





<< clase >> Archivo Tablero
- ifstream archivo;
+ ArchivoTablero (); + bool estaAbierto (); + void cargarTablero (Grafo tablero); + ~ArchivoTablero (); - Casillero* crearCasillero (char tipo);

<< clase >> Archivo Tablero
- ifstream archivo;
+ ArchivoTablero (); + bool estaAbierto (); + void cargarTablero (): Grafo tablero; ~ ArchivoTablero (); - Casillero* crearCasillero () : char tipo

<< clase >> Archivo Partida
- ifstream archivoEntrada; - ofstream archivoSalida;
+ ArchivoPartida() + bool hayPartidaGuardada (); + void cargarPartida (); Juego & juego; + void eliminarArchivo (); + ~ArhcivoPartida (); - void cargarPersonajes (Grafo& tablero, Jugador &jugador) - Personaje* crearPersonaje () string, string, string, string, array<int,2> posicion - void guardarPersonajes() array<Personaje* MAX_PERSONAJES> personajes, int cantPersonajes

<< clase >> Agua
+ Agua () string : nombre + string obtenerElemento () : override; + void mostrar () : override; + void alimentar () : override; + void atacar (array<Personaje*, MAX_PERSONAJES> enemigos) + void defender (array<Personaje*, MAX_PERSONAJES> aliados, Grafo&tablero) +int calcularAtkEntrante(Personaje* enemigo); - void aumentarEnergia (); - void curarAliado (Personaje* aliado); - void mostrarMsDefensa () : int nuevoVidaPropia;

<< clase >> Fuego
+ Fuego() string : nombre + string obtenerElemento () : override; + void mostrar () : override; + void alimentar () : override; + void atacar (array<Personaje*, MAX_PERSONAJES> enemigos) + void defender (array<Personaje*, MAX_PERSONAJES> aliados, Grafo&tablero) +int calcularAtkEntrante(Personaje* enemigo); - void aumentarEnergia (); - void aumentarVida (); - bool vidaMaxima ();

<< clase >> Aire
+ Aire () string : nombre + string obtenerElemento () : override; + void mostrar () : override; + void alimentar () : override; + void atacar (array<Personaje*, MAX_PERSONAJES> enemigos) + void defender (array<Personaje*, MAX_PERSONAJES> aliados, Grafo&tablero) +int calcularAtkEntrante(Personaje* enemigo); - void aumentarEnergia ();

<< clase >> Tierra
+ Tierra() string : nombre + string obtenerElemento () : override; + void mostrar () : override; + void alimentar () : override; + void atacar (array<Personaje*, MAX_PERSONAJES> enemigos) + void defender (array<Personaje*, MAX_PERSONAJES> aliados, Grafo&tablero) + int calcularAtkEntrante(Personaje* enemigo); - void aumentarEnergia (); - bool energiaMaxima ();

<< clase >> Archivo Personaje
- ifstream archivo;
+ ArchivoPersonajes (); + bool estaAbierto (); + void procesarArchivo (DiccionarioABB<string, Personaje*> &diccionario) + ~ArchivoPersonaje (); - Personaje* crearPersonaje () string, string, string, string);

<< clase >> Menu principal
- opcion: int
+ void iniciar() + void pausar(); + void interfazPrincipal(Juego& juego, string, Personaje*>&diccionario) - void limpiarPantalla(); - void mostrarOpciones(); - void pedirOpcion (); - void AgregarPersonaje (DiccionarioABB<string, Personaje*> diccionario); - void eliminarPersonaje (DiccionarioABB<string, Personaje*> diccionario); - void mostrarPersonaje (DiccionarioABB<string, Personaje*> diccionario); - void detallePersonaje (DiccionarioABB<string, Personaje*> diccionario); - void pedirElemento (); string elemento; - void pedirNomre (); string nombre;; - void validarElemento (); string elemento; - void pasarAMinuscula () string : cadena; - Personaje* crearPersonaje(string, string); - void cargarPersonajes (jugador1. jugador2, DiccionarioABB<string, Personaje*>diccionario); - void menuJuego (jugador1. jugador2, DiccionarioABB<string, Personaje*>diccionario); - void mostrarJuego (); - void seleccionarPersonaje (jugador1, jugador2, DiccionarioABB<string, Personaje*>diccionario);



