Player(const std::string name, const bool is_human)	Initialize a Player with the given name and whether or not it is human
void ChangePoints(const int x)	Update the points_ value according to the int passed in
void SetPosition(Position pos)	Update the Player's pos to the new position
std::string ToRelativePosition(Position other)	Translate the other position into a direction relative to the Player by comparing other with pos_

Maze()	Maze object constructor that is used to create a new instance of the maze game
void NewGame(Player *human, const int enemies)	Initializes a new game with 1 human player and a value of enemies to create
void TakeTurn(Player *p)	Prompts the user to choose what the Player will do on its turn
Player *GetNextPlayer()	Gets the Player object of the next player whose turn it is
bool IsGameOver()	Checks if game ending conditions are met
std::string GenerateReport()	Reports stats about the game and how it ended
friend std::ostream& operator<<(std::ostream &os, const Maze &m)	Writes the report of the game to a separate text file

Board()	Board object constructor that creates a new maze game board
int get_rows()	Returns the number of rows on the game board
int get_cols()	Returns the number of columns on the game board
SquareType get_square_value(Position pos) const.	Returns the type of square that the position is; either open, wall, player, treasure, or end
void SetSquareValue(Position pos, SquareType value)	Sets the type of square to the position parameter
std::vector<Position> GetMoves(Player *p)	Returns all the possible moves a player can make from its current position
bool MovePlayer(Player *p, Position pos)	Moves player to a different position on the board, returns false if the move cannot be made
SquareType GetExitOccupant()	Returns the square type of the exit square
friend std::ostream& operator<<(std::ostream& os, const Board &b)	Writes data about the current Board to a separate text file