

Sprawozdanie SK2

Sieciowa gra logiczna: Connect Four

1. Opis projektu

Krótki opis

Stworzona przeze mnie sieciowa, turowa gra logiczna to **Connect Four** (Czwórki). W rozgrywce uczestniczy dwóch graczy – jeden gra symbolem **X**, a drugi **O**.

Celem każdego gracza jest:

- ułożenie czterech swoich symboli w linii (poziomo, pionowo lub ukośnie), zanim zrobi to przeciwnik,
- blokowanie przeciwnika w taki sposób, aby uniemożliwić mu ułożenie czterech symboli.

Gra toczy się na planszy o wymiarach **6x7**. Każdy gracz w swojej turze wybiera numer kolumny, do której zostanie dodany jego symbol. Symbole opadają na najniższe dostępne pole w kolumnie.

Gra kończy się zwycięstwem, gdy jeden z graczy ułoży cztery symbole w linii lub remisem, gdy plansza zostanie całkowicie zapelniona i żaden z graczy nie osiągnął celu.

Użyta technologia

Gra została zaimplementowana w języku C++ i przystosowana do działania w systemie Linux. W projekcie wykorzystano:

Sockets: Obsługa komunikacji sieciowej pomiędzy serwerem a klientami (protokół TCP),

Wielowątkowość: Mechanizm `std::thread` umożliwia prowadzenie wielu rozgrywek równocześnie,

Standardowe biblioteki C++: Obsługa wejścia/wyjścia, struktury danych oraz synchronizacja wątków.

2. Opis komunikacji pomiędzy serwerem i klientami

Komunikacja opiera się na protokole TCP. Serwer pełni rolę centralnego koordynatora, obsługując graczy oraz zarządzając stanem planszy.

Schemat działania:

1) Oczekiwanie na graczy

Serwer nasłuchuje połączeń na określonym porcie. Po nawiązaniu połączenia dwóch klientów tworzy nową rozgrywkę.

2) Przepływ wiadomości:

Rozpoczęcie gry:

Serwer przypisuje graczom symbole (X lub O) oraz wysyła im początkowy stan planszy.

Ruch gracza:

Gracz przesyła do serwera numer kolumny, do której chce wprowadzić symbol.

- Jeśli ruch jest **niepoprawny** (np. kolumna jest pełna), serwer odsyła odpowiedni komunikat i prosi o poprawienie ruchu.

- Jeśli ruch jest **poprawny**, serwer aktualizuje stan planszy i przesyła go do obu graczy.

Naprzemienność tur:

Gracze wykonują ruchy na zmianę, aż do zakończenia gry.

3) Koniec gry

- Po wykryciu zwycięstwa lub remisu serwer przesyła odpowiednią informację do obu graczy wraz z końcowym stanem planszy.

- Jeśli jeden z graczy się rozłączy, serwer informuje drugiego o przerwaniu gry.

3. Podsumowanie

Najważniejsze informacje o implementacji

Współbieżność: Serwer obsługuje wiele gier równocześnie dzięki wielowątkowości.

Rozgrywka sieciowa: Gra działa w trybie online z wykorzystaniem protokołu TCP.

Walidacja danych: Serwer sprawdza poprawność każdego ruchu, zapewniając zgodność z zasadami gry oraz odpowiedni format przesyłanych danych.

Obsługa błędów: Mechanizmy wykrywania niepoprawnych ruchów oraz obsługi sytuacji wyjątkowych, takich jak rozłączenie gracza.

Co sprawiło trudność?

Stworzenie współbieżnego serwera dla dwóch graczy

Opracowanie serwera umożliwiającego współbieżną obsługę dwóch graczy.

Problemy z komunikacją

Podczas implementacji komunikacji między klientami a serwerem pojawiały się trudności związane z błędnymi wiadomościami przesyłanymi między graczami. Wymagało to dokładnej analizy przesyłanych danych i wprowadzenia dodatkowej walidacji komunikatów.

Testowanie aplikacji sieciowej

Testowanie aplikacji było czasochłonne, ponieważ w trakcie testów często pojawiały się drobne błędy, które wymagały natychmiastowego rozwiązania.