}
printf("\n");
}
bool canUpdate(int x, int y, int shipType, char place){
if(x < 0 || x >= 10) {
printf("Invalid X coordinate!\n\n");
return false;
}
if(y < 0 || y >= 10) {
printf("Invalid Y coordinate!\n\n");
return false;
}
if !((place == 'H' || place == 'h') || (place == 'V' || place == 'v'))){
printf("Invalid placement input! Please try again!\n\n");
return false;
}
if(shipType == 1){
if(place == 'H' || place == 'h'){
if((x) + 5 > 9){
printf("Ship placement would be out of bounds! Please try again!\n\n");
return false;
}
} else if(place == 'V' || place == 'v'){
if((y) + 5 > 9){
printf("Ship placement would be out of bounds! Please try again!\n\n");
return false;
}
}
} else if(shipType == 2){
if(place == 'H' || place == 'h'){
if((x) + 4 > 9){
printf("Ship placement would be out of bounds! Please try again!\n");
return false;
}
for(int i = x; i < (x) + 4; i++){
if(this->theGrid_[y][i] != '-') {
printf("A ship is present at this location! Please try again\n\n");
return false;
}
}
printf("\n");
} else if(place == 'V' || place == 'v'){
if((y) + 4 > 9){
main*
Live Share
You, yesterday
Ln 353, Col 27
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

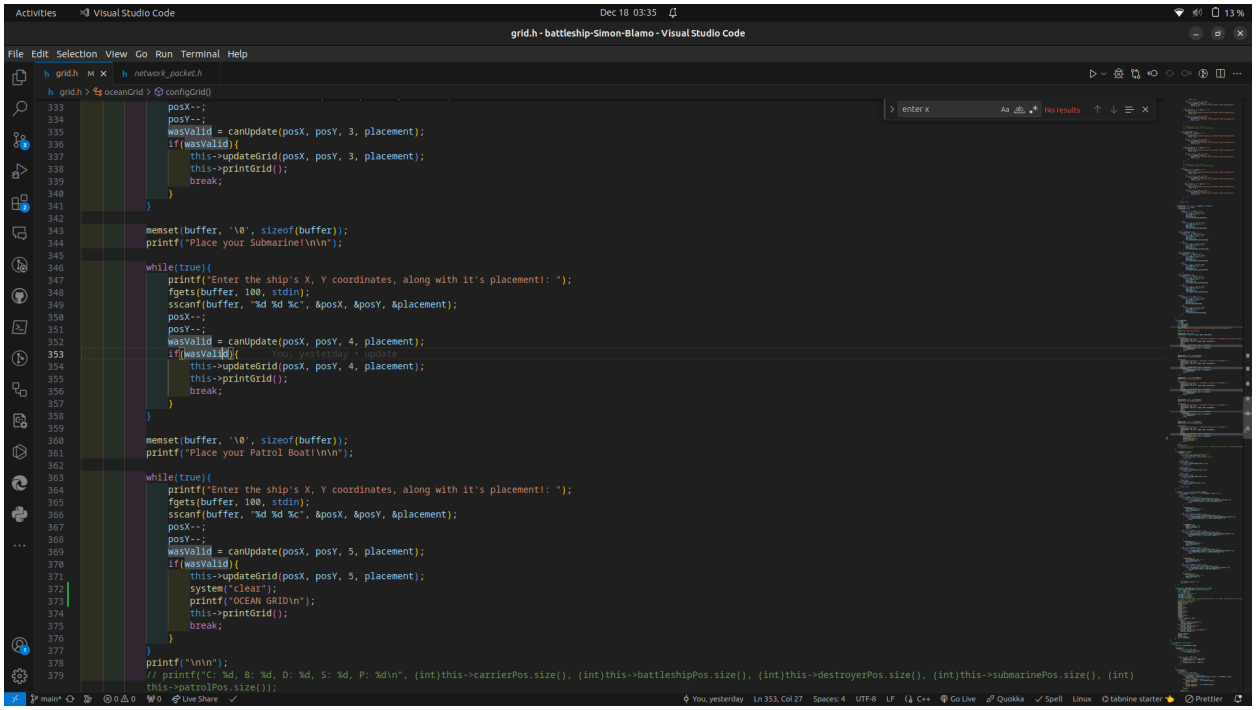

Activities
Visual Studio Code
Dec 18 03:35
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
printf("Ship placement would be out of bounds! Please try again!\n");
return false;
}
for(int i = y; i < (y) + 4; i++){
if(this->theGrid_ [i][x] != '-') {
printf("A ship is present at this location! Please try again\n\n");
return false;
}
}
}
} else if(shipType == 3){
if(place == 'H' || place == 'h'){
if((x) + 3 > 9){
printf("Ship placement would be out of bounds! Please try again!\n\n");
return false;
}
for(int i = x; i < (x) + 3; i++){
if(this->theGrid_ [y][i] != '-') {
printf("A ship is present at this location! Please try again\n\n");
return false;
}
}
} else if(place == 'V' || place == 'v'){
if((y) + 3 > 9){
printf("Ship placement would be out of bounds! Please try again!\n\n");
return false;
}
for(int i = y; i < (y) + 3; i++){
if(this->theGrid_ [i][x] != '-') {
printf("A ship is present at this location! Please try again\n\n");
return false;
}
}
}
// for(int i = x; i < x + 3; i++){
// if(theGrid_ [i][y] != '-') return false;
// }
} else if(shipType == 4){
if(place == 'H' || place == 'h'){
if((x) + 3 > 9){
printf("Ship placement would be out of bounds! Please try again!\n\n");
return false;
}
for(int i = x; i < (x) + 3; i++){
main*
Live Share
You, yesterday
Ln 353, Col 27
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

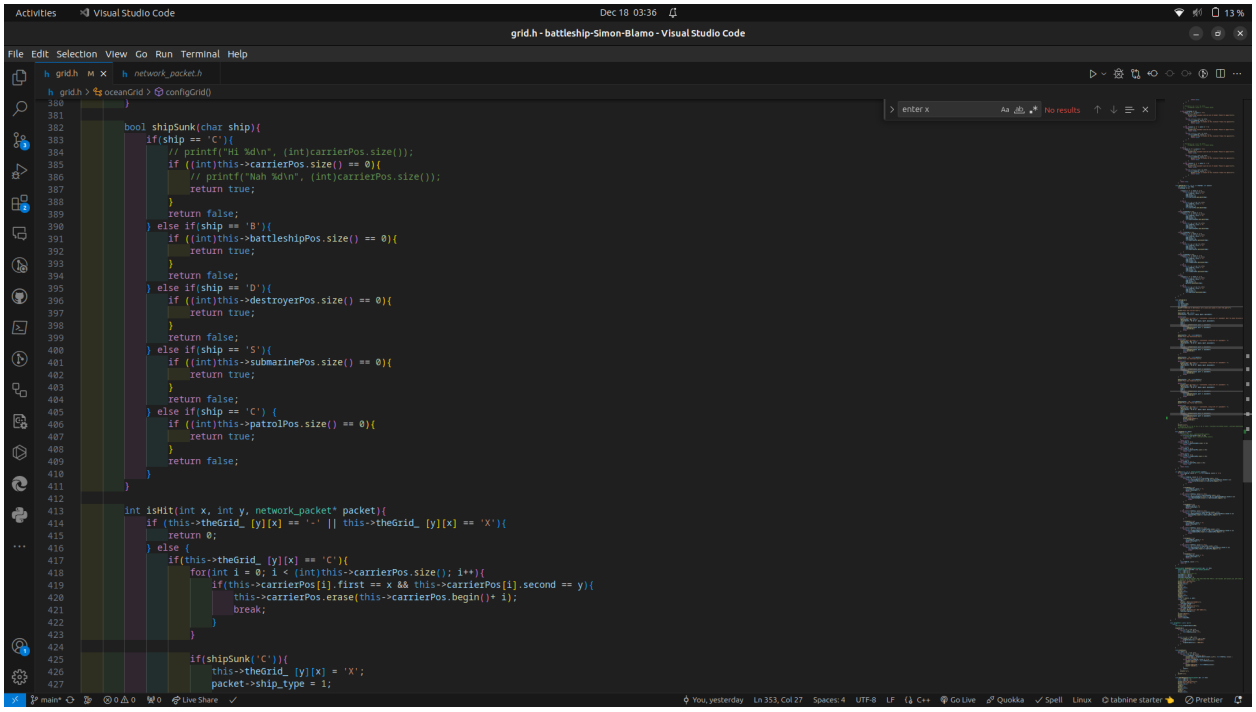

Activities
Visual Studio Code
Dec 18 03:35
13 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
printf("A ship is present at this location! Please try again\n\n");
return false;
printf("Ship placement would be out of bounds! Please try again!\n\n");
return false;
return true;
main*
Live Share
You, yesterday
Ln 353, Col 27
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

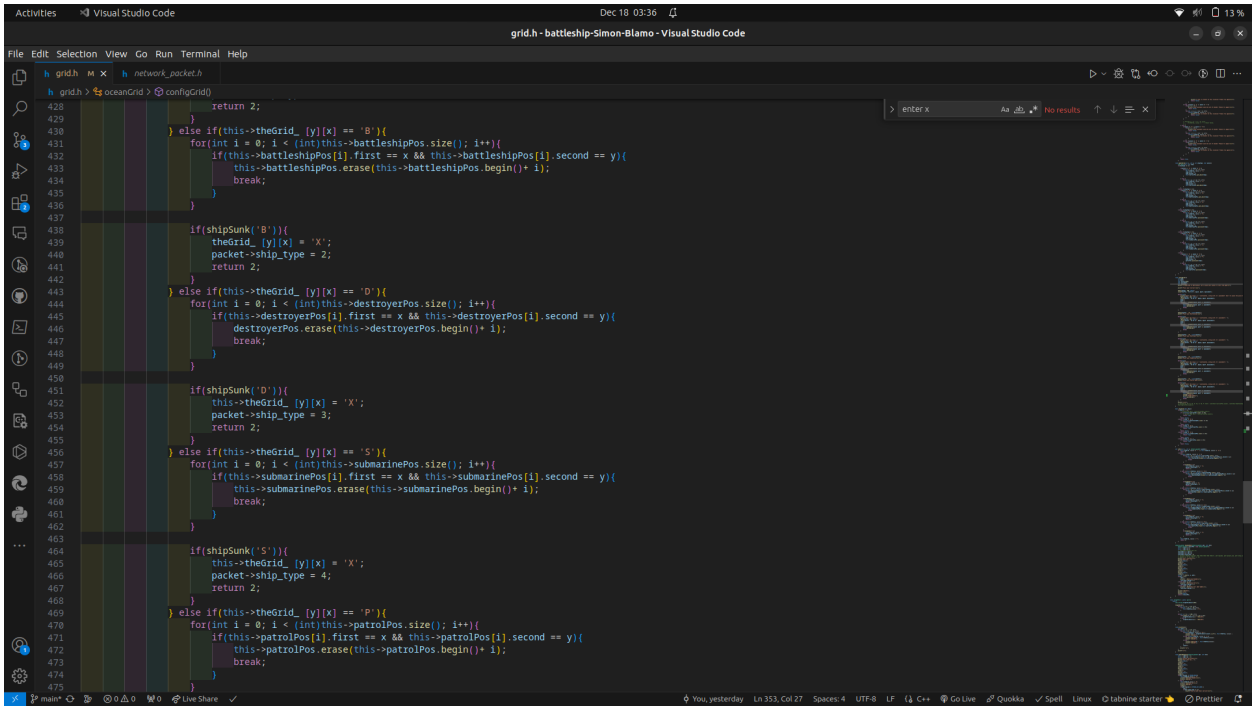

Activities
Visual Studio Code
Dec 18 03:35
13 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
void updateGrid(int x, int y, int shipType, char place){
std::pair<int, int> temp;
if(shipType == 1){
if(place == 'H' || place == 'h'){
for(int i = x; i < (x) + 5; i++){
this->theGrid_ [y][i] = 'C';
temp.first = i;
temp.second = y;
this->carrierPos.push_back(temp);
}
} else {
for(int i = y; i < (y) + 5; i++){
this->theGrid_ [i][x] = 'C';
temp.first = x;
temp.second = i;
this->carrierPos.push_back(temp);
}
}
} else if(shipType == 2){
if(place == 'H' || place == 'h'){
for(int i = x; i < (x) + 4; i++){
this->theGrid_ [y][i] = 'B';
temp.first = i;
temp.second = y;
this->battleshipPos.push_back(temp);
}
} else {
for(int i = y; i < (y) + 4; i++){
this->theGrid_ [i][x] = 'B';
temp.first = x;
temp.second = i;
this->battleshipPos.push_back(temp);
}
}
} else if(shipType == 3){
if(place == 'H' || place == 'h'){
for(int i = x; i < (x) + 3; i++){
this->theGrid_ [y][i] = 'D';
temp.first = i;
temp.second = y;
this->destroyerPos.push_back(temp);
}
} else {
for(int i = y; i < (y) + 3; i++){
main*
Live Share
You, yesterday
Ln 353, Col 27
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

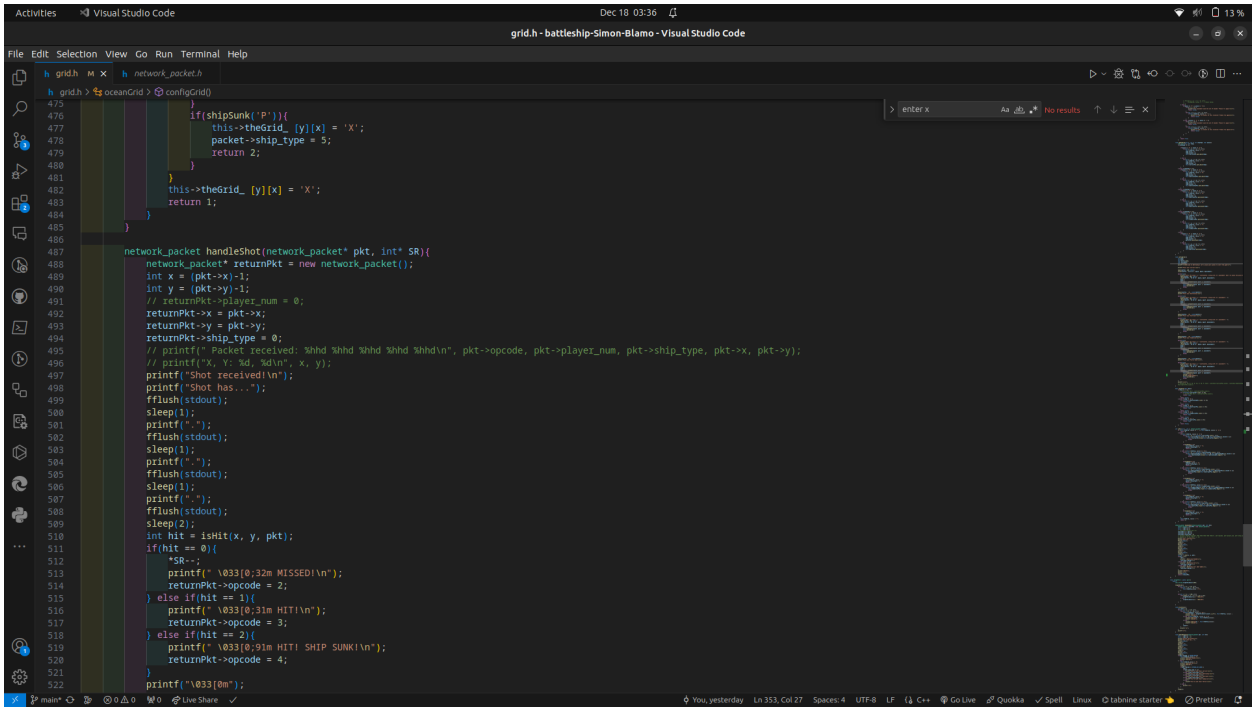

Activities
Visual Studio Code
Dec 18 03:35
13 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
this->theGrid_ [i][x] = 'D';
temp.first = x;
temp.second = i;
this->destroyerPos.push_back(temp);
} else if(shipType == 4){
if(place == 'H' || place == 'h'){
for(int i = x; i < (x) + 3; i++){
this->theGrid_ [y][i] = 'S';
temp.first = i;
temp.second = y;
this->submarinePos.push_back(temp);
this->theGrid_ [i][x] = 'S';
temp.first = x;
temp.second = i;
this->submarinePos.push_back(temp);
} else {
if(place == 'H' || place == 'h'){
for(int i = x; i < (x) + 2; i++){
this->theGrid_ [y][i] = 'P';
temp.first = i;
temp.second = y;
patrolPos.push_back(temp);
for(int i = y; i < (y) + 2; i++){
this->theGrid_ [i][x] = 'P';
temp.first = x;
temp.second = i;
this->patrolPos.push_back(temp);
void configGrid(){
int posX;
int posY;
char buffer[100];
char placement;
bool wasValid;
You, yesterday Ln 353, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier

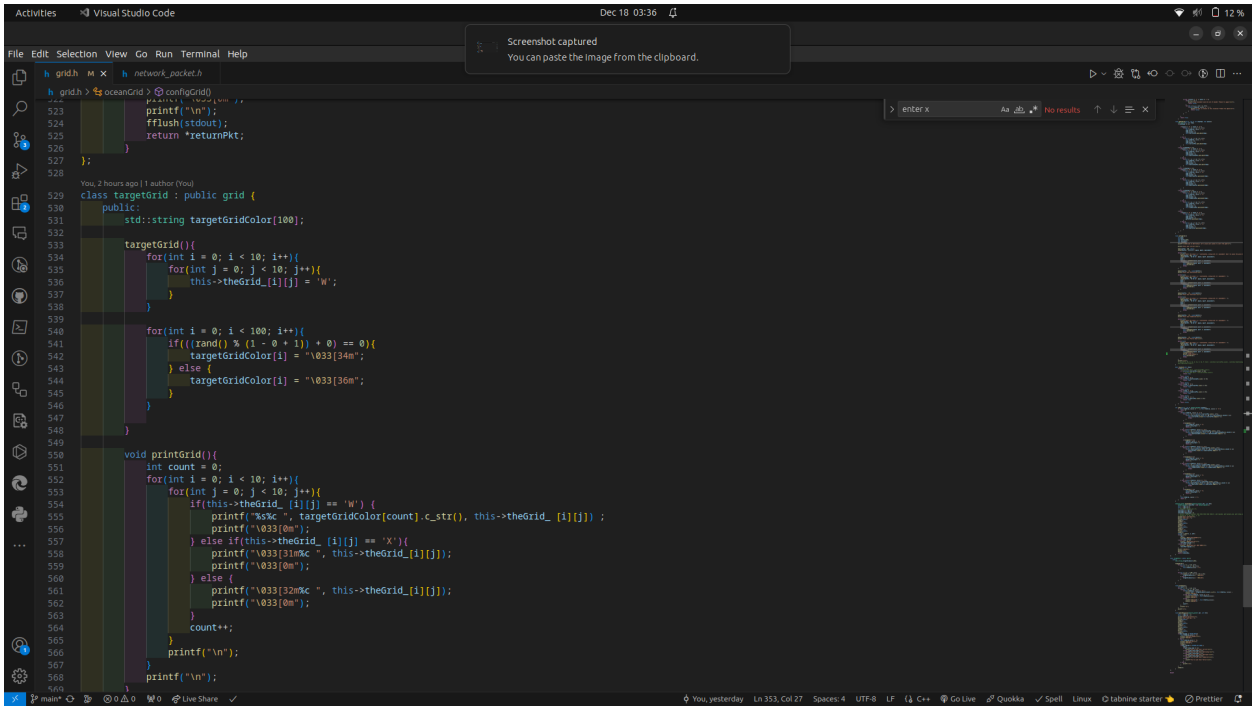

Activities
Visual Studio Code
Dec 18 03:35
13 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
char buffer[100];
char placement;
bool wasValid;
printf("\n\nWelcome to Battleship! Let's place your pieces to start the game!\n");
printf("Place your Carrier!\n\n");
fgets(buffer, 100, stdin);
sscanf(buffer, "%d %d %c", &posX, &posY, &placement);
while(true){
printf("Enter the ship's X, Y coordinates, along with it's placement! Use H to place the piece horizontally, and user V to place the ship vertically! (e. g. 1 2 H): ");
fgets(buffer, 100, stdin);
sscanf(buffer, "%d %d %c", &posX, &posY, &placement);
posX--;
posY--;
wasValid = canUpdate(posX, posY, 1, placement);
if(wasValid){
this->updateGrid(posX, posY, 1, placement);
this->printGrid();
break;
}
}
memset(buffer, '\0', sizeof(buffer));
printf("Place your Battleship!\n\n");
while(true){
printf("Enter the ship's X, Y coordinates, along with it's placement!: ");
fgets(buffer, 100, stdin);
sscanf(buffer, "%d %d %c", &posX, &posY, &placement);
posX--;
posY--;
wasValid = canUpdate(posX, posY, 2, placement);
if(wasValid){
this->updateGrid(posX, posY, 2, placement);
this->printGrid();
break;
}
}
memset(buffer, '\0', sizeof(buffer));
printf("Place your Destroyer!\n\n");
while(true){
printf("Enter the ship's X, Y coordinates, along with it's placement!: ");
fgets(buffer, 100, stdin);
sscanf(buffer, "%d %d %c", &posX, &posY, &placement);
You, yesterday Ln 353, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier


Activities
Visual Studio Code
Dec 18 03:35
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
posX--;
posY--;
wasValid = canUpdate(posX, posY, 3, placement);
if(wasValid){
this->updateGrid(posX, posY, 3, placement);
this->printGrid();
break;
}
}
memset(buffer, '\0', sizeof(buffer));
printf("Place your Submarine!\n\n");
while(true){
printf("Enter the ship's X, Y coordinates, along with it's placement!: ");
fgets(buffer, 100, stdin);
sscanf(buffer, "%d %d %c", &posX, &posY, &placement);
posX--;
posY--;
wasValid = canUpdate(posX, posY, 4, placement);
if(wasValid){
this->updateGrid(posX, posY, 4, placement);
this->printGrid();
break;
}
}
memset(buffer, '\0', sizeof(buffer));
printf("Place your Patrol Boat!\n\n");
while(true){
printf("Enter the ship's X, Y coordinates, along with it's placement!: ");
fgets(buffer, 100, stdin);
sscanf(buffer, "%d %d %c", &posX, &posY, &placement);
posX--;
posY--;
wasValid = canUpdate(posX, posY, 5, placement);
if(wasValid){
this->updateGrid(posX, posY, 5, placement);
system("clear");
printf("OCEAN GRID\n");
this->printGrid();
break;
}
}
printf("\n\n");
// printf("C: %d, B: %d, D: %d, S: %d, P: %d\n", (int)this->carrierPos.size(), (int)this->battleshipPos.size(), (int)this->destroyerPos.size(), (int)this->submarinePos.size(), (int)
this->patrolPos.size());
main*
You, yesterday Ln 353, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier

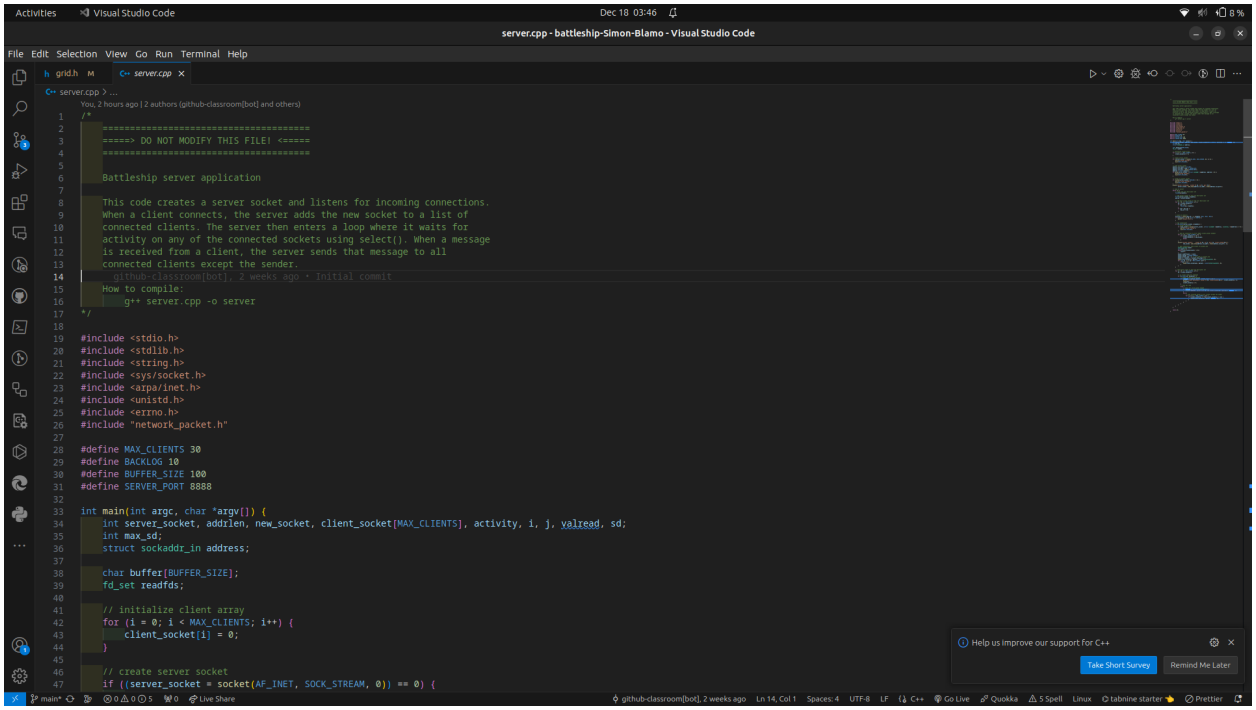

Activities
Visual Studio Code
Dec 18 03:36
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
}
bool shipSunk(char ship){
if(ship == 'C'){
// printf("Hi %d\n", (int)carrierPos.size());
if ((int)this->carrierPos.size() == 0){
// printf("Nah %d\n", (int)carrierPos.size());
return true;
}
return false;
} else if(ship == 'B'){
if ((int)this->battleshipPos.size() == 0){
return true;
}
return false;
} else if(ship == 'D'){
if ((int)this->destroyerPos.size() == 0){
return true;
}
return false;
} else if(ship == 'S'){
if ((int)this->submarinePos.size() == 0){
return true;
}
return false;
} else if(ship == 'C') {
if ((int)this->patrolPos.size() == 0){
return true;
}
return false;
}
}
int isHit(int x, int y, network_packet* packet){
if (this->theGrid_ [y][x] == '-' || this->theGrid_ [y][x] == 'X'){
return 0;
} else {
if(this->theGrid_ [y][x] == 'C'){
for(int i = 0; i < (int)this->carrierPos.size(); i++){
if(this->carrierPos[i].first == x && this->carrierPos[i].second == y){
this->carrierPos.erase(this->carrierPos.begin()+ i);
break;
}
}
if(shipSunk('C')){
this->theGrid_ [y][x] = 'X';
packet->ship_type = 1;
main*
You, yesterday Ln 353, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier

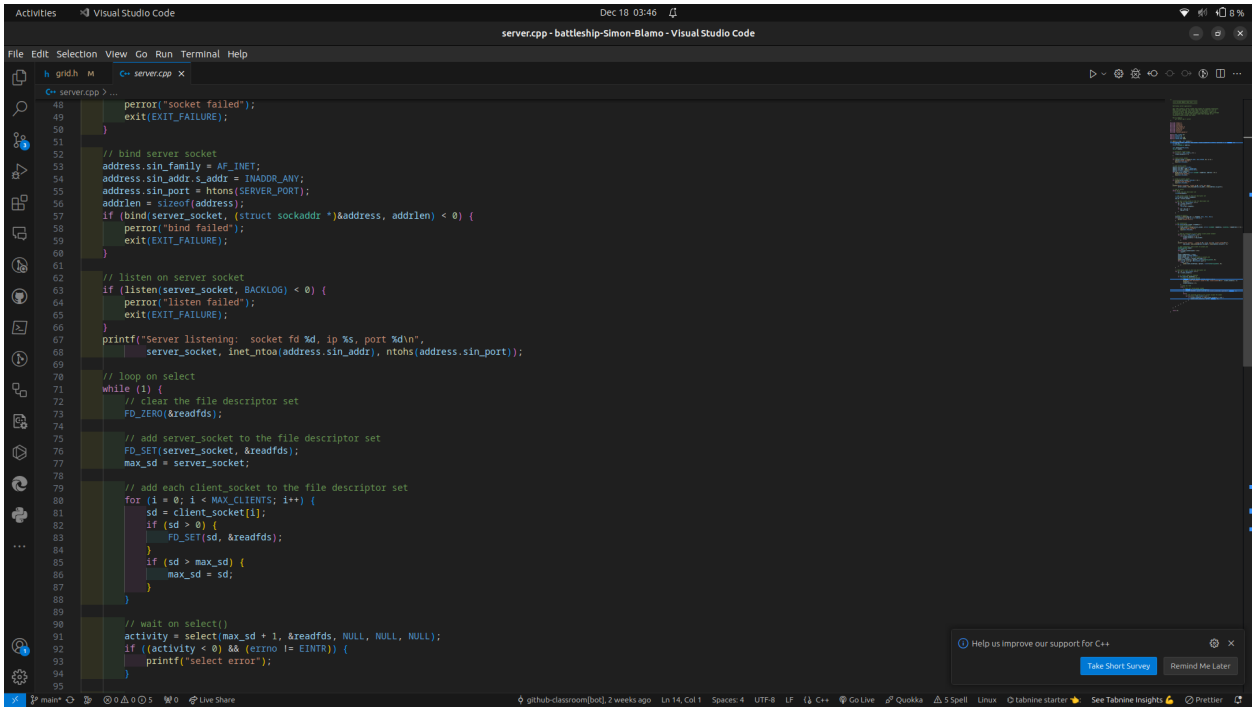

Activities
Visual Studio Code
Dec 18 03:36
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
return 2;
}
} else if(this->theGrid_ [y][x] == 'B'){
for(int i = 0; i < (int)this->battleshipPos.size(); i++){
if(this->battleshipPos[i].first == x && this->battleshipPos[i].second == y){
this->battleshipPos.erase(this->battleshipPos.begin()+ i);
break;
}
}
if(shipSunk('B')){
theGrid_ [y][x] = 'X';
packet->ship_type = 2;
return 2;
}
} else if(this->theGrid_ [y][x] == 'D'){
for(int i = 0; i < (int)this->destroyerPos.size(); i++){
if(this->destroyerPos[i].first == x && this->destroyerPos[i].second == y){
destroyerPos.erase(this->destroyerPos.begin()+ i);
break;
}
}
if(shipSunk('D')){
this->theGrid_ [y][x] = 'X';
packet->ship_type = 3;
return 2;
}
} else if(this->theGrid_ [y][x] == 'S'){
for(int i = 0; i < (int)this->submarinePos.size(); i++){
if(this->submarinePos[i].first == x && this->submarinePos[i].second == y){
this->submarinePos.erase(this->submarinePos.begin()+ i);
break;
}
}
if(shipSunk('S')){
this->theGrid_ [y][x] = 'X';
packet->ship_type = 4;
return 2;
}
} else if(this->theGrid_ [y][x] == 'P'){
for(int i = 0; i < (int)this->patrolPos.size(); i++){
if(this->patrolPos[i].first == x && this->patrolPos[i].second == y){
this->patrolPos.erase(this->patrolPos.begin()+ i);
break;
}
}
main*
You, yesterday Ln 353, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier

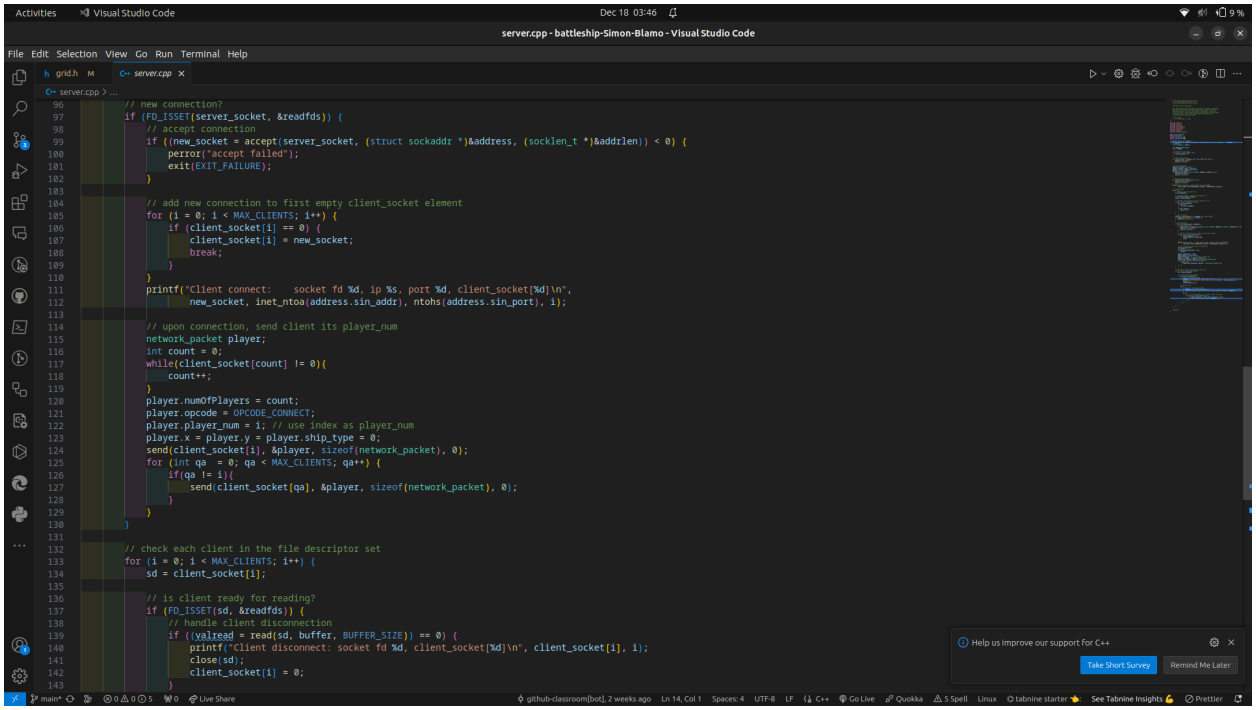

Activities
Visual Studio Code
Dec 18 03:36
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
}
if(shipSunk('P')){
this->theGrid_ [y][x] = 'X';
packet->ship_type = 5;
return 2;
}
}
this->theGrid_ [y][x] = 'X';
return 1;
}
}
network_packet handleShot(network_packet* pkt, int* SR){
network_packet* returnPkt = new network_packet();
int x = (pkt->x)-1;
int y = (pkt->y)-1;
// returnPkt->player_num = 0;
returnPkt->x = pkt->x;
returnPkt->y = pkt->y;
returnPkt->ship_type = 0;
// printf(" Packet received: %hhd %hhd %hhd %hhd %hhd\n", pkt->opcode, pkt->player_num, pkt->ship_type, pkt->x, pkt->y);
// printf("X, Y: %d, %d\n", x, y);
printf("Shot received!\n");
printf("Shot has...");
fflush(stdout);
sleep(1);
printf(".");
fflush(stdout);
sleep(1);
printf(".");
fflush(stdout);
sleep(1);
printf(".");
fflush(stdout);
sleep(2);
int hit = isHit(x, y, pkt);
if(hit == 0){
*SR--;
printf(" \033[0;32m MISSED!\n");
returnPkt->opcode = 2;
} else if(hit == 1){
printf(" \033[0;31m HIT!\n");
returnPkt->opcode = 3;
} else if(hit == 2){
printf(" \033[0;91m HIT! SHIP SUNK!\n");
returnPkt->opcode = 4;
}
printf("\033[0m");
main*
You, yesterday Ln 353, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier

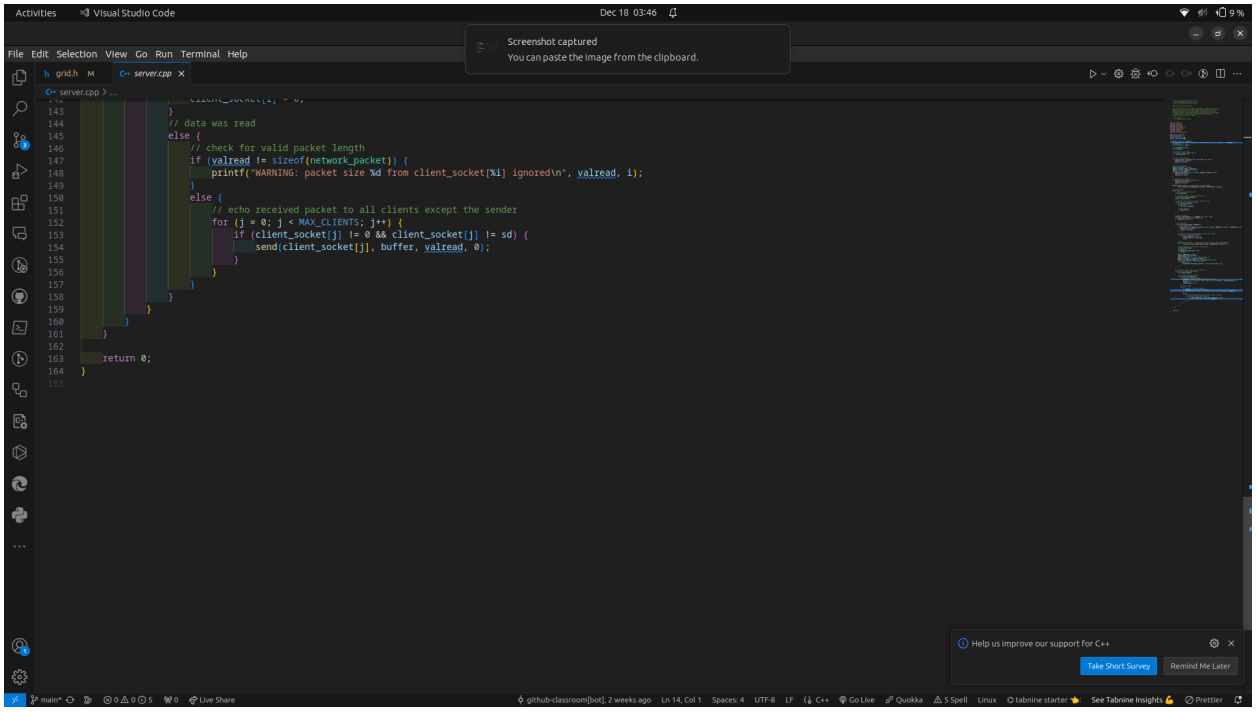

Activities
Visual Studio Code
Dec 18 03:36
12 %
Screenshot captured
You can paste the image from the clipboard.
File Edit Selection View Go Run Terminal Help
grid.h M
network_packet.h
grid.h > oceanGrid > configGrid()
enter x
No results
printf("\n");
fflush(stdout);
return *returnPkt;
};
You, 2 hours ago | 1 author (You)
class targetGrid : public grid {
public:
std::string targetGridColor[100];
targetGrid(){
for(int i = 0; i < 10; i++){
for(int j = 0; j < 10; j++){
this->theGrid_[i][j] = 'W';
}
}
for(int i = 0; i < 100; i++){
if((rand() % (1 - 0 + 1)) + 0) == 0){
targetGridColor[i] = "\033[34m";
} else {
targetGridColor[i] = "\033[36m";
}
}
}
void printGrid(){
int count = 0;
for(int i = 0; i < 10; i++){
for(int j = 0; j < 10; j++){
if(this->theGrid_ [i][j] == 'W') {
printf("%s%c ", targetGridColor[count].c_str(), this->theGrid_ [i][j]) ;
printf("\033[0m");
} else if(this->theGrid_ [i][j] == 'X'){
printf("\033[31m%c ", this->theGrid_[i][j]);
printf("\033[0m");
} else {
printf("\033[32m%c ", this->theGrid_[i][j]);
printf("\033[0m");
}
count++;
}
printf("\n");
}
printf("\n");
}
main*
Live Share
You, yesterday
Ln 353, Col 27
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier

server.cpp

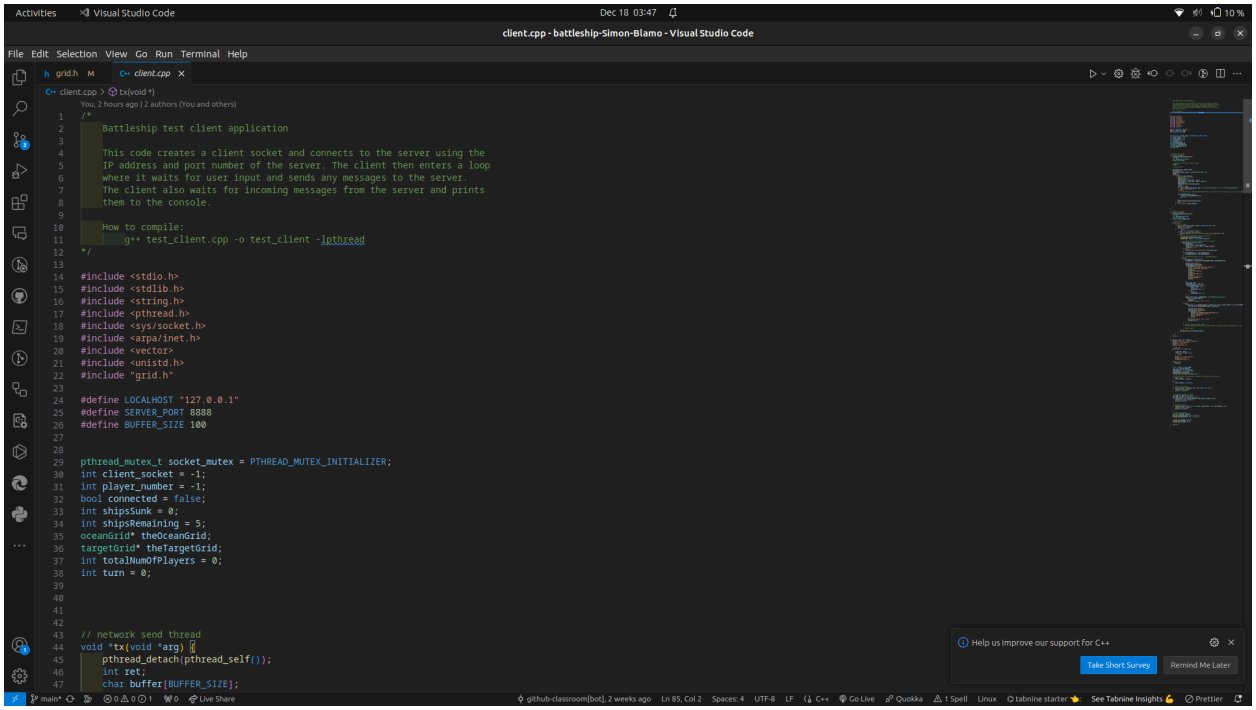

Activities
Visual Studio Code
Dec 18 03:46
server.cpp - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
server.cpp
server.cpp > ...
You, 2 hours ago | 2 authors (github-classroom[bot] and others)
/*
==========
=====> DO NOT MODIFY THIS FILE! <=====
==========
Battleship server application
This code creates a server socket and listens for incoming connections.
When a client connects, the server adds the new socket to a list of
connected clients. The server then enters a loop where it waits for
activity on any of the connected sockets using select(). When a message
is received from a client, the server sends that message to all
connected clients except the sender.
github-classroom[bot], 2 weeks ago • Initial commit
How to compile:
g++ server.cpp -o server
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <errno.h>
#include "network_packet.h"
#define MAX_CLIENTS 30
#define BACKLOG 10
#define BUFFER_SIZE 100
#define SERVER_PORT 8888
int main(int argc, char *argv[]) {
int server_socket, addrlen, new_socket, client_socket[MAX_CLIENTS], activity, i, j, valread, sd;
int max_sd;
struct sockaddr_in address;
char buffer[BUFFER_SIZE];
fd_set readfds;
// initialize client array
for (i = 0; i < MAX_CLIENTS; i++) {
client_socket[i] = 0;
}
// create server socket
if ((server_socket = socket(AF_INET, SOCK_STREAM, 0)) == 0) {
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
Live Share
github-classroom[bot], 2 weeks ago
Ln 14, Col 1
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
5 Spell
Linux
tabnine starter
Prettier

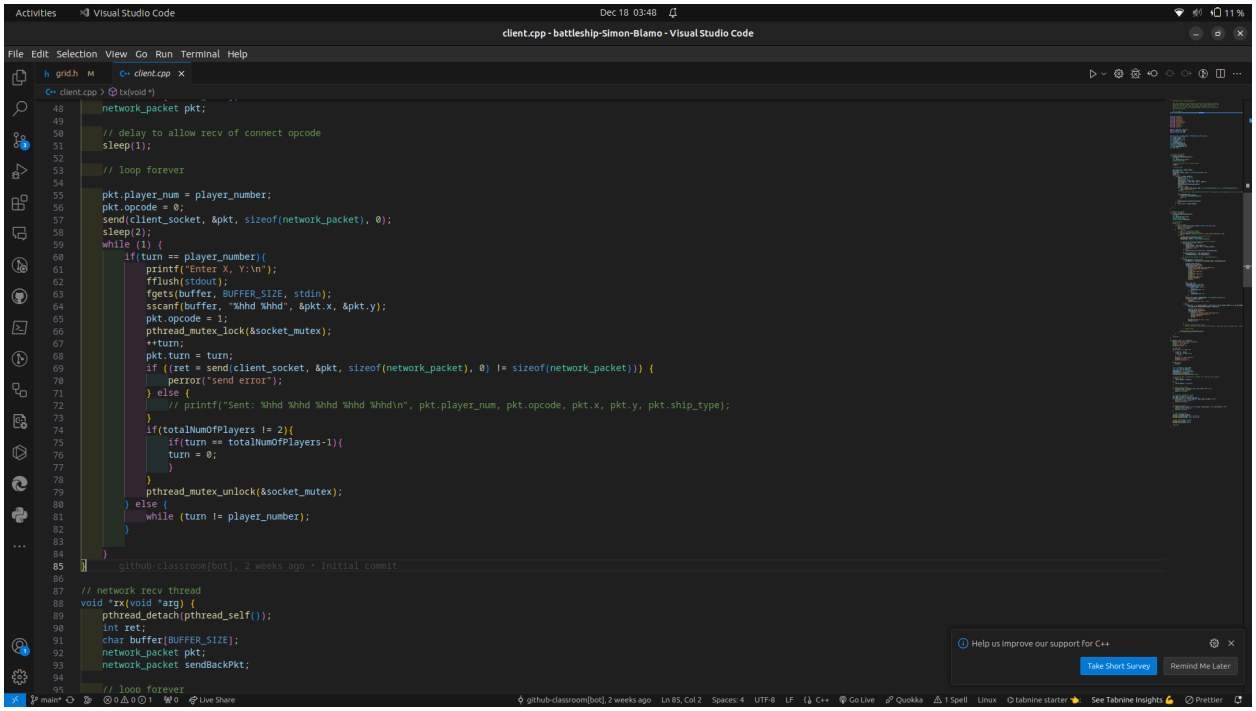

Activities
Visual Studio Code
Dec 18 03:46
server.cpp - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
server.cpp
server.cpp > ...
perror("socket failed");
exit(EXIT_FAILURE);
}
// bind server socket
address.sin_family = AF_INET;
address.sin_addr.s_addr = INADDR_ANY;
address.sin_port = htons(SERVER_PORT);
addrlen = sizeof(address);
if (bind(server_socket, (struct sockaddr *)&address, addrlen) < 0) {
perror("bind failed");
exit(EXIT_FAILURE);
}
// listen on server socket
if (listen(server_socket, BACKLOG) < 0) {
perror("listen failed");
exit(EXIT_FAILURE);
}
printf("Server listening: socket fd %d, ip %s, port %d\n",
server_socket, inet_ntoa(address.sin_addr), ntohs(address.sin_port));
// loop on select
while (1) {
// clear the file descriptor set
FD_ZERO(&readfds);
// add server_socket to the file descriptor set
FD_SET(server_socket, &readfds);
max_sd = server_socket;
// add each client_socket to the file descriptor set
for (i = 0; i < MAX_CLIENTS; i++) {
sd = client_socket[i];
if (sd > 0) {
FD_SET(sd, &readfds);
}
if (sd > max_sd) {
max_sd = sd;
}
}
// wait on select()
activity = select(max_sd + 1, &readfds, NULL, NULL, NULL);
if ((activity < 0) && (errno != EINTR)) {
printf("select error");
}
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
Live Share
github-classroom[bot], 2 weeks ago
Ln 14, Col 1
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
5 Spell
Linux
tabnine starter
See Tabnine Insights
Prettier

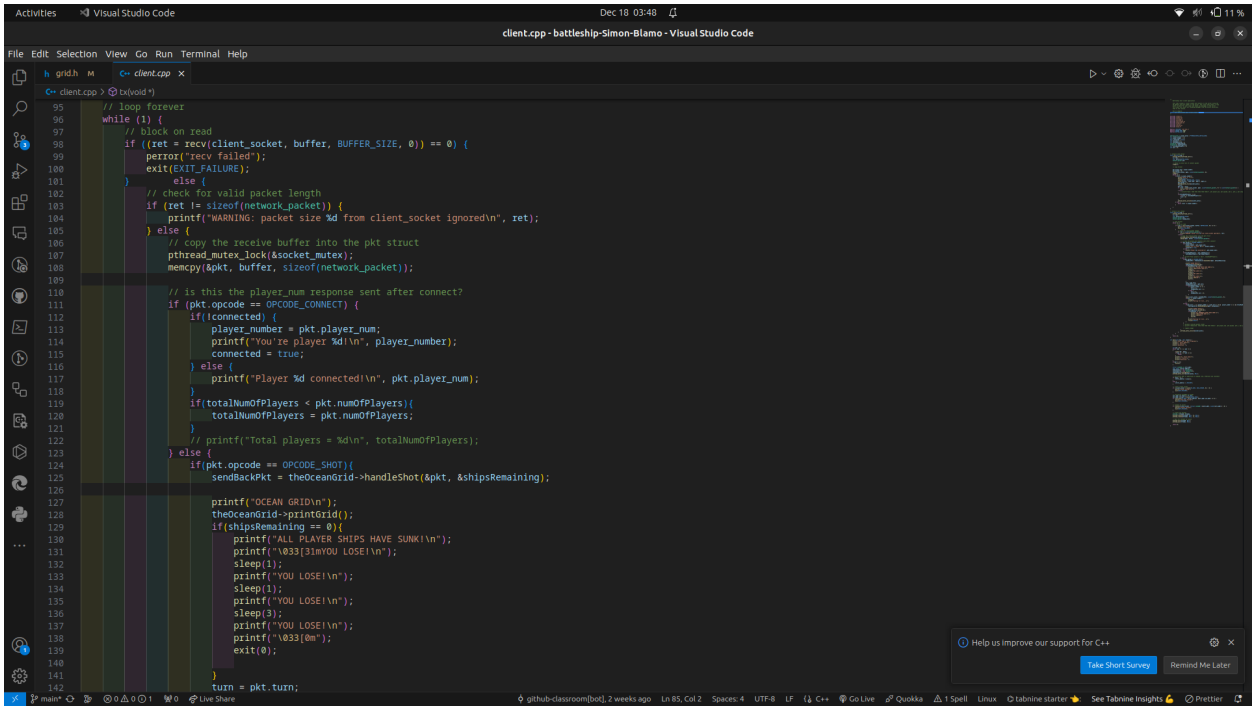

Activities
Visual Studio Code
Dec 18 03:46
server.cpp - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
server.cpp
server.cpp > ...
// new connection?
if (FD_ISSET(server_socket, &readfds)) {
// accept connection
if ((new_socket = accept(server_socket, (struct sockaddr *)&address, (socklen_t *)&addrlen)) < 0) {
perror("accept failed");
exit(EXIT_FAILURE);
}
// add new connection to first empty client_socket element
for (i = 0; i < MAX_CLIENTS; i++) {
if (client_socket[i] == 0) {
client_socket[i] = new_socket;
break;
}
}
printf("Client connect: socket fd %d, ip %s, port %d, client_socket[%d]\n",
new_socket, inet_ntoa(address.sin_addr), ntohs(address.sin_port), i);
// upon connection, send client its player_num
network_packet player;
int count = 0;
while(client_socket[count] != 0){
count++;
}
player.numOfPlayers = count;
player.opcode = OPCODE_CONNECT;
player.player_num = i; // use index as player_num
player.x = player.y = player.ship_type = 0;
send(client_socket[i], &player, sizeof(network_packet), 0);
for (int qa = 0; qa < MAX_CLIENTS; qa++) {
if(qa != i){
send(client_socket[qa], &player, sizeof(network_packet), 0);
}
}
}
// check each client in the file descriptor set
for (i = 0; i < MAX_CLIENTS; i++) {
sd = client_socket[i];
// is client ready for reading?
if (FD_ISSET(sd, &readfds)) {
// handle client disconnection
if ((valread = read(sd, buffer, BUFFER_SIZE)) == 0) {
printf("Client disconnect: socket fd %d, client_socket[%d]\n", client_socket[i], i);
close(sd);
client_socket[i] = 0;
}
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
github-classroom[bot], 2 weeks ago
Ln 14, Col 1
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
5 Spell
Linux
tabnine starter
See Tabnine Insights
Prettier

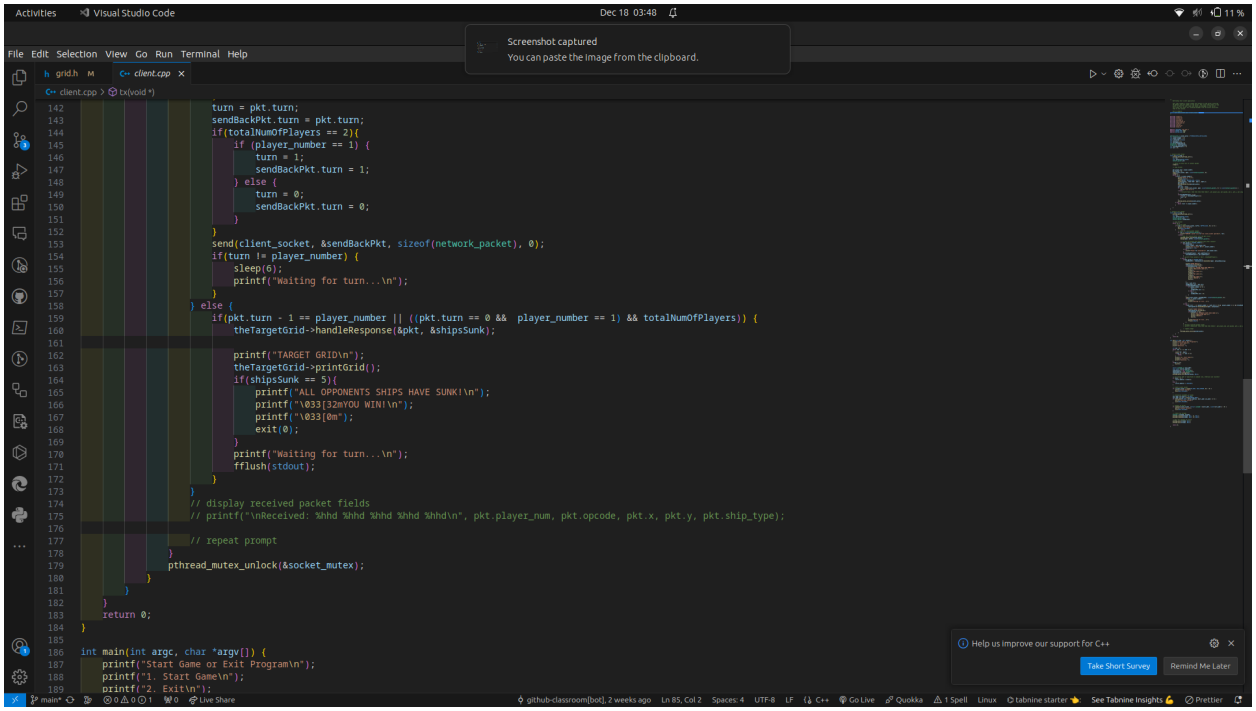

Activities
Visual Studio Code
Dec 18 03:46
Screenshot captured
You can paste the image from the clipboard.
File Edit Selection View Go Run Terminal Help
grid.h M
server.cpp
server.cpp > ...
}
// data was read
else {
// check for valid packet length
if (valread != sizeof(network_packet)) {
printf("WARNING: packet size %d from client_socket[%i] ignored\n", valread, i);
}
else {
// echo received packet to all clients except the sender
for (j = 0; j < MAX_CLIENTS; j++) {
if (client_socket[j] != 0 && client_socket[j] != sd) {
send(client_socket[j], buffer, valread, 0);
}
}
}
}
}
}
}
return 0;
}
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
github-classroom[bot], 2 weeks ago
Ln 14, Col 1
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
5 Spell
Linux
tabnine starter
See Tabnine Insights
Prettier

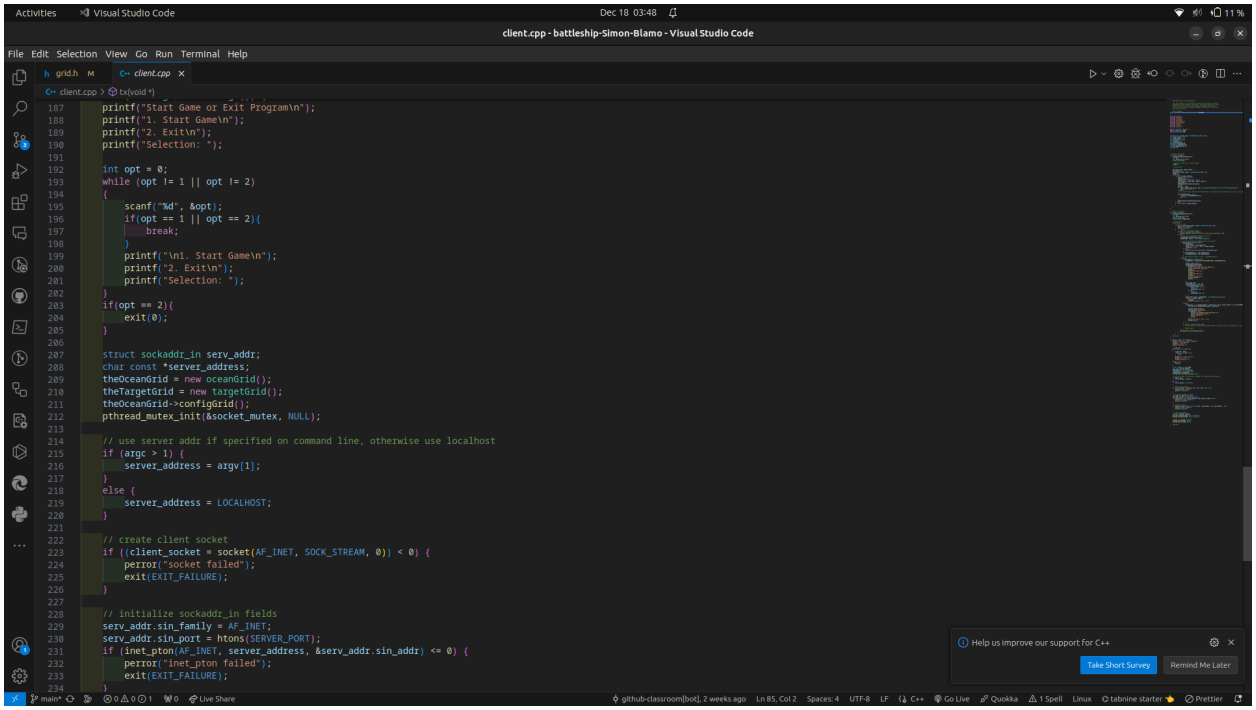
client.cpp

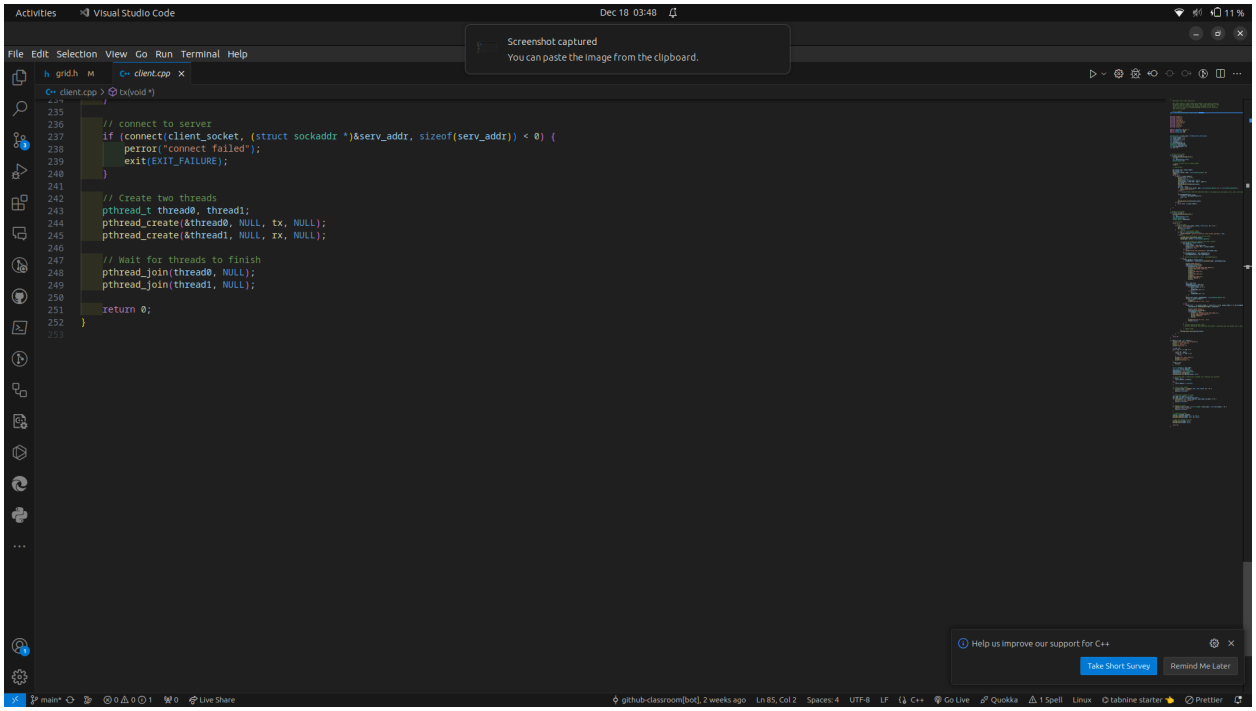

Activities
Visual Studio Code
Dec 18 03:47
client.cpp - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
client.cpp
client.cpp > tx(void *)
You, 2 hours ago | 2 authors (You and others)
/*
Battleship test client application
This code creates a client socket and connects to the server using the
IP address and port number of the server. The client then enters a loop
where it waits for user input and sends any messages to the server.
The client also waits for incoming messages from the server and prints
them to the console.
How to compile:
g++ test_client.cpp -o test_client -lpthread
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <pthread.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <vector>
#include <unistd.h>
#include "grid.h"
#define LOCALHOST "127.0.0.1"
#define SERVER_PORT 8888
#define BUFFER_SIZE 100
pthread_mutex_t socket_mutex = PTHREAD_MUTEX_INITIALIZER;
int client_socket = -1;
int player_number = -1;
bool connected = false;
int shipsSunk = 0;
int shipsRemaining = 5;
oceanGrid* theOceanGrid;
targetGrid* theTargetGrid;
int totalNumOfPlayers = 0;
int turn = 0;
// network send thread
void *tx(void *arg) {
pthread_detach(pthread_self());
int ret;
char buffer[BUFFER_SIZE];
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
github-classroom[bot], 2 weeks ago
Ln 85, Col 2
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
1 Spell
Linux
tabnine starter
See Tabnine Insights
Prettier


Activities
Visual Studio Code
Dec 18 03:48
client.cpp - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
client.cpp
client.cpp > tx(void *)
network_packet pkt;
// delay to allow recv of connect opcode
sleep(1);
// loop forever
pkt.player_num = player_number;
pkt.opcode = 0;
send(client_socket, &pkt, sizeof(network_packet), 0);
sleep(2);
while (1) {
if(turn == player_number){
printf("Enter X, Y:\n");
fflush(stdout);
fgets(buffer, BUFFER_SIZE, stdin);
sscanf(buffer, "%hhd %hhd", &pkt.x, &pkt.y);
pkt.opcode = 1;
pthread_mutex_lock(&socket_mutex);
++turn;
pkt.turn = turn;
if ((ret = send(client_socket, &pkt, sizeof(network_packet), 0) != sizeof(network_packet))) {
perror("send error");
} else {
// printf("Sent: %hhd %hhd %hhd %hhd %hhd\n", pkt.player_num, pkt.opcode, pkt.x, pkt.y, pkt.ship_type);
}
if(totalNumOfPlayers != 2){
if(turn == totalNumOfPlayers-1){
turn = 0;
}
}
pthread_mutex_unlock(&socket_mutex);
} else {
while (turn != player_number);
}
}
}
github-classroom[bot], 2 weeks ago • Initial commit
// network recv thread
void *rx(void *arg) {
pthread_detach(pthread_self());
int ret;
char buffer[BUFFER_SIZE];
network_packet pkt;
network_packet sendBackPkt;
// loop forever
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
github-classroom[bot], 2 weeks ago
Ln 85, Col 2
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
1 Spell
Linux
tabnine starter
See Tabnine Insights
Prettier

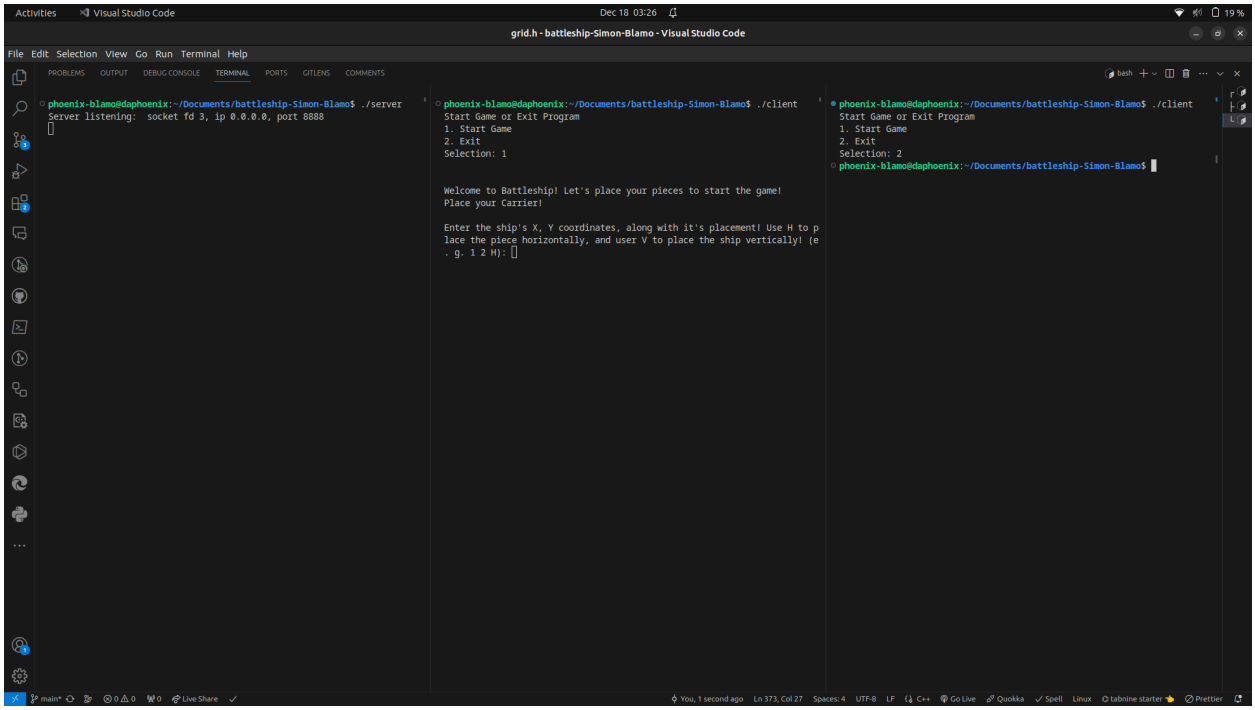

Activities
Visual Studio Code
Dec 18 03:48
client.cpp - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
client.cpp
client.cpp > tx(void *)
// loop forever
while (1) {
// block on read
if ((ret = recv(client_socket, buffer, BUFFER_SIZE, 0)) == 0) {
perror("recv failed");
exit(EXIT_FAILURE);
} else {
// check for valid packet length
if (ret != sizeof(network_packet)) {
printf("WARNING: packet size %d from client_socket ignored\n", ret);
} else {
// copy the receive buffer into the pkt struct
pthread_mutex_lock(&socket_mutex);
memcpy(&pkt, buffer, sizeof(network_packet));
// is this the player_num response sent after connect?
if (pkt.opcode == OPCODE_CONNECT) {
if(!connected) {
player_number = pkt.player_num;
printf("You're player %d!\n", player_number);
connected = true;
} else {
printf("Player %d connected!\n", pkt.player_num);
}
if(totalNumOfPlayers < pkt.numOfPlayers){
totalNumOfPlayers = pkt.numOfPlayers;
}
// printf("Total players = %d\n", totalNumOfPlayers);
} else {
if(pkt.opcode == OPCODE_SHOT){
sendBackPkt = theOceanGrid->handleShot(&pkt, &shipsRemaining);
printf("OCEAN GRID\n");
theOceanGrid->printGrid();
if(shipsRemaining == 0){
printf("ALL PLAYER SHIPS HAVE SUNK!\n");
printf("\033[31mYOU LOSE!\n");
sleep(1);
printf("YOU LOSE!\n");
sleep(1);
printf("YOU LOSE!\n");
sleep(3);
printf("YOU LOSE!\n");
printf("\033[0m");
exit(0);
}
turn = pkt.turn;
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
0 0 1
0
Live Share
github-classroom[bot], 2 weeks ago
Ln 85, Col 2
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
1 Spell
Linux
tabnine starter
See Tabnine Insights
Prettier

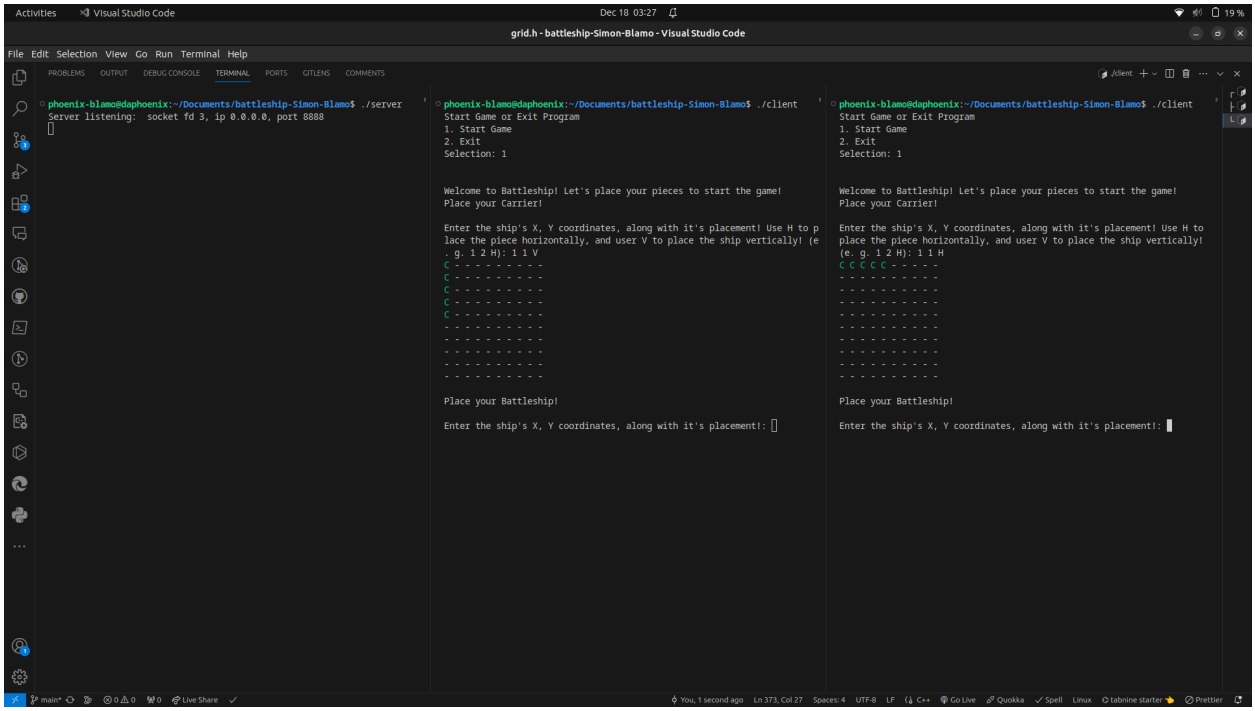

Activities
Visual Studio Code
Dec 18 03:48
Screenshot captured
You can paste the image from the clipboard.
File Edit Selection View Go Run Terminal Help
grid.h M
client.cpp
client.cpp > tx(void *)
turn = pkt.turn;
sendBackPkt.turn = pkt.turn;
if(totalNumOfPlayers == 2){
if (player_number == 1) {
turn = 1;
sendBackPkt.turn = 1;
} else {
turn = 0;
sendBackPkt.turn = 0;
}
}
send(client_socket, &sendBackPkt, sizeof(network_packet), 0);
if(turn != player_number) {
sleep(6);
printf("Waiting for turn...\n");
}
} else {
if(pkt.turn - 1 == player_number || ((pkt.turn == 0 && player_number == 1) && totalNumOfPlayers)) {
theTargetGrid->handleResponse(&pkt, &shipsSunk);
printf("TARGET GRID\n");
theTargetGrid->printGrid();
if(shipsSunk == 5){
printf("ALL OPPONENTS SHIPS HAVE SUNK!\n");
printf("\033[32mYOU WIN!\n");
printf("\033[0m");
exit(0);
}
printf("Waiting for turn...\n");
fflush(stdout);
}
}
// display received packet fields
// printf("\nReceived: %hhd %hhd %hhd %hhd %hhd\n", pkt.player_num, pkt.opcode, pkt.x, pkt.y, pkt.ship_type);
// repeat prompt
}
pthread_mutex_unlock(&socket_mutex);
}
}
}
return 0;
}
int main(int argc, char *argv[]) {
printf("Start Game or Exit Program\n");
printf("1. Start Game\n");
printf("2. Exit\n");
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
0 0 1
0
Live Share
github-classroom[bot], 2 weeks ago
Ln 85, Col 2
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
1 Spell
Linux
tabnine starter
See Tabnine Insights
Prettier

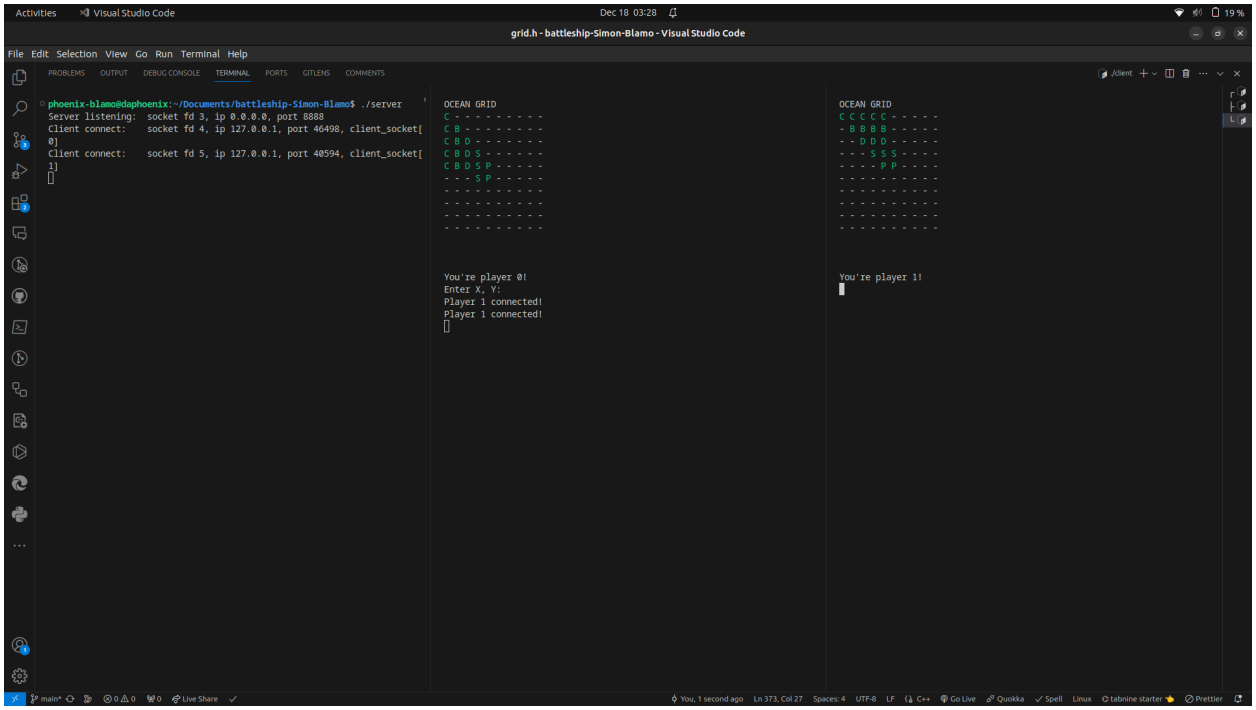

Activities
Visual Studio Code
Dec 18 03:48
client.cpp - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
grid.h M
client.cpp
client.cpp > tx(void *)
printf("Start Game or Exit Program\n");
printf("1. Start Game\n");
printf("2. Exit\n");
printf("Selection: ");
int opt = 0;
while (opt != 1 || opt != 2)
{
scanf("%d", &opt);
if(opt == 1 || opt == 2){
break;
}
printf("\n1. Start Game\n");
printf("2. Exit\n");
printf("Selection: ");
}
if(opt == 2){
exit(0);
}
struct sockaddr_in serv_addr;
char const *server_address;
theOceanGrid = new oceanGrid();
theTargetGrid = new targetGrid();
theOceanGrid->configGrid();
pthread_mutex_init(&socket_mutex, NULL);
// use server addr if specified on command line, otherwise use localhost
if (argc > 1) {
server_address = argv[1];
}
else {
server_address = LOCALHOST;
}
// create client socket
if ((client_socket = socket(AF_INET, SOCK_STREAM, 0)) < 0) {
perror("socket failed");
exit(EXIT_FAILURE);
}
// initialize sockaddr_in fields
serv_addr.sin_family = AF_INET;
serv_addr.sin_port = htons(SERVER_PORT);
if (inet_pton(AF_INET, server_address, &serv_addr.sin_addr) <= 0) {
perror("inet_pton failed");
exit(EXIT_FAILURE);
}
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
Live Share
github-classroom[bot], 2 weeks ago
Ln 85, Col 2
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
1 Spell
Linux
tabnine starter
Prettier

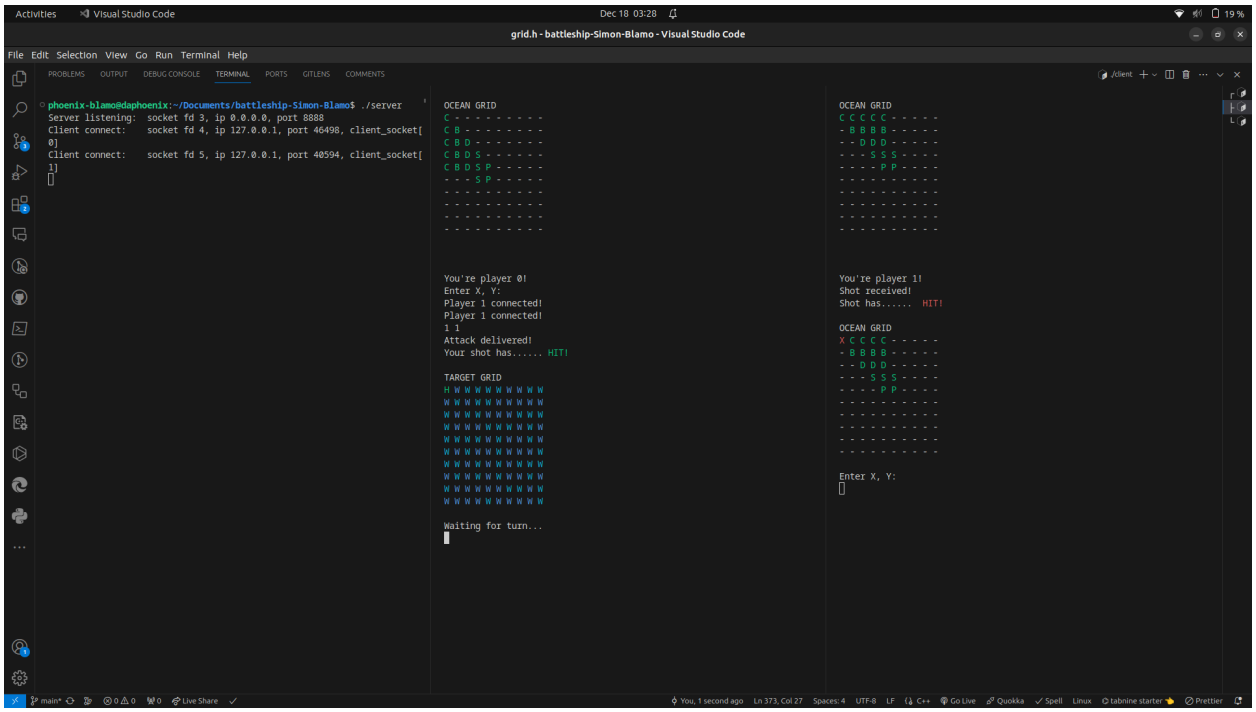

Activities
Visual Studio Code
Dec 18 03:48
Screenshot captured
You can paste the image from the clipboard.
File Edit Selection View Go Run Terminal Help
grid.h M
client.cpp
client.cpp > tx(void *)
// connect to server
if (connect(client_socket, (struct sockaddr *)&serv_addr, sizeof(serv_addr)) < 0) {
perror("connect failed");
exit(EXIT_FAILURE);
}
// Create two threads
pthread_t thread0, thread1;
pthread_create(&thread0, NULL, tx, NULL);
pthread_create(&thread1, NULL, rx, NULL);
// Wait for threads to finish
pthread_join(thread0, NULL);
pthread_join(thread1, NULL);
return 0;
}
Help us improve our support for C++
Take Short Survey
Remind Me Later
main*
Live Share
github-classroom[bot], 2 weeks ago
Ln 85, Col 2
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
1 Spell
Linux
tabnine starter
Prettier

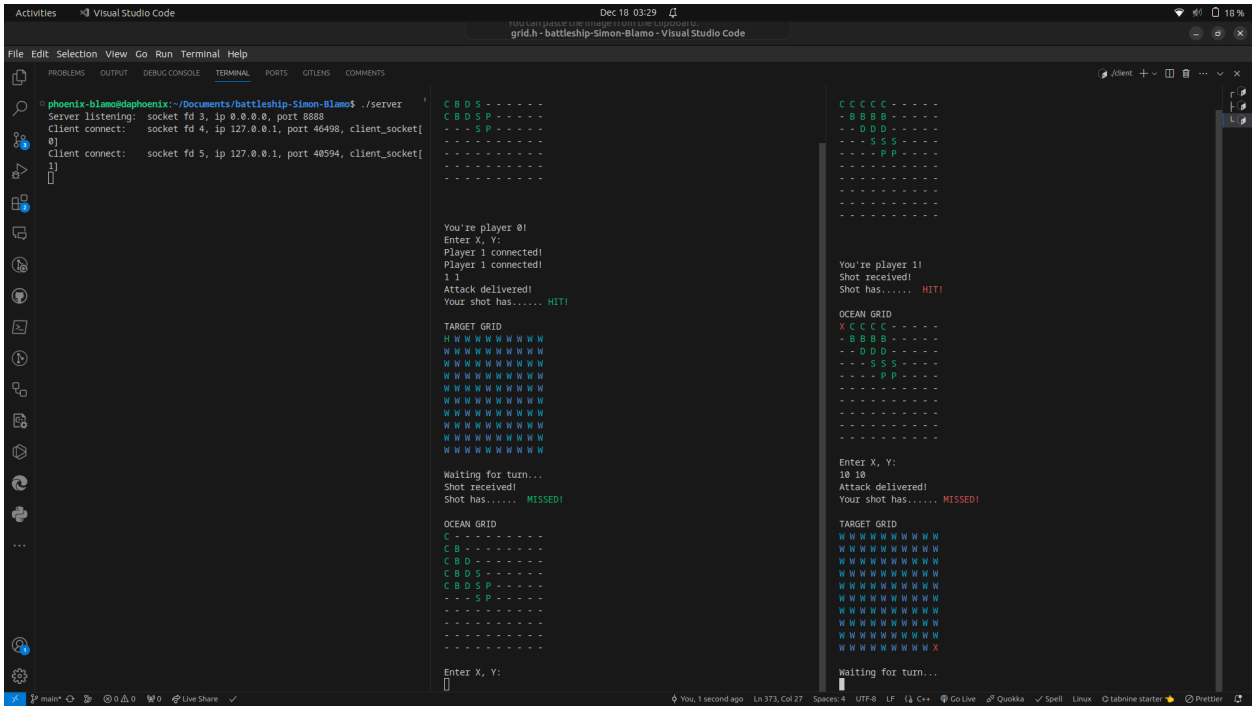
8. Output


Activities
Visual Studio Code
Dec 18 03:26
19 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./server
Server listening: socket fd 3, ip 0.0.0.0, port 8888
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./client
Start Game or Exit Program
1. Start Game
2. Exit
Selection: 1
Welcome to Battleship! Let's place your pieces to start the game!
Place your Carrier!
Enter the ship's X, Y coordinates, along with it's placement! Use H to place the piece horizontally, and user V to place the ship vertically! (e. g. 1 2 H):
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./client
Start Game or Exit Program
1. Start Game
2. Exit
Selection: 2
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$
You, 1 second ago Ln 373, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier


Activities
Visual Studio Code
Dec 18 03:27
19 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./server
Server listening: socket fd 3, ip 0.0.0.0, port 8888
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./client
Start Game or Exit Program
1. Start Game
2. Exit
Selection: 1
Welcome to Battleship! Let's place your pieces to start the game!
Place your Carrier!
Enter the ship's X, Y coordinates, along with it's placement! Use H to place the piece horizontally, and user V to place the ship vertically! (e. g. 1 2 H): 1 1 V
C - - - - - - - - - -
C - - - - - - - - - -
C - - - - - - - - - -
C - - - - - - - - - -
C - - - - - - - - - -
Place your Battleship!
Enter the ship's X, Y coordinates, along with it's placement!:
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./client
Start Game or Exit Program
1. Start Game
2. Exit
Selection: 1
Welcome to Battleship! Let's place your pieces to start the game!
Place your Carrier!
Enter the ship's X, Y coordinates, along with it's placement! Use H to place the piece horizontally, and user V to place the ship vertically! (e. g. 1 2 H): 1 1 H
C C C C C - - - - -
Place your Battleship!
Enter the ship's X, Y coordinates, along with it's placement!:
You, 1 second ago Ln 373, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier


Activities
Visual Studio Code
Dec 18 03:28
19 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
./client
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./server
Server listening: socket fd 3, ip 0.0.0.0, port 8888
Client connect: socket fd 4, ip 127.0.0.1, port 46498, client_socket[0]
Client connect: socket fd 5, ip 127.0.0.1, port 40594, client_socket[1]
OCEAN GRID
C - - - - - - - - - -
C B - - - - - - - - -
C B D - - - - - - - -
C B D S - - - - - - -
C B D S P - - - - - -
- - - S P - - - - - -
OCEAN GRID
C C C C C - - - - -
- B B B B - - - - -
- - D D D - - - - -
- - - S S S - - - -
- - - - P P - - - -
You're player 0!
Enter X, Y:
Player 1 connected!
Player 1 connected!
You're player 1!
main* 0 0 0 Live Share
You, 1 second ago Ln 373, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier


Activities
Visual Studio Code
Dec 18 03:28
19 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
./client
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./server
Server listening: socket fd 3, ip 0.0.0.0, port 8888
Client connect: socket fd 4, ip 127.0.0.1, port 46498, client_socket[0]
Client connect: socket fd 5, ip 127.0.0.1, port 40594, client_socket[1]
OCEAN GRID
C - - - - - - - - - -
C B - - - - - - - - -
C B D - - - - - - - -
C B D S - - - - - - -
C B D S P - - - - - -
- - - S P - - - - - -
You're player 0!
Enter X, Y:
Player 1 connected!
Player 1 connected!
1 1
Attack delivered!
Your shot has..... HIT!
TARGET GRID
H W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
Waiting for turn...
OCEAN GRID
C C C C C - - - - -
- B B B B - - - - -
- - D D D - - - - -
- - - S S S - - - -
- - - - P P - - - -
You're player 1!
Shot received!
Shot has..... HIT!
OCEAN GRID
X C C C C - - - - -
- B B B B - - - - -
- - D D D - - - - -
- - - S S S - - - -
- - - - P P - - - -
Enter X, Y:
main* 0 0 0 Live Share
You, 1 second ago Ln 373, Col 27 Spaces: 4 UTF-8 LF C++ Go Live Quokka Spell Linux tabnine starter Prettier


Activities
Visual Studio Code
Dec 18 03:29
18 %
grid.h - battleship-Simon-Blamo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
./client
phoenix-blamo@daphoenix:~/Documents/battleship-Simon-Blamo$ ./server
Server listening: socket fd 3, ip 0.0.0.0, port 8888
Client connect: socket fd 4, ip 127.0.0.1, port 46498, client_socket[0]
Client connect: socket fd 5, ip 127.0.0.1, port 40594, client_socket[1]
C B D S - - - - - -
C B D S P - - - - -
- - - S P - - - - -
You're player 0!
Enter X, Y:
Player 1 connected!
Player 1 connected!
1 1
Attack delivered!
Your shot has..... HIT!
TARGET GRID
H W W W W W W W W W
Waiting for turn...
Shot received!
Shot has..... MISSED!
OCEAN GRID
C - - - - - - - - -
C B - - - - - - - -
C B D - - - - - - -
C B D S - - - - - -
C B D S P - - - - -
- - - S P - - - - -
Enter X, Y:
C C C C C - - - - -
- B B B B - - - - -
- - D D D - - - - -
- - - S S S - - - -
- - - - P P - - - -
You're player 1!
Shot received!
Shot has..... HIT!
OCEAN GRID
X C C C C - - - - -
- B B B B - - - - -
- - D D D - - - - -
- - - S S S - - - -
- - - - P P - - - -
Enter X, Y:
10 10
Attack delivered!
Your shot has..... MISSED!
TARGET GRID
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W W
W W W W W W W W W X
Waiting for turn...
main*
0 0
0
Live Share
You, 1 second ago
Ln 373, Col 27
Spaces: 4
UTF-8
LF
C++
Go Live
Quokka
Spell
Linux
tabnine starter
Prettier