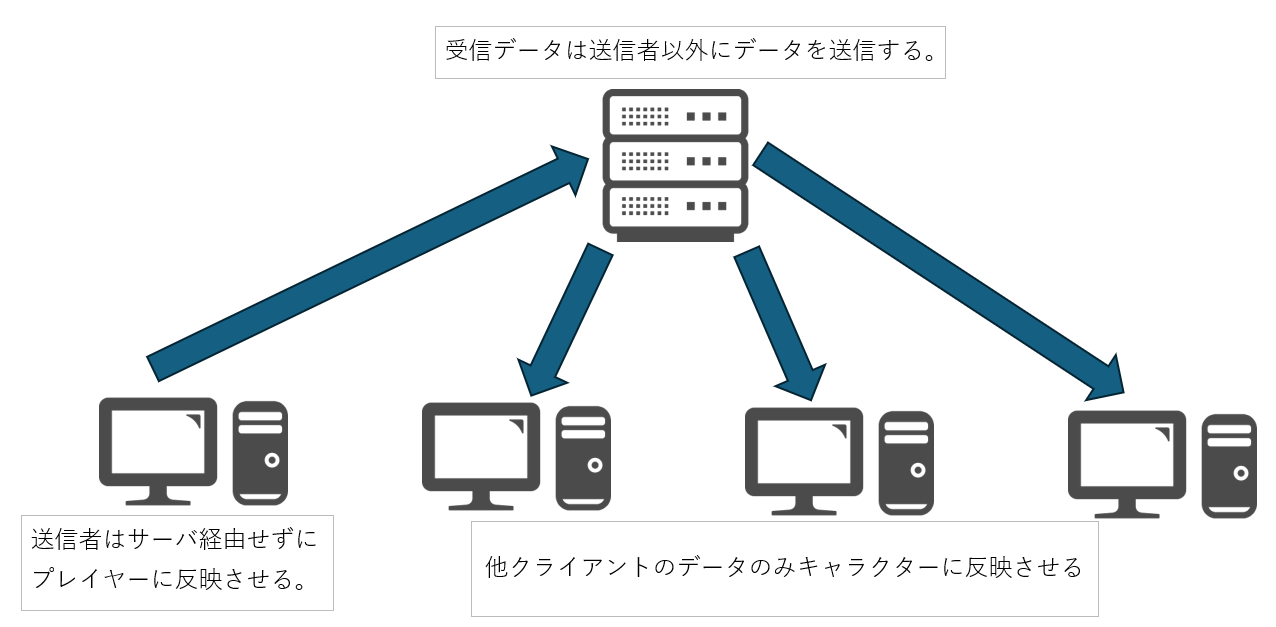
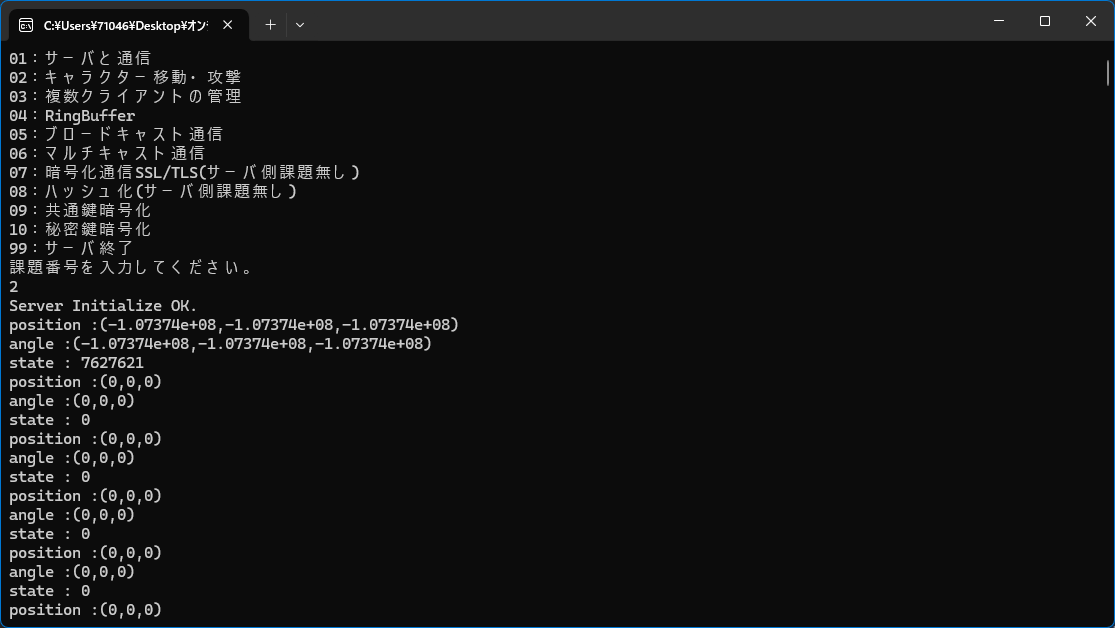
**---今回はUDPデータ通信でサーバーへプレイヤー情報を送り、返信を受け取ります---**

**(クライアント側では返信を受け取りますが、そのデータに対しての処理は行いません)**

* **内容**構造体データの送受信をUDP通信で行う。  
  （今回はサーバー、クライアント両側を作成します。）
* **目標**クライアント側：プレイヤー情報をサーバーへ送信、送り返されてくる情報を受け取る。  
  サーバー側　　：クライアントからプレイヤー情報を受け取ってログ画面へ表示、クライアントへ受信する。
* **プレイヤーの同期**プレイヤーの同期について（今回課題には関係ありません）、UDP通信を使用するリアルタイム性が求められるゲームでは毎フレーム、もしくは一定周期のタイマー関数を使用してデータをサーバへ送信します。  
  今回の課題ではサーバはそのままクライアントへデータを送り返していますが実際には送信者以外のプレイヤーにしか送りません。  
  全体同期のタイミングを除いて送信したプレイヤーは他プレイヤーのデータのみを反映し、自分自身は入力したデータをそのまま操作キャラクターへ反映させていきます。(自プレイヤーに関しては普通のキー入力からの動きとなります。)  
  全体同期や遅延対策は今後の授業で行っていきます。  
  
* **課題内容  
  ・サーバ側**クライアントから送信されてきたPlayerInformationを受信し、クライアントへ返す。  
  (TODO 02\_01～09まで)  
    
  **・クライアント側**サーバへPlayerInformationを送信し、サーバからデータを受け取る。  
  このとき、受け取ったデータはプレイヤー情報へ反映をさせない。  
  (TODO 02\_10～19)  
  前回の課題と同じコードの部分はコピーしても構いませんが、スレッドの関数名（クラス名）は02なので注意して下さい。

**プレイヤー操作**移動：AWSD or ←↑↓→  
攻撃：Ｘキー



課題としては、送信者にデータを送り返すが、送り返されたデータはプレイヤーに反映させない

プレイヤーのposition、angle、stateを  
PlayerInformation構造体としてまとめて送信する

**予備課題：**現在の仕様ではサーバー側doループ内にログ表示を作成するので、プレイヤー状況に変化が無くても、サーバー側ログにはデータが表示され続けます、そこで、変化があった時だけ、サーバー側ログ表示されるようにしてください。

（クライアントからの送信、サーバーからの送り返しはそのまま（常時送受信）でOKです。）