Alejandro Molines

Santiago Jorques

Magicards

pROGRAMACION 1

Descripción

El programa consiste en un juego de cartas contra la máquina. Tanto el jugador como la máquina recibe tres cartas con un número del 1 al 9 y debes eliminar las cartas del contrario o acabar con la suma de cartas más alta en caso de no poder seguir jugando.

Existen naturalezas y magias para cambiar el curso de la partida. Por ejemplo, aumentar el daño de tus cartas o recuperar una eliminada anteriormente. Sólo puedes tener una magia en cada partida y sólo tiene un uso, por lo que debes elegir sabiamente el momento de usarla para poder ganar.

También hay una estadística que indica la suma de daño implementada a tus cartas. Ésta varía al usar poderes que afecten al daño.

Implementación

Librerías utilizadas:

* <iostream>
* <cstdlib>
* <stdlib.h>
* <time.h>
* <vector>

Módulos utilizados en el programa:

|  |  |
| --- | --- |
| *Nombre* | *error* |
| *Tarea* | Muestra los errores del programa |
| *Parámetros de entrada* |  |
| *Parámetros de salida* | Devuelve el error cometido |

|  |  |
| --- | --- |
| *Nombre* | correccionnumeros |
| *Tarea* | Función auxiliar para que no hayan dos cartas iguales |
| *Parámetros de entrada* | Dos cartas iguales |
| *Parámetros de salida* | Las cartas arregladas |

|  |  |
| --- | --- |
| *Nombre* | correccionzeros |
| *Tarea* | Función auxiliar para que |
| *Parámetros de entrada* |  |
| *Parámetros de salida* | Las cartas arregladas |

|  |  |
| --- | --- |
| *Nombre* | ganarperder |
| *Tarea* | Menú para dar el resultado de la partida |
| *Parámetros de entrada* |  |
| *Parámetros de salida* | Devuelve el mensaje de victoria o derrota |

|  |  |
| --- | --- |
| *Nombre* | mapa |
| *Tarea* | Mapa del juego donde se observan las cartas y el estado de la partida |
| *Parámetros de entrada* |  |
| *Parámetros de salida* | Devuelve el mapa con las cartas |

|  |  |
| --- | --- |
| *Nombre* | showTurnUsu |
| *Tarea* | Menú para mostrar las opciones de juego |
| *Parámetros de entrada* | Opción 1 o 2 |
| *Parámetros de salida* | La acción que has elegido |

|  |  |
| --- | --- |
| *Nombre* | maxusu |
| *Tarea* | Función que muestra la carta con más vida del jugador |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | maxmaq |
| *Tarea* | Función que muestra la carta con más vida de la máquina |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | minmaq |
| *Tarea* | Función que muestra la carta con menos vida de la máquina |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | switchmagia |
| *Tarea* | Función que devuelve el tipo de magia que nos ha tocado |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | usomagico |
| *Tarea* | Módulo para poder usar las distintas magias que podamos tener |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | switchnature |
| *Tarea* | Función que nos devuelve la naturaleza que obtendrá la máquina |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | usonaturaleza |
| *Tarea* | Módulo para usar los distintos tipos de naturaleza |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | operaciones |
| *Tarea* | Función que calcula la diferencia de cartas cuando se ataca o se defiende |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | attack |
| *Tarea* | Función que pide la carta con la que atacar y el objetivo |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | ataqueia |
| *Tarea* | Función donde la máquina elegirá cómo atacar |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | repartircartas |
| *Tarea* | Módulo que da el valor a cada una de las cartas al empezar la partida |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | versusmachine |
| *Tarea* | Módulo principal del modo de jugar contra la máquina |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | showRulesMenu |
| *Tarea* | Menú que muestra todas las pestañas de reglas |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | maparegla |
| *Tarea* | Función que muestra el mapa de ejemplo de partida |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | rules |
| *Tarea* | Muestra las reglas dependiendo de la pestaña que hayas seleccionado previamente |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

|  |  |
| --- | --- |
| *Nombre* | showMainMenu |
| *Tarea* | Muestra el menú inicial |
| *Parámetros de entrada* |  |
| *Parámetros de salida* |  |

Conclusiones

* *Indicar si se han cumplido los objetivos iniciales del proyecto*

Al ser un proyecto bastante amplio, es normal que apareciesen errores puntuales que requerían tiempo para localizarlos. Sólo había que probar todas las posibilidades que ofrecía el juego y asegurar la perfección.

Ha sido un proyecto interesante en el que hemos podido explorar muchas posibilidades con el código, alcanzar metas que no imaginábamos al principio de curso y no sólo eso, sino que nos salió bien. Estamos orgullosos de nuestro trabajo.

Referencias bibliográficas