



Universidad Carlos III de Madrid
Escuela Politécnica Superior

GENERACIÓN AUTOMÁTICA DE NARRATIVAS: *Aventura Fantástica*

TRABAJO FIN DE GRADO

Autor

D. Francisco Javier Blanco Álvarez

Titulación

Grado en Ingeniería Informática

Tutor

Dr. D. Daniel Borrajo Millán

Septiembre de 2014



Agradecimientos

En primer lugar, quisiera agradecer a Isabel haberme soportado durante el desarrollo de este trabajo, con mis momentos de éxito, fracaso y mis ideas brillantes y absurdas.

En segundo lugar a mi padre por apoyarme y darme sus valiosos consejos y animarme en acabar este proyecto.

A mi madre y mi hermana agradecerles que hayan estado conmigo en estos años de carrera, al igual que a mis amigos Manuel, Sergio y Jorge.

A las maravillosas 587 personas que han participado en la encuesta para evaluar las narraciones obtenidas en este trabajo, darles las gracias por participar y darme sus opiniones.

Por último, a mis compañeros de clase y en especial a mi profesor Daniel, con el que he aprendido muchísimo y ha sabido aconsejarme y guiarme para ir por buen camino en este trabajo.



Resumen

En este trabajo se desarrolla un dominio que, al ejecutarlo en un planificador, genere historias de interés narrativo. Se emplea la *aventura fantástica* como temática principal de la narración, basada en la disputa por objetos poderosos entre personajes con buenos y malos motivos. Para el desarrollo de dicho dominio, se emplea el lenguaje *PDDL* para su creación y el planificador *Sayphi* para su ejecución. Estudiaremos sus objetivos, planificación y diseño, así como el análisis de los resultados obtenidos en los problemas creados para él.



Abstract

This bachelor thesis develops a domain which, when executed in a planner, generates stories of narrative interest. The *fantastic adventure* is used as the main theme of the story, based on powerful items dispute between characters with good and evil motives. For its development, *PDDL* language is used for the creation of that domain, and *Sayphi* planner for its execution. We are going to study its objectives, planning and design, as well as the analysis of the results obtained from the problems created for it.



Índice de contenido

1. Introducción	10
2. Estado del arte.....	11
2.1 Planificación automática de tareas	11
2.2 PDDL	11
2.3 Sayphi	14
2.4 Trabajos similares.....	15
3. Objetivo y motivación	17
4. Requisitos.....	18
4.1 Requisitos Funcionales	18
4.2 Requisitos No Funcionales	19
5. Diseño del sistema	20
5.1 Planteamiento	20
5.1.1 Movimiento en grupos	20
5.1.2 Movimiento conjunto.....	22
5.1.3 Omnisciencia	22
5.1.4 Movimiento final	23
5.1.4 Motivaciones	24
5.2 Definición de tipos de objetos	25
5.3 Definición de predicados.....	26
5.3.1 Lugares	26
5.3.2 Estado de personajes	26
5.3.3 Amistad	27
5.3.4 Objetos	28
5.3.5 Motivos.....	29
5.4 Definición de acciones.....	30
5.4.1 Interactuar con objetos.....	30
5.4.2 Herir y matar	33
5.4.3 Animales	36
5.4.4 Magia.....	39
5.4.5 Motivos.....	43
5.4.6 Aprendizaje y movimiento	50
5.4.7 Acciones que completan motivos	57
5.5 Creación de oraciones.....	60



5.5.1 Interactuar con objetos	60
5.5.2 Herir y matar	60
5.5.3 Animales	60
5.5.4 Magia	60
5.5.5 Motivos	61
5.5.6 Aprendizaje y movimiento	61
5.5.7 Acciones que completan motivos	61
6. Experimentación	62
6.1 Plan de pruebas	62
6.2 Creación de problemas	64
6.3 Problema P1	72
6.3.1 Objetivo del problema	72
6.3.2 Resultados obtenidos	72
6.3.3 Resultados de la encuesta	74
6.4 Problema P2	75
6.4.1 Objetivo del problema	75
6.4.2 Resultados obtenidos	75
6.4.3 Resultados de la encuesta	78
6.5 Problema P3	79
6.5.1 Objetivo del problema	79
6.5.2 Resultados obtenidos	79
6.5.3 Resultados de la encuesta	82
6.6 Problema P4	83
6.6.1 Objetivo del problema	83
6.6.2 Resultados obtenidos	83
6.6.3 Resultados de la encuesta	86
6.7 Problema P5	87
6.7.1 Objetivo del problema	87
6.7.2 Resultados obtenidos	87
6.7.3 Resultados de la encuesta	91
6.8 Comentarios generales de los resultados	92
7. Gestión del proyecto	93
7.1 Planificación inicial	93
7.2 Planificación final	94
7.3 Presupuesto	95



8. Conclusiones.....	97
9. Referencias.....	98
Anexo I: Manual de usuario	99
Anexo II: Código	101
Dominio.....	101
Problema P1.PDDL	119
Problema P2.PDDL	124
Problema P3.PDDL	129
Problema P4.PDDL	134
Problema P5.PDDL	139
Anexo III: Resultados	144
Problema P1.PDDL	144
Problema P1.PDDL (0.1)	144
Problema P1.PDDL (0.3)	144
Problema P1.PDDL (0.5)	145
Problema P1.PDDL (0.7)	145
Problema P1.PDDL (0.9)	145
Problema P2.PDDL	146
Problema P2.PDDL (0.1)	146
Problema P2.PDDL (0.3)	147
Problema P2.PDDL (0.5)	148
Problema P2.PDDL (0.7)	149
Problema P2.PDDL (0.9)	150
Problema P3.PDDL	151
Problema P3.PDDL (0.1)	151
Problema P3.PDDL (0.3)	152
Problema P3.PDDL (0.5)	153
Problema P3.PDDL (0.7)	154
Problema P3.PDDL (0.9)	155
Problema P4.PDDL	156
Problema P4.PDDL (0.1)	156
Problema P4.PDDL (0.3)	157
Problema P4.PDDL (0.5)	158
Problema P4.PDDL (0.7)	159
Problema P4.PDDL (0.9)	161



Problema P5.PDDL	163
Problema P5.PDDL (0.1)	163
Problema P5.PDDL (0.3)	164
Problema P5.PDDL (0.5)	166
Problema P5.PDDL (0.7)	168
Problema P5.PDDL (0.9)	169
Anexo IV: Encuesta realizada al público	172
Página 1: Introducción	172
Página 2: Problema 1	174
Página 3: Problema 2	174
Página 4: Problema 3	177
Página 5: Problema 4	179
Página 6: Problema 5	182
Resultados de la encuesta.....	185

Índice de ilustraciones

Ilustración 1 - Esquema de tipos de objetos	25
Ilustración 2 - Participación por edad y sexo	63
Ilustración 3 - Esquema de motivos y objetos	64
Ilustración 4 - Encuesta P1 (0.3).....	74
Ilustración 5 - Encuesta P1 (0.9).....	74
Ilustración 6 - Encuesta P2 (0.1).....	78
Ilustración 7 - Encuesta P2 (0.7).....	78
Ilustración 8 - Encuesta P3 (0.5).....	82
Ilustración 9 - Encuesta P3 (0.9).....	82
Ilustración 10 - Encuesta P4 (0.1).....	86
Ilustración 11 - Encuesta P4 (0.5).....	86
Ilustración 12 - Encuesta P5 (0.1).....	91
Ilustración 13 - Encuesta P5 (0.9).....	91
Ilustración 14 - Diagrama de Gantt Inicial.....	93
Ilustración 15 - Diagrama de Gantt Final.....	94
Ilustración 16 - Rutas .sbclrc	99
Ilustración 17 - Rutas .init.....	99
Ilustración 18 - Encuesta pág. 1 parte 1.....	172
Ilustración 19 - Encuesta pág. 1 parte 2.....	173
Ilustración 20 - Encuesta pág. 2	174
Ilustración 21 - Encuesta pág. 3 parte 1.....	175
Ilustración 22 - Encuesta pág. 3 parte 2.....	176
Ilustración 23 - Encuesta pág. 4 parte 1.....	177
Ilustración 24 - Encuesta pág. 4 parte 2.....	178
Ilustración 25 - Encuesta pág. 5 parte 1.....	179
Ilustración 26 - Encuesta pág. 5 parte 2.....	180
Ilustración 27 - Encuesta pág. 5 parte 3.....	181
Ilustración 28 - Encuesta pág. 6 parte 1.....	182
Ilustración 29 - Encuesta pág. 6 parte 2.....	183
Ilustración 30 - Encuesta pág. 6 parte 3.....	184

Índice de tablas

Tabla 1 - Requisitos funcionales.....	18
Tabla 2 - Requisitos no funcionales.....	19
Tabla 3 - Resultados P1 (0.1).....	73
Tabla 4 - Resultados P1 (0.3).....	73
Tabla 5 - Resultados P1 (0.5).....	73
Tabla 6 - Resultados P1 (0.7).....	73
Tabla 7 - Resultados P1 (0.9).....	73
Tabla 8 - Resultados P2 (0.1).....	77
Tabla 9 - Resultados P2 (0.3).....	77
Tabla 10 - Resultados P2 (0.5).....	77
Tabla 11 - Resultados P2 (0.7).....	77
Tabla 12 - Resultados P2 (0.9).....	77
Tabla 13 - Resultados P3 (0.1).....	81
Tabla 14 - Resultados P3 (0.3).....	81
Tabla 15 - Resultados P3 (0.5).....	81
Tabla 16 - Resultados P3 (0.7).....	81
Tabla 17 - Resultados P3 (0.9).....	81
Tabla 18 - Resultados P4 (0.1).....	85
Tabla 19 - Resultados P4 (0.3).....	85
Tabla 20 - Resultados P4 (0.5).....	85
Tabla 21- Resultados P4 (0.7).....	85
Tabla 22 - Resultados P4 (0.9).....	85
Tabla 23 - Resultados P5 (0.1).....	90
Tabla 24 - Resultados P5 (0.3).....	90
Tabla 25 - Resultados P5 (0.5).....	90
Tabla 26 - Resultados P5 (0.7).....	90
Tabla 27 - Resultados P5 (0.9).....	90
Tabla 28 - Coste de personal.....	95
Tabla 29 - Amortizaciones.....	96
Tabla 30 - Presupuesto final.....	96
Tabla 31 - Enumeración de narraciones.....	185
Tabla 32 - Resultados de la encuesta.....	197

1. Introducción

Con el uso de la informática para la generación automática de narrativas, en lugar de seguir con el común desarrollo a mano de una historia, se pueden mejorar muchos de sus aspectos. Por un lado, la capacidad de computación, que supera con creces la imaginación humana, es capaz de realizar en pocos segundos miles de permutaciones y combinaciones de eventos de una misma historia. Por otro lado, evita el error humano al estar diseñado para llegar a un objetivo sin fallos en el argumento. Para este fin, emplearemos la planificación automática.

Se entiende como planificación automática al campo que busca, dada la descripción de un dominio (a través de las acciones que se pueden realizar), un estado inicial y un conjunto de metas, generar un plan que permita obtener dichas metas a partir del estado inicial por la ejecución de las acciones del plan.

En este Trabajo de Fin de Grado, nuestro dominio describirá las acciones de la *aventura fantástica*, como por ejemplo es el caso del cuento *El Hobbit* o de la saga *El Señor de los Anillos*. Armas, viajes, objetos poderosos, magia... todo ello implementado en una serie de acciones que evaluarán si se cumplen las precondiciones para llevar a cabo su resultado. Estudiaremos en detalle las acciones que permiten matar, herir y destruir, así como las acciones de tratar objetos (dar, comerciar, coger) y sobre todo, las de aprendizaje, que determinarán las motivaciones de los personajes con las que querrán hacer una acción concreta u otra. Estos personajes serán capaces de entablar amistad entre ellos o bien enfrentarse por alguna razón. Es realmente interesante la capacidad que tendrán de cambiar sus relaciones sociales para conseguir sus objetivos, ya que según les convenga podrán elegir aliarse con alguien o bien querer matarlo. Claro está, todo con cierto límite. Ahí es donde entran las motivaciones de los personajes, que definirán la naturaleza de su objetivo. Si varios tienen la misma motivación, optarán por aliarse, mientras que si sus motivos están en desacuerdo, se enemistarán.

A lo largo de las siguientes páginas se presentará el desarrollo del dominio de planificación. En el siguiente apartado estudiaremos el estado del arte, explicando la planificación automática de tareas, así como el lenguaje *PDDL* con el que se programará el proyecto, la herramienta *Sayphi* en el que lo ejecutaremos y los proyectos similares. Seguiremos con los objetivos y motivaciones del trabajo, donde explicaremos las metas que se pretenden conseguir con su desarrollo. Para su realización, expondremos los requisitos necesarios, tanto funcionales como no funcionales. A continuación, entraremos en detalle en el diseño del sistema, donde explicaremos minuciosamente cada parte tanto del dominio como de los problemas. Una vez explicados, analizaremos los resultados de dichos problemas para medir su interés y aleatoriedad, veremos las diferencias entre ellos y estudiaremos el porqué de los resultados. Ya mostrado su desarrollo y experimentación, analizaremos la gestión del proyecto (planificación y presupuesto) y finalizaremos con las conclusiones, donde veremos las partes que hemos aprendido, el trabajo que se ha realizado y las mejoras que se podrían implementar en el futuro.

2. Estado del arte

2.1 Planificación automática de tareas

La planificación automática de tareas es una rama dentro de la Inteligencia Artificial que consiste en la resolución de un problema dado un estado inicial, una serie de acciones y un conjunto de metas. El plan se forma a partir de un conjunto de esas acciones elegidas de manera automática y que deben concluir con la resolución del problema planteado.

En ciertos problemas es importante tener en cuenta el consumo de recursos, si los hubiera, así como realizar el mínimo número de acciones o bien hacerlo en el menor tiempo posible. Este campo dota la capacidad de realizar ciertas tareas sin necesidad de un operador, ya que la propia inteligencia trazará un plan de acciones para llevar a cabo la tarea a realizar.

Para llevar a cabo dichas tareas, se necesitarán por un lado el dominio, formado por el conjunto de posibles estados y el conjunto de operadores o acciones que se pueden realizar, y el problema, que está definido por un estado inicial y un conjunto de metas. La tarea de la planificación es alcanzar dichas metas del problema realizando una serie de las posibles acciones que harán pasar el problema de un estado a otro.

2.2 PDDL

El *Planning Domain Definition Language* (PDDL) es un intento de estandarizar los lenguajes de planificación. Fue desarrollado por Drew McDermott y sus colegas en 1998 (inspirado en STRIPS y ADL entre otros) principalmente para el *International Planning Competition* (IPC) de 1998, y luego evolucionó en cada competición [1].

- **PDDL1.2:** Fue el lenguaje oficial del 1^{er} y 2^o IPC (1998 y 2000). Separó el modelo del problema de planificación en dos partes: *dominio* y *problema*.
 - El **dominio** define los requisitos, los tipos de objetos y sus jerarquías, constantes, predicados y las posibles acciones. Estas acciones, a su vez, incluyen los parámetros necesarios, las precondiciones que deben cumplirse para poder ser ejecutadas, y los efectos resultantes de ejecutar dichas acciones.
 - El **problema**, basándose en los tipos de objetos, predicados y constantes del dominio, establece las condiciones iniciales y el objetivo final que se desea conseguir.
- **PDDL2.1:** Fue el lenguaje oficial del 3^{er} IPC de 2002 [2]. Introdujeron las *funciones numéricas*, útiles para medir variables contables como el dinero, tiempo, distancia... También se incluyeron las *métricas*, que hacen posible evaluar la calidad de los planes y se añadieron *acciones durativas*.
- **PDDL2.2:** Fue el lenguaje oficial del 4^o IPC de 2004 [3]. Se introdujeron los *predicados derivados* para modelar la dependencia de los hechos dados con otros hechos.

- **PDDL3.0:** Fue el lenguaje oficial del 5º IPC de 2006 [4]. Se incluyeron *restricciones de transición de estados* y *preferencias* que adjudican un nivel de importancia a los objetivos.
- **PDDL3.1:** Fue el lenguaje oficial del 6º y 7º IPC de 2008 y 2011 respectivamente y es la última versión de PDDL hasta la fecha. Se incluyeron las funciones-objeto, de tal manera que ya las funciones no sólo tenían un valor entero o real, si no que podían ser cualquier tipo de objeto.

En un principio, se pensó en emplear PDDL2.1 para disponer de las funciones numéricas, ya que iban a ser de gran ayuda, pero tras varias pruebas, no era totalmente compatible con *Sayphi*. De tal manera, el **dominio** se ha implementado en PDDL1.2, que tiene la siguiente estructura:

- En primer lugar, se especifica el nombre del dominio:

```
(define (domain narrative)
```

- Después, las características de PDDL empleadas en el dominio precedidos de dos puntos “:”:

```
(:requirements :strips :typing :equality)
```

- Establecemos la jerarquía de objetos:

```
(:types  
  edible dangerous - animal  
  character animal - entity  
  place  
  goodmotive evilmotive neutralmotive - motive  
  arm usefobj - storable  
  entity storable - object  
  status  
)
```

- Una vez declarados los tipos de objetos, podemos ya indicar las constantes, que son instancias de objetos definidas en el propio dominio que tendrán en común todos los problemas:

```
(:constants  
  Destroyed Acquired - status  
  Neutral - neutralmotive  
)
```

- A continuación se definen los predicados que se emplearán. En primer lugar se nombra el predicado, seguido del objeto u objetos empleados en él, precedidos del signo de interrogación “?” y seguidos del tipo de objeto con un guion “-”:

```
(:predicates
  (at ?obj - object ?place - place)
  (can-be-destroyed-in ?obj - object ?place - place)
  [...]
)
```

La lectura de los predicados sería: “*?obj can-be-destroyed-in ?place*”.

- Ya declarados los objetos y los predicados podemos establecer las acciones, que a su vez se dividen en 3 partes, parámetros (los objetos que van a intervenir en la acción), precondiciones (el conjunto de predicados que deben cumplirse para poder ejecutarse) y efecto (los cambios que surgen al hacer esta acción):

```
(:action take-obj
  :parameters (?char - character ?obj - storable ?place -
  place)
  :precondition (and
    (not (dead ?char))
    (at ?char ?place)
    (at ?obj ?place)
  )
  :effect (and
    (not (at ?obj ?place))
    (have-an-obj ?char ?obj)
  )
)
```

Ya disponemos de todas las partes que requiere el dominio. A continuación nos quedaría definir un **problema** para dicho dominio, que debe hacerse con la siguiente estructura:

- En primer lugar, como hicimos en el dominio, definimos el problema:

```
(define (problem p1)
```

- Después, indicamos a cuál dominio pertenece:

```
(:domain narrative)
```

- Ya podemos declarar los objetos que vamos a emplear en nuestro problema. Los que sean de un mismo tipo los podemos declarar simultáneamente separándolos por un espacio:

```
(:objects
  Nazgul - character
  Orc1 Orc2 Orc3 - character
  Rabbit1 Rabbit2 - edible

  [...]

)
```

- A continuación, haciendo uso de estos objetos, y los predicados y variables globales del dominio, establecemos el estado inicial en que se encuentran dichos objetos:

```
(:init
  (at Frodo La_Comarca)
  (have-an-obj Frodo Ring)
  (have-motive Frodo Peace)
  (friends Bilbo Frodo)
  (weak Frodo)

  [...]

)
```

- Por último, especificamos el objetivo final que deseamos obtener:

```
(:goal (and
  (is Ring Destroyed)
  (dead Sauron)
)

)
```

2.3 Sayphi

Sayphi es un planificador automático construido con el objetivo de integrar varias técnicas de aprendizaje aplicadas a la planificación. Tiene un planificador de búsqueda hacia adelante como FF incluyendo muchos algoritmos de búsqueda [5]. También incluye cuatro subsistemas de aprendizaje, todos ellos competidores en la primera IPC con track de aprendizaje.

Sayphi permite ejecutar cualquier dominio, ya que es independiente de él, y puede generar planes, con sus técnicas de búsqueda, de manera aleatoria (escogiendo al azar entre todas las acciones hasta llegar al objetivo) o pseudo-aleatoria (escogiendo acciones determinadas por los mecanismos de planificación empleados, pudiendo en algunos casos introducir un grado de aleatoriedad en su elección de acciones). En nuestro caso, se ha hecho uso de esta última modalidad, en particular empleando el algoritmo *rrt*, que nos permite manipular su aleatoriedad mediante el parámetro *probability-ehc*, que tiene valores entre 0 (muy aleatorio) y 1 (nada aleatorio).

Para cargar, compilar y ejecutar Sayphi se empleará el programa *Steel Banck Common Lisp* (SBCL), un compilador de lenguaje Lisp gratuito y de código abierto.

2.4 Trabajos similares

Hay varios estudios dedicados a la generación de historias que tratan de mejorarlas mediante la planificación automática en los últimos años.

Un trabajo muy parecido al realizado en este proyecto es "*Narrative Planning: Compilations to Classical Planning*" de Patrik Haslum [6]. En él se busca generar historias que alcancen una meta teniendo en cuenta las intencionalidades de los personajes. Esta intencionalidad puede cambiar a lo largo del desarrollo de la historia en cada personaje con tal de llegar al fin deseado. El autor se basó en un trabajo previo "*Narrative planning: Balancing plot and character*" de M. Riedl y R. Young [7]. En dicho trabajo, se describe la generación de historias basándose en la intencionalidad, clasificando las acciones en *intencionales* y que *describen sucesos*.

Otro artículo interesante es "*Evaluation of automatic generation of basic stories*" de Federico Peinado y Pablo Gervás [8]. En él, se pretende generar una historia del tipo fabula con el sistema *KIIDS*, que genera historias paso a paso, añadiendo episodios narrativos y elementos de simulación cuando es necesario continuar con el proceso de la narración. Su núcleo es la *memoria imaginativa*, que crea nuevos episodios a partir de los anteriores.

También está el artículo "*Applying Planning to Interactive Storytelling: Narrative Control using State Constraints*" de J. Porteous, M. Cavazza y F. Charles [9]. Se propone utilizar PDDL3.0 para emplear las *restricciones de transición de estados* (restricciones fuertes) y las *preferencias* (restricciones débiles). De esta manera, los planes generados deberán cumplir todas las restricciones fuertes y satisfacer el mayor número de restricciones débiles.

Por último, el trabajo más similar a este es "*Generación Automática de Narrativas*" de P. Pérez [10]. En este proyecto también se hace uso del planificador Sayphi y del lenguaje PDDL1.2 para la generación de historias. Estas historias son textos de género negro (mafia, policías, gánsters...). Este trabajo coincide con el presente en diversos aspectos:

- En primer lugar, hace uso de *motivos* por los cuales los personajes tendrán intencionalidad para hacer unas acciones u otras.
- Dispone de acciones de *movimiento intencionado*. Es decir, un personaje decide ir a un lugar (según su intención) y realiza dicho viaje en una misma acción. Esto facilita mucho la capacidad de cómputo del planificador. También hace uso de esta técnica en otras acciones, como las de matar u obtener objetos.
- En algunos casos, emplea *objetos como indicadores de estado* en lugar de predicados. Por ejemplo, para medir el dinero de un personaje, crea los diferentes niveles de riqueza y los asigna a los personajes mediante predicados. En el presente trabajo también se hace uso de esta técnica para los estados finales de algunos objetos.

Por otro lado, a pesar de sus muchas similitudes, hay muchas diferencias entre ellos:

- En primer lugar, por supuesto, es la temática principal. Mientras que en ese trabajo se centra en el género negro, en éste se desarrollan historias de aventuras fantásticas.
- Sus acciones se centran en el dinero, la traición, asesinato y encarcelamiento. Sin embargo en el dominio de este proyecto se describen acciones que interactúen con objetos, manipular magia, forjar amistades, crear enemigos, y sus motivaciones principales se centran en adquirir o destruir ciertos objetos.
- Mientras que en ese trabajo se desarrolla un gran árbol de objetos, muy específicos, en éste se emplea más los predicados para definir las cualidades de dichos objetos, generando un menor número de los mismos.
- Aunque ambos dispongan de acciones directas, como explicamos anteriormente, en el presente trabajo se ha decidido respetar algunas oportunidades que pueden aparecer si esto no es así. Es decir, supongamos una acción que mueva a un personaje de lugar para coger un objeto. En ese proyecto, una vez ejecutada la acción el personaje se encontrará en el lugar del objeto y lo cogerá. Sin embargo, en el presente trabajo, se ha decidido no realizar esta última acción. Un personaje podrá moverse a un lugar porque le interesa coger un objeto que esté allí, pero no lo cogerá en esa acción, así se hace posible que ocurra algo que pueda evitarlo.
- De igual manera sucede con la intención para enfrentarse a un enemigo. En la historia de género negro, una persona decide matar a alguien. Sin embargo, en la aventura fantástica los personajes crean una relación de amistad o enemistad que les permitirá entrar en combate entre sí o participar en otras diversas acciones.
- En último lugar, en cuanto a la experimentación del trabajo, P. Pérez hace un análisis de la longitud y la diversidad (tanto entre los planes como entre las acciones dentro de un mismo plan). En el presente trabajo no se ha considerado la diversidad de acciones en el mismo plan, sin embargo sí se ha tenido en cuenta el interés que generan los planes mediante una encuesta al público.

3. Objetivo y motivación

El objetivo principal de este proyecto consiste en la generación automática de narraciones de interés, que sean producto de la ejecución en un planificador de un dominio y sus correspondientes problemas. El dominio establecerá los diferentes tipos de objetos necesarios, así como los predicados y las acciones que formarán el argumento de nuestras historias.

Como ya mencionamos anteriormente, nuestro dominio estará centrado en la *aventura fantástica*, que a la par que interesante, también resulta muy compleja. Hay una gran cantidad de personajes, así como objetos y lugares, lo que afectará directamente al tiempo empleado en la resolución de los problemas.

Estos problemas serán ejecutados varias veces, cambiando la aleatoriedad con la que se planifican sus acciones. Estudiaremos los cambios producidos entre ellos, y escogeremos algunos para realizar una encuesta a un público diverso, para analizar si dichos cambios se reflejan en el interés que generan las historias.

En definitiva, los principales objetivos a conseguir en este trabajo son los siguientes:

- Diseñar un dominio de planificación de narrativas de aventuras fantásticas.
- Que permita la inclusión de nuevos personajes, recursos, acciones y propiedades de forma declarativa.
- Que permita generar diferentes historias cada vez que se ejecute.
- Que las historias generadas tengan coherencia.
- Que las historias generadas tengan interés.
- Que las historias generadas se puedan integrar fácilmente con software que permita, entre otras cosas, generar narración a partir de la historia.

La motivación principal de este trabajo es escalarlo a un ámbito superior, es decir, que las historias generadas puedan servir de soporte para otros artes como el cine, televisión, libros e incluso videojuegos.

4. Requisitos

El sistema se desarrollará cumpliendo una serie de requisitos funcionales y no funcionales.

4.1 Requisitos Funcionales

Los requisitos funcionales describen la interacción entre el sistema y su ambiente independientemente de su implementación. El ambiente incluye al usuario y cualquier otro sistema externo que interactúa con el sistema.

CÓDIGO	NOMBRE	DESCRIPCIÓN	PRIORIDAD
RF-001	Lenguaje de programación	El dominio y los problemas estarán escritos en PDDL1.2	Alta
RF-002	Ejecución con dominio y problema	El planificador se podrá ejecutar utilizando un dominio y un problema escritos en PDDL.	Alta
RF-003	Ejecución aleatoria	El sistema se podrá ejecutar en modo aleatorio.	Alta
RF-004	Modificar aleatoriedad	El usuario podrá modificar la aleatoriedad al ejecutar el sistema.	Media
RF-005	Modificar iteraciones	El usuario podrá elegir el número de iteraciones que ejecutará el sistema.	Media
RF-006	Ejecución mediante heurísticas	El sistema se podrá ejecutar en modo planificación mediante heurísticas.	Alta
RF-007	Creación de problemas	El usuario podrá crear nuevos problemas.	Alta
RF-008	Elección de dominio	El usuario podrá elegir el dominio que desee ejecutar.	Alta
RF-009	Elección de problema	El usuario podrá elegir el problema que desee ejecutar.	Alta
RF-010	Solución en consola	El sistema podrá mostrar la solución por la consola de comandos.	Alta
RF-011	Tiempo máximo de ejecución	El sistema podrá tener un tiempo máximo establecido por el usuario para generar el plan.	Media
RF-012	Notificar solución	El sistema podrá notificar al usuario si ha sido posible encontrar solución al problema.	Media

CÓDIGO	NOMBRE	DESCRIPCIÓN	PRIORIDAD
RF-013	Notificar variaciones en los resultados	El sistema podrá notificar los cambios entre las iteraciones de la ejecución.	Baja
RF-014	Modificar las oraciones	El usuario podrá modificar las oraciones que componen las acciones del dominio.	Baja

Tabla 1 - Requisitos funcionales

4.2 Requisitos No Funcionales

Los requisitos no funcionales describen aspectos del sistema que son visibles por el usuario. No incluyen una relación directa con el comportamiento funcional del sistema. Incluyen restricciones como el tiempo de respuesta, precisión, recursos consumido, seguridad... etc.

CODIGO	NOMBRE	DESCRIPCIÓN	PRIORIDAD
RNF-001	Sistemas operativos	El sistema se podrá ejecutar en Windows, Linux y Mac OS X.	Alta
RNF-002	Tiempo límite	El sistema no tardará en generar un plan más del tiempo máximo indicado por el usuario.	Alta
RNF-003	SBCL	El usuario deberá acceder a SBCL para ejecutar el sistema.	Alta

Tabla 2 - Requisitos no funcionales

5. Diseño del sistema

Como se ha mencionado anteriormente, el objetivo de este sistema consiste en crear un dominio y problemas que generen historias de manera aleatoria, que ofrezcan interés y mantengan una cierta coherencia. Se ha escogido la *aventura fantástica* como base de nuestra narración. Deberán aparecer personajes buenos y malos que peleen entre sí, objetos mágicos que otorguen poder, armas, animales, motivaciones, amistades y enemistades.

Se ha dividido este apartado en sub-apartados más ligeros en los cuales se detallarán el planteamiento general del sistema y definiremos los tipos de objetos empleados, los predicados y las acciones, así como la creación de oraciones de dichas acciones.

5.1 Planteamiento

La primera cosa que debe estar clara es el planteamiento que se ha seguido en el desarrollo de este dominio. A lo largo de su diseño se han tenido buenas y malas ideas, hasta que finalmente se han juntado las mejores para darle forma. A continuación veremos los diferentes tipos de mecánicas que se han empleado o bien se han descartado para dar lugar al desarrollo de las más efectivas.

5.1.1 Movimiento en grupos

El primer gran problema que surge al intentar programar este tipo de narrativa es la capacidad de movimiento de los personajes. Sería realmente adecuado que los personajes se moviesen juntos, ese siempre fue el punto de partida original. Sin embargo, tras varios intentos de diversas maneras, se han descartado todas ellas ya que no han resultado viables.

En un principio se desarrollaron una serie de objetos, predicados y acciones que permitiesen el movimiento por **grupos** de personajes. Se definió una serie de objetos del tipo “*group*”, predicados para indicar que alguien pertenecía a esos grupos y acciones de entrar, salir y crear o disolver el grupo. Estos grupos cumplían su objetivo a la perfección, permitían el movimiento de varios personajes simultáneamente desde un lugar de origen a uno de destino.

Los problemas de los grupos comienzan cuando se aprecia su complejidad. En primer lugar, se debe llevar un contador de miembros, básico para poder interactuar con ellos. Dicho contador se podría implementar de dos maneras:

- Por un lado, se podrían hacer uso de las *funciones* de PDDL2.1, pero tras varias pruebas, se vio que el planificador Sayphi no funciona correctamente con dichas funciones.
- Haciendo uso de nuevos objetos, que podríamos denominar “numero”, y crear tantos como fuese necesario “cero uno dos tres...”. También se crearía un predicado que los relacione de dos en dos, al igual que otro que relacione el grupo con su número. De esta manera tendríamos una escala que nos indicaría en todo momento la ocupación del grupo, que variaría en las acciones

correspondientes. Es un método sencillo y efectivo, por lo que fue escogido como solución temporal.

Una vez solucionado el problema de la ocupación, nos sumergimos en el siguiente: el movimiento. Para mover a varios personajes a la vez, deben “introducirse” en un grupo, y dicho grupo debe ser quien se mueva por los lugares establecidos por el problema. Dicho en otras palabras, los personajes no están en un lugar, sino en un grupo que está en ese lugar. Cuando se van desarrollando más acciones, en casi su totalidad es necesario conocer en qué lugar está el personaje que ejecuta la acción. En otros lenguajes como Java se pueden implementar funciones del tipo *returnPlace()* que nos devuelva el lugar donde está el personaje, independientemente de si tiene que consultar el grupo o si está directamente en él, es transparente al planificador, ya que sólo se querría conocer el lugar donde se haya el personaje. Sin embargo, no es el caso de PDDL, cuya sencillez, que a veces es de agradecer, en este ámbito dificulta muchísimo su resolución. Para explicarlo con mayor claridad, pondremos el ejemplo de la acción “dar un objeto”:

Esta acción consiste en un personaje P1, que se encuentra en un determinado lugar L, tiene un objeto O, y le da a un personaje P2 dicho objeto en ese lugar L. Hay 5 posibilidades distintas en esta sencilla acción:

- El caso más sencillo consistiría en que ninguno de ellos, ni P1 ni P2 estuviesen en grupo. Con confirmar que ambos estén en el mismo lugar L podría ejecutarse la acción (dejando al margen de este ejemplo el resto de comprobaciones).
- Si P1 está en un grupo G y P2 no, ya no sirve la acción anterior, habría que crear otra que confirme que el lugar es el mismo para G y P2, ya que P1 no está en ningún lugar.
- El mismo problema surge si es P2 quien está en el grupo G y P1 no está en ninguno. La acción es distinta, ya que hay que diferenciar entre emisor y receptor del objeto.
- Por supuesto, ambos pueden estar en dos grupos distintos G1 y G2, en cuyo caso, son sus lugares los que debemos consultar.
- Y por último, si ambos están en el mismo grupo, resulta irrelevante el lugar, ya que sabemos con certeza que están en el mismo al tratarse de miembros de un mismo grupo. Aunque por simplificar la computación al planificador, esta opción funcionaría correctamente en la anterior, siempre que no haya restricción de que los grupos sean diferentes.

Es decir, hasta un total de 5 acciones para contemplar cada acción que requiera saber el lugar en el que están los personajes. Si la acción es de un solo personaje serían solo dos (con o sin grupo) y si no requiere el lugar, tan sólo una. Por desgracia, teniendo en cuenta la gran cantidad de acciones que supone una narración de este tipo, ha tenido que ser descartada esta idea ya que no resulta escalable, afectando directamente al rendimiento.

5.1.2 Movimiento conjunto

Seguimos teniendo el mismo problema que en un principio, necesitamos una manera de que los personajes se muevan juntos. La otra alternativa más simple es el **movimiento conjunto**. Al igual que una acción de movimiento requiere de un personaje, un origen y un destino, tan sólo habría que añadir otro personaje a la ecuación: dos personajes que estén en un origen van a un destino.

Resulta una acción compleja, pero efectiva sin duda. Su complejidad se debe al propio dominio, ya que para que tuviese sentido, se deben incluir hechos como que sean amigos, o quieran ir al mismo lugar... Al ejecutar problemas con y sin la acción de movimiento conjunto se puede apreciar el tiempo extra que genera al ser tan compleja, ya que es casi como dos acciones en una.

Al experimentar con esta metodología, también se ha intentado hacer la acción de tres personajes. Es igual de simple de desarrollar, pero su complejidad se eleva exponencialmente, ya que hay que tratar a los personajes entre sí para comparar su amistad o deseos. Su coste también es exponencial, ya que incluso problemas relativamente sencillos requieren una gran cantidad de memoria.

Tiene la gran ventaja que no tenía el desplazamiento por grupos: es tan sólo una acción, aunque muchísimo más compleja. Sin embargo, como estudiaremos en detalle a continuación, esto no es cierto del todo. No se trata de una simple acción, ya que como comentamos antes, requiere conocer el “deseo” de ir a un lugar por parte de los personajes. Se desarrollarán acciones que enseñen a los personajes a qué lugar deben querer ir. Es decir, primero querrán saber a dónde ir, y luego irán, 2 acciones con un simple personaje. Si son dos personajes, 2 acciones para crear el deseo de ir, y una de movimiento conjunto. Como podemos observar, tampoco satisface nuestras necesidades de algo simple pero efectivo, ya que requiere un gran coste y de acciones auxiliares que generen su deseo de ir al lugar de destino.

5.1.3 Omnisciencia

Siguiendo con nuestro dilema del movimiento, además de viajar en grupo, la idea principal consistía en generar en el problema un mapa donde los lugares estuvieran interconectados. De esta manera, si por ejemplo tuviésemos tres lugares A, B y C tales que A está conectado con B y B con C, habría que pasar obligatoriamente por B para ir de A a C. Desde el punto de vista de la narración parece la metodología más interesante, ya que se podría ver la travesía que recorren los personajes para conseguir sus fines.

Como comentamos en el apartado anterior, los personajes se van a regir por un deseo, ya sea de ir a un lugar, conseguir un objeto... Surgen así nuestras siguientes complicaciones:

- Si un personaje está en un lugar A (siguiendo el ejemplo anterior) y en un determinado momento desea ir a C, ¿cómo calcular la ruta que le dirija allí? Tendría que disponer de un planificador para el simple hecho del movimiento.

- Si un personaje necesita un objeto, ¿cómo sabrá dónde ir? Algo que se ha aprendido en este proyecto es que los deseos pueden ser imprecisos. Es decir, supongamos que el personaje está en A, y de alguna manera se entera de que hay un objeto en C que él necesita, pero el planificador en lugar de centrarse en que este personaje vaya a C, sigue con otras acciones. Antes de que el personaje decida ir a C, el objeto puede haber cambiado de ubicación a otra D por interacción con otros personajes, y nuestro personaje podría desear buscarlo de nuevo, queriendo ir a D. Se habría llegado a un punto en que el personaje quiere ir a C y D al mismo tiempo, y tan sólo le sería útil uno de los lugares.

Para evitar esta serie de problemas, se ha optado por la *omnisciencia*, es decir, todos los personajes son capaces de saber el estado de todos los objetos en todo momento. Por otro lado, también se ha decidido implementar los lugares sin comunicación entre ellos, haciendo así posible un **movimiento libre** de los personajes. Ahora, un personaje P1 situado en A puede ir a C en una acción, siendo así más sencillo implementar el viaje si un personaje quiere ir a un lugar.

5.1.4 Movimiento final

Terminando la metodología del movimiento, hasta ahora tenemos:

- Omnisciencia, que nos permite saber todo en todo momento.
- No hay restricciones al viajar, si un personaje quiere ir a un lugar, puede viajar directamente.
- Movimiento en solitario.
- Varias acciones que determinen el deseo por ir a un lugar.

Hasta ahora, hemos reducido bastante el coste en acciones para simbolizar los viajes, pero aún se puede perfilar un poco más: unir el movimiento a los deseos.

Es decir, por el momento, un personaje para desplazarse de un lugar A a otro B necesita estar en A y desear ir a B. Por tanto, para realizar dicho movimiento deberá hacer las siguientes acciones:

1. Desear ir a un lugar.
2. Ir a un lugar deseado.

Se puede simplificar más, si **en las acciones de generar el deseo implantamos el movimiento directamente**, deshaciéndonos de la necesidad de querer ir a algún lugar. De tal manera que tendremos varias acciones de viajar, cada una con su lógica, pero de tal manera que el viaje se realiza en una sola acción.

5.1.4 Motivaciones

La primera impresión al comenzar con esta narrativa, para clasificar a los personajes, fue la de declararlos *buenos* o *malos*. Entre ellos podrían pelear, los que fuesen iguales podrían interactuar con objetos... pero no sería práctico ya que no describirían un comportamiento interesante, los malos matan a los buenos, los buenos matan a los malos y siempre sería lo mismo. Partiendo de esta base, se ha evolucionado la idea para dar lugar a los **motivos**.

Los motivos o motivaciones son objetos abstractos que poseen los personajes que mantienen la base bueno-malo, es decir, al igual que hay motivos buenos también habrá motivos malos.

Supongamos en nuestro caso la motivación “Destruir el anillo”, un motivo bueno. Este motivo tendrá su propio sub-objetivo o *sub-goal*, en su caso, que el objeto anillo sea destruido. Los personajes que tengan esta motivación aprenderán lo necesario para llevar a cabo su *sub-goal*, y al ser un motivo bueno, aquellos que tengan otro malo como “Adquirir el anillo” serán sus enemigos y tratarán de evitarlo, así como llevar a cabo su propio *sub-goal*.

Los motivos pueden relacionarse entre sí. Un motivo genérico como la “Paz” puede conseguirse a través de otros buenos como “Destruir el anillo”, de tal manera que podemos anidarlos a través de predicados y generar acciones que te enseñen nuevos motivos.

Además de enemigos, también pueden generar amigos si tienen un motivo en común, o bien si un amigo no comparte su motivo, puede pedirle que le ayude siguiendo su motivo.

Los motivos han resultado fáciles y útiles, son la base con la que se unen las acciones de forma coherente y crean el esqueleto principal de la narración.

5.2 Definición de tipos de objetos

Vamos a explicar los diferentes tipos de objetos que se han empleado para la realización del diseño del sistema.

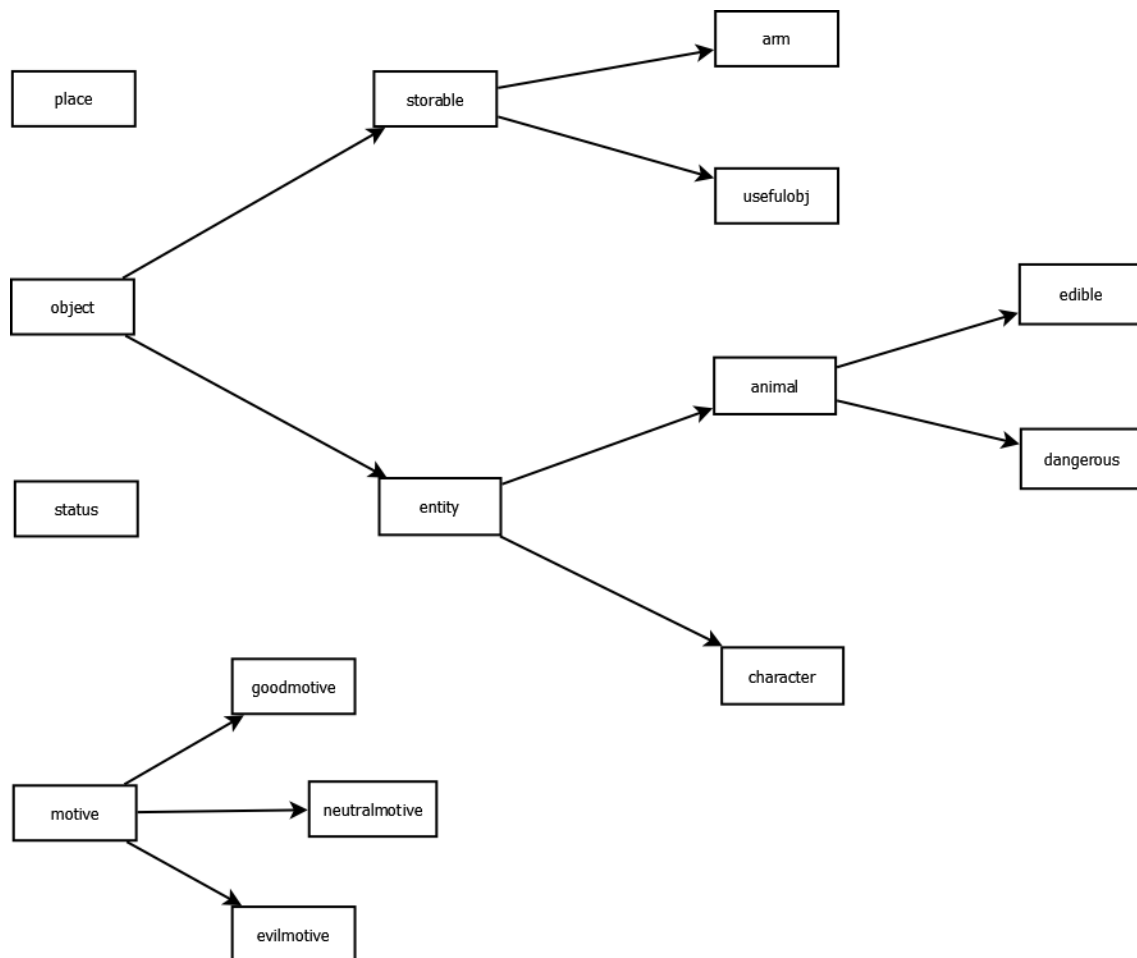


Ilustración 1 - Esquema de tipos de objetos

En primer lugar tenemos el genérico **object** que engloba los objetos físicos que pueden localizarse en un lugar. Por esta razón el objeto **place** ha quedado al margen. Estos **objects** pueden ser **storable**, es decir, objetos que se pueden almacenar o llevar consigo como **arm** (armas como espadas, arcos, hachas...) o **usefulobj** (objetos útiles como un anillo, medicina, un bastón...). O bien pueden ser **entities** que serían las criaturas vivas como los **characters** (personajes principales) o **animals** diferenciados entre **edibles** o **dangerous** (comestibles o peligrosos, como un conejo o un lobo respectivamente).

Por otro lado, los objetos abstractos son los **status**, para definir los estados en los que se encuentran algunos objetos (como destruido, adquirido...) y los **motives**, los motivos que pueden clasificarse, según su naturaleza como **goodmotive**, **neutralmotive** o **evilmotive**, según si son buenos, neutrales o malos.

5.3 Definición de predicados

Para el uso de estos objetos se han definido una serie de predicados que los interconectarán para dar lugar a relaciones lógicas con las que actuar en las futuras acciones.

5.3.1 Lugares

(at ?obj - object ?place - place)

"?obj is at ?place"

Simboliza la ubicación de un objeto en un lugar.

(can-be-destroyed-in ?obj - object ?place - place)

"?obj can be destroyed in ?place"

Con este predicado se puede establecer un lugar donde un objeto puede ser destruido.

5.3.2 Estado de personajes

(hurt ?char - character)

"?char is hurt"

(dead ?ent - entity)

"?ent is dead"

Los personajes pueden estar en tres estados diferentes: sano, herido o muerto. Sin embargo, los animales solo pueden estar sanos o muertos, por lo que en estos predicados solo los *characters* pueden usar el predicado *hurt*, y todas las *entities* pueden estar *dead*. La ausencia de alguno de estos predicados simboliza el estado sano.

(invisible ?char - character)

"?char is invisible"

Algunos objetos pueden otorgar invisibilidad, lo que impedirá que puedan herir o matar a un personaje.

(weak ?char - character)

"?char is weak"

(strong ?obj - object)

"?obj is strong"

Algunos personajes, como en el caso de los *hobbits* en nuestra narrativa enfocada en El Señor de los Anillos, serán débiles, lo que les impedirá herir a otros sin el uso de algún arma. Por el contrario, algunos personajes serán fuertes, y tan sólo podrán ser asesinados por medio de algún arma que también sea fuerte, de ahí que todos los objetos puedan serlo, para utilizar el mismo predicado para dichos *characters* y *arms*.

(isNPC ?ent - entity)

"?ent is a NPC"

En ocasiones, a la narración no le interesa que algunas *entities* tengan demasiada importancia. Para estos casos, se declaran como *NPCs* (*Non-Player Character*) como se suele hacer en algunos videojuegos. Este predicado impedirá el uso de ciertas acciones, como viajar, para crear personajes más estáticos.

(killed-by ?ent - entity ?ent2 - entity)

"?ent has been killed by ?ent2"

Algunas acciones permitirán matar a un enemigo, el cual será declarado como *dead*, pero además, también es interesante guardar la información sobre quien ha sido el responsable, de tal manera que podemos forzar que a una entidad lo mate alguien en especial en el objetivo.

(is ?obj - object ?st - status)

"?obj is ?st"

Los objetos en ocasiones pueden estar en algún estado especial (definido por el problema, aunque usados de manera global) como por ejemplo "*Destroyed*". De esta manera en los objetivos se puede fijar que un objeto acabe de alguna forma predefinida.

5.3.3 Amistad

(friends ?char - character ?char2 - character)

"?char and ?char2 are friends"

(enemies ?char - character ?char2 - character)

"?char and ?char2 are enemies"

Los personajes, según sus intereses, podrán establecer tres tipos de relaciones sociales con el resto de personajes. Pueden ser amigos, lo que les dará lugar a hacer acciones amistosas, o enemigos que les permitirá enfrentarse, o bien la ausencia de ambos, que indicaría una situación neutral.

5.3.4 Objetos

(have-an-obj ?char - character ?obj - storable)

"?char has got a ?obj"

(wish-an-obj ?char - character ?obj - storable)

"?char wishes a ?obj"

Los personajes pueden llevar consigo objetos. Cuando un objeto pertenece a un personaje, el objeto no se encuentra en ningún lugar, sino que el personaje lo lleva consigo. Por otro lado, para conseguir un objeto, salvo que lo encuentre de casualidad, debe desearlo, por lo que se define también este predicado.

(magic-invisible ?obj - usefobj)

"?obj is magic (invisibility)"

(magic-kill ?obj - usefobj)

"?obj is magic (kill)"

(magic-heal ?obj - usefobj)

"?obj is magic (heal)"

Algunos *usefobj*, los objetos útiles, pueden ser mágicos. Se han definido tres tipos de magia: invisibilidad, poder matar y sanación. Ya hemos hablado de la invisibilidad, que vuelve invisible a un personaje ocultándolo de enemigos, haciendo imposible que se le pueda dañar. La habilidad mágica *kill* permite emplear como arma un objeto, con la diferencia que al tratarse de magia, no hiere, sino que mata de un golpe. Por último, la sanación permite eliminar el estado *hurt* de un personaje, volviéndolo sano. Un objeto puede tener varios tipos de magia a la vez.

(can-use-magic-kill ?char - character)

"?char can use magic (kill)"

(can-use-magic-heal ?char - character)

"?char can use magic (heal)"

Estos objetos mágicos no puede usarlos cualquier personaje. Debe ser un personaje con conocimiento de dicha magia. Estos predicados indican dicho conocimiento, sin contar con la invisibilidad, que asumimos que todo el mundo sería capaz de usarla si tiene el objeto en cuestión. Un personaje puede ser capaz de emplear varios tipos de magia a la vez.

5.3.5 Motivos

(have-motive ?char - character ?m - motive)

"?char has the motive ?m"

Como vimos anteriormente, los personajes se dirigen mediante motivos. Estos motivos pueden ser buenos o malos y marcarán su propio objetivo. Con este predicado marcamos la unión directa entre un personaje y su motivo, pudiendo tener más de uno para completar sus *sub-goals*.

(implies-that ?m - motive ?m2 - motive)

"?m implies-that ?m2"

Los motivos pueden estar relacionados. Para un motivo genérico, se puede completar mediante uno más específico. De tal manera que si un personaje tiene un motivo *?m*, podría querer tener el motivo *?m2* para completarlo.

(want-motive ?m - motive ?obj - object ?st - status)

"?m wants ?obj be ?st"

Para definir estos *sub-goals*, se hace uso de un predicado que relacione el motivo en cuestión con el objeto (pudiendo ser una *entity* o *storable*) y el estado en que lo quiere, por ejemplo, *"Destroyed"*.

(motive-complete ?char - character ?m - motive)

"?char has complete his motive ?m"

Por último, si un personaje completa este *sub-goal*, podemos almacenarlo en un predicado que relacione el personaje con el motivo completado para establecerlo como meta en los problemas.

5.4 Definición de acciones

Con estos predicados ya podemos crear las acciones del dominio que generen una historia aleatoria a la par que interesante, en las que los personajes luchen, viajen, interactúen con objetos y aprendan.

5.4.1 Interactuar con objetos

Las acciones que manipulan objetos son bastante básicas, son aquellas que permiten coger, dar, intercambiar o despojar:

Take-obj

```
(:action take-obj ; Coger un objeto
  :parameters (?char - character ?obj - storable ?place
    - place)
  :precondition (and
    (not (dead ?char))
    (at ?char ?place)
    (at ?obj ?place)
  )
  :effect (and
    (not (at ?obj ?place))
    (have-an-obj ?char ?obj)
  )
)
```

La acción de coger un objeto es la más básica. Hay un personaje y un objeto en el mismo lugar, y el efecto consiste en que el objeto ya no está en ese lugar y lo tiene el personaje. Como podemos apreciar, como sucederá en casi todas las acciones, nos aseguramos de que el personaje no esté muerto, ya que como puede imaginarse, una vez muerto no puede realizar acciones.

No hay restricción de deseo en esta acción. Esto es debido a que simula un “encuentro casual” entre el personaje y el objeto, de tal manera que si un personaje ve un objeto, aunque no lo quiera, ¿por qué no cogerlo? Tampoco hay restricción en cuanto a la invisibilidad, asumiendo así que se puede coger siendo invisible.

Sin embargo, no se ha desarrollado una acción *drop* que haga lo contrario, que suelte un objeto en un lugar. En un principio se desarrolló, pero su efectividad era muy deficiente, ya que no tenía sentido para la narración.

Give-obj

```
(:action give-obj ; Dar un objeto
  :parameters (?char - character ?char2 - character
    ?obj - storable ?place - place)
  :precondition (and
    (not (dead ?char))
    (not (dead ?char2))
    (not (= ?char ?char2))
    (at ?char ?place)
    (at ?char2 ?place)
    (have-an-obj ?char ?obj)
    (not (invisible ?char))
    (not (invisible ?char2))

    ; El que lo da no debe quererlo
    (not (wish-an-obj ?char ?obj))

    ; Deben ser amigos
    (or
      (friends ?char ?char2)
      (friends ?char2 ?char)
    )
  )
  :effect (and
    (not (have-an-obj ?char ?obj))
    (have-an-obj ?char2 ?obj)
  )
)
```

Esta es la acción de dar un objeto a otro personaje. Ya requiere de muchas más precondiciones al ser una acción entre dos personajes. Podemos ver una nueva comprobación para asegurarnos que ambos no son el mismo personaje, ya que podría darse el caso en el planificador. Ambos deben estar en el mismo lugar y visibles, o no podrían darse el objeto. Por último, introducimos la mecánica de deseos y amistad. El que da el objeto no debe quererlo. Sin embargo, no establecemos que el receptor lo quiera, ya que esta acción está enfocada a ser un “regalo”, porque son amigos.

Aunque en las siguientes acciones veremos que cuando dos personajes se hacen amigos se crean los dos predicados de amistad (P1 con P2 y P2 con P1) en las acciones las comprobaremos mediante un *or*, ya que en los problemas no tiene porqué ser así. Sin embargo, si asumiésemos que en los problemas se declaran siempre ambos predicados al establecer amistad o enemistad, se podría eliminar este *or*, ya que es más complejo computacionalmente.

El efecto es sencillo, el que tenía el objeto ya no lo tiene y el que no lo tenía, ahora lo tiene.

Trade-obj

```
(:action trade-obj ; Comerciar con un objeto
:parameters (?char - character ?char2 - character
?obj - storable ?obj2 - storable ?place - place)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (= ?char ?char2))
  (at ?char ?place)
  (at ?char2 ?place)
  (have-an-obj ?char ?obj)
  (have-an-obj ?char2 ?obj2)
  (not (invisible ?char))
  (not (invisible ?char2))

  (wish-an-obj ?char2 ?obj)
  (wish-an-obj ?char ?obj2)
)
:effect (and
  (not (have-an-obj ?char ?obj))
  (not (have-an-obj ?char2 ?obj2))
  (have-an-obj ?char2 ?obj)
  (have-an-obj ?char ?obj2)
)
)
```

Esta vez hay dos objetos en juego. Ahora se requiere de un deseo mutuo por el objeto del contrario, ya que no requiere amistad, se trata de un *quid pro quo*. El efecto, como en el caso anterior, es que los objetos cambian de manos.

Loot-obj

```
(:action loot-obj ; Despojar un objeto
:parameters (?char - character ?char2 - character
?obj - storable ?place - place)
:precondition (and
  (dead ?char2)
  (not (dead ?char))
  (at ?char ?place)
  (at ?char2 ?place)
  (have-an-obj ?char2 ?obj)
)
:effect (and
  (not (have-an-obj ?char2 ?obj))
  (have-an-obj ?char ?obj)
)
)
```

Despojar un objeto consiste en arrebatarse un objeto a un personaje muerto, independientemente de la amistad o el deseo. Una acción muy similar a las anteriores.

5.4.2 Herir y matar

Estas acciones representarán la lucha entre los personajes. Como explicamos en el apartado anterior con los predicados, los personajes primero deben estar heridos antes de morir, salvo casos excepcionales como los fuertes o los animales, que mueren al primer golpe.

Hurt-punch

```
(:action hurt-punch
  :parameters (?char - character ?char2 - character
    ?place - place)
  :precondition (and
    (enemies ?char ?char2)
    (not (= ?char ?char2))
    (not (dead ?char))
    (not (dead ?char2))
    (at ?char ?place)
    (at ?char2 ?place)
    (not (hurt ?char2))
    (not (invisible ?char2))
    ; No debe ser debil, si lo es, debe ser
    invisible, asi que...
    ; O bien no es debil o es invisible
    (or
      (not (weak ?char))
      (invisible ?char)
    )
    (not (strong ?char2))
  )
  :effect (and
    (hurt ?char2)
  )
)
```

Comenzamos con la acción de lucha más básica, herir mediante puñetazos, una acción que no requiere de armas. Al igual que en las anteriores acciones, cada vez que interactúan dos personajes debemos asegurarnos de que no estén muertos, que estén en el mismo lugar y que no sean el mismo personaje.

Por supuesto, se comprueba que sean enemigos o no tendría sentido la acción. También es importante que el personaje que va a ser herido no esté ya herido o no tendría sentido ejecutar esta acción, así como que no sea invisible o fuerte, o no se le podría golpear. Por último, el que golpea no debe ser débil, o si bien lo es, debe ser invisible para asestar el golpe, debe cumplir una de esas dos condiciones. Como resultado, el que recibe el golpe pasa a estar herido.

Hurt-arm

```
(:action hurt-arm
  :parameters (?char - character ?char2 - character
    ?place - place ?arm - arm)
  :precondition (and
    (enemies ?char ?char2)
    (not (dead ?char))
    (not (dead ?char2))
    (not (= ?char ?char2))
    (not (hurt ?char2))
    (at ?char ?place)
    (at ?char2 ?place)
    (have-an-obj ?char ?arm)
    (not (invisible ?char2))
    (not (strong ?char2))
  )
  :effect (and
    (hurt ?char2)
  )
)
```

En esta acción, un personaje hiere a otro mediante el uso de un arma. En comparación con la anterior, en este caso el atacante puede ser débil, ya que el arma le otorga la oportunidad de herir fácilmente, y da igual si es invisible o no. El resto de la acción es idéntica a la anterior.

Kill-arm

```
(:action kill-arm
  :parameters (?char - character ?char2 - character
    ?place - place ?arm - arm)
  :precondition (and
    (not (dead ?char))
    (not (= ?char ?char2))
    (at ?char ?place)
    (at ?char2 ?place)

    (enemies ?char ?char2)

    (have-an-obj ?char ?arm)
    (hurt ?char2)

    (not (invisible ?char2))
    (not (strong ?char2))
  )
  :effect (and
    (not (hurt ?char2))
    (dead ?char2)
    (killed-by ?char2 ?char)
  )
)
```

Matar a un personaje es idéntico a herirlo, con la premisa de que el personaje que va a ser asesinado ya esté herido. Por supuesto, al hacer efecto esta acción, el personaje pasará de herido a muerto, y se guardará quién ha sido su asesino.

Si nos fijamos detenidamente, no comprobamos que *?char2* no esté muerto, ya que es imposible que esté muerto y herido a la vez, con confirmar que está herido ya conocemos su estado.

5.4.3 Animales

Al haber varios tipos de animales, hay que tratarlos de forma separada en acciones distintas. Su función es simple: los animales comestibles se pueden comer para recuperar las heridas, y los peligrosos pueden atacar al personaje como haría cualquier otro, pero puede matarle sin el uso de armas.

Kill-and-eat-animal

```
(:action kill-and-eat-animal
  :parameters (?char - character ?animal - edible
    ?place - place ?arm - arm)
  :precondition (and
    (not (dead ?char))
    (not (dead ?animal))
    (at ?char ?place)
    (at ?animal ?place)
    (have-an-obj ?char ?arm)

    ; Debe estar herido para comer y restaurarse
    (hurt ?char)
  )
  :effect (and
    (not (hurt ?char))
    (dead ?animal)
  )
)
```

Esta acción recrea la caza de un animal comestible para curar sus heridas. Para matar a dicho animal, se debe disponer de un arma y, por supuesto, estar herido y estar en el mismo lugar. Como efecto, el personaje deja de estar herido y el animal muere.

Hurt-by-animal

```
(:action hurt-by-animal
  :parameters (?char - character ?animal - dangerous
    ?place - place)
  :precondition (and
    (not (dead ?char))
    (not (dead ?animal))
    (not (hurt ?char))
    (at ?char ?place)
    (at ?animal ?place)
    (not (invisible ?char))
    (not (strong ?char))
  )
  :effect (and
    (hurt ?char)
  )
)
```

Si se va a un lugar donde hay animales peligrosos, estos pueden atacar a dicho personaje. En primer lugar, como indica esta acción, el animal le hiere, al igual que las acciones *hurt-punch* y *hurt-arm*, como efecto, el personaje acaba herido.

Kill-by-animal

```
(:action kill-by-animal
  :parameters (?char - character ?animal - dangerous
    ?place - place)
  :precondition (and
    (not (dead ?char))
    (not (dead ?animal))

    (hurt ?char)

    (at ?char ?place)
    (at ?animal ?place)
    (not (invisible ?char))
    (not (strong ?char))
  )
  :effect (and
    (not (hurt ?char))
    (dead ?char)
    (killed-by ?char ?animal)
  )
)
```

Nuevamente, la acción de que un animal mate a un personaje herido es muy similar a la anterior, tan sólo debemos asegurarnos de que el personaje esté herido y se le pueda atacar (comprobar invisibilidad y fuerza). Si se cumple la acción, el animal habrá matado al personaje, cambiando su estado a *dead* y almacenando su muerte a manos del animal.

Kill-animal

```
(:action kill-animal
  :parameters (?char - character ?animal - dangerous
    ?place - place ?arm - arm)
  :precondition (and
    (not (dead ?char))
    (not (dead ?animal))
    (have-an-obj ?char ?arm)
    (at ?char ?place)
    (at ?animal ?place)
    (not (invisible ?char))
  )
  :effect (and
    (dead ?animal)
    (killed-by ?animal ?char)
  )
)
```

En esta acción, un personaje armado mata a un animal peligroso. Como podemos observar, el animal no es necesario que esté herido previamente como pasa con los personajes. A diferencia de los otros personajes, asumimos que los animales pueden detectar a los personajes invisibles mediante sus sentidos más agudizados, por tanto no debe ser invisible. Como sucede en todas las acciones de matar, el animal muere y se almacena su muerte a manos del personaje.

5.4.4 Magia

Los *usefulobj*, u objetos útiles, en ocasiones pueden tener propiedades mágicas. Pero eso no significa que todos puedan usarlos, tan sólo aquellos que dispongan del conocimiento necesario.

Kill-magic

```
(:action kill-magic
  :parameters (?char - character ?char2 - character
    ?place - place ?obj - usefulobj)
  :precondition (and
    (not (dead ?char))
    (not (dead ?char2))
    (at ?char ?place)
    (at ?char2 ?place)

    (enemies ?char ?char2)

    (have-an-obj ?char ?obj)
    (magic-kill ?obj)
    (can-use-magic-kill ?char)
    (not (invisible ?char2))
    (not (strong ?char2))
  )
  :effect (and
    (not (hurt ?char2))
    (dead ?char2)
    (killed-by ?char2 ?char)
  )
)
```

Comenzamos con la acción de matar a otro personaje mediante magia. Como es lógico, el personaje que vaya a emplear la magia debe tener un objeto que tenga dicho poder, así como el conocimiento de usar esa magia. El resto de predicados siguen la misma estructura de las acciones de pelea, con la excepción de que al tratarse de magia, no es necesario que el personaje atacado esté herido para morir, ya que morirá igualmente.

Heal-magic

```
(:action heal-magic
  :parameters (?char - character ?char2 - character
    ?place - place ?obj - usefuleobj)
  :precondition (and
    (not (dead ?char))
    (at ?char ?place)
    (at ?char2 ?place)
    (not (= ?char ?char2))

    ;Para sanarle, debe conocerle, por tanto deben
    ser amigos al menos
    (or
      (friends ?char ?char2)
      (friends ?char2 ?char)
    )

    (hurt ?char2)

    (have-an-obj ?char ?obj)
    (magic-heal ?obj)
    (can-use-magic-heal ?char)

    (not (invisible ?char2))
  )
  :effect (and
    (not (hurt ?char2))
  )
)
```

En esta acción un personaje cura mediante magia (o un objeto curativo) a un amigo. Por tanto, deben ser amigos y el personaje al que se va a curar debe estar herido o no habría nada que curar. El objeto debe tener la propiedad mágica de curar, así como el que realiza la acción de curar debe tener los conocimientos de curación. Al realizar la acción, el segundo personaje estará curado, por tanto no tiene heridas.

Heal-magic-himself

```
(:action heal-magic-himself
  :parameters (?char - character ?obj - usefobj)
  :precondition (and
    (not (dead ?char))

    (hurt ?char)

    (have-an-obj ?char ?obj)
    (magic-heal ?obj)
    (can-use-magic-heal ?char)
  )
  :effect (and
    (not (hurt ?char))
  )
)
```

Esta vez, el personaje se cura a sí mismo, por tanto es irrelevante el lugar en el que esté. Tan sólo debe estar herido y tener el objeto mágico y saber usarlo. Al ejecutar la acción, se habrá quitado las heridas.

Magic-become-invisible

```
(:action magic-become-invisible
  :parameters (?char - character ?obj - usefobj)
  :precondition (and
    (not (dead ?char))
    (have-an-obj ?char ?obj)
    (magic-invisible ?obj)
    (not (invisible ?char))
  )
  :effect (and
    (invisible ?char)
  )
)
```

A diferencia de las otras magias, la invisibilidad no requiere de habilidad, cualquiera puede usarlo. Por tanto, esté donde esté, el personaje tan sólo necesita un objeto con la habilidad mágica de invisibilidad para volverse invisible, y por supuesto, no serlo ya.

Magic-become-visible

```
(:action magic-become-visible
  :parameters (?char - character ?obj - usefobj)
  :precondition (and
    (not (dead ?char))
    (invisible ?char)
    (have-an-obj ?char ?obj)
    (magic-invisible ?obj)
  )
  :effect (and
    (not (invisible ?char))
  )
)
```

Esta acción vuelve visible a un personaje invisible. Exactamente igual que la acción anterior, pero al revés, empieza siendo invisible y pasa a no serlo.

5.4.5 Motivos

Las siguientes acciones son las más importantes, son las que dotan de sentido el resto, guiando a los personajes a perseguir objetivos, desear objetos, creando amistades y enemistades...

Same-motive-friends

(:action same-motive-friends ; Pedirle a un amigo que siga tu motivo

```
:parameters (?char - character ?char2 - character
?place - place ?m - motive)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (invisible ?char))
  (not (invisible ?char2))

  (or
    (friends ?char ?char2)
    (friends ?char2 ?char)
  )

  (have-motive ?char ?m)
  (not (have-motive ?char2 ?m))
)
:effect (and
  (have-motive ?char2 ?m)
)
)
```

Esta acción simula una petición de un personaje a un amigo para que siga su motivo, es decir, que le ayude a que el *sub-goal* del motivo sea completado. Como siempre, al tratarse de una acción física, no deben estar muertos, ni ser invisibles, deben estar en el mismo lugar y deben ser amigos. Con esta acción, conseguimos que los personajes transmitan los motivos entre sus amigos.

Make-friends-motive

```
(:action make-friends-motive ; Hacerse amigo de alguien por
algun motivo comun
:parameters (?char - character ?char2 - character
?place - place ?m - motive)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (invisible ?char))
  (not (invisible ?char2))

  (have-motive ?char ?m)
  (have-motive ?char2 ?m)

  (not (friends ?char ?char2)) ; Si ya son amigos
no tiene sentido esta accion
  (not (friends ?char2 ?char))
)
:effect (and
  (friends ?char ?char2)
  (friends ?char2 ?char)
  (not (enemies ?char ?char2)) ; Recalcamos el
hecho de que no sean enemigos
  (not (enemies ?char2 ?char)) ; Asi no pueden ser
amigos y enemigos a la vez
)
)
```

Esta sería la acción contraria a la anterior. En la anterior, dos amigos comparten un motivo. En esta, dos personajes con un motivo común se hacen amigos. Como podemos observar en las precondiciones, los personajes tienen el mismo motivo y no son amigos. En el efecto, se convierten en amigos, y lo más importante, dejan de ser enemigos, si no, podrían ser amigos y enemigos al mismo tiempo. Veremos esta metodología en las siguientes acciones muy a menudo.

En su lugar, también se podría haber seguido una metodología más estricta, indicando en las precondiciones que no fuesen enemigos, pero daría como resultado planes muy fijos, donde las amistades no cambiarían.

De la forma implementada, en cierto momento de la historia dos personajes pueden ser amigos pero más tarde dejar de serlo para convertirse en enemigos.

Make-friends-want-kill

```
(:action make-friends-want-kill ; Hacerse amigo de alguien por
algun enemigo comun
:parameters (?char - character ?char2 - character
?place - place ?enemy - character)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (dead ?enemy))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (invisible ?char))
  (not (invisible ?char2))

  (enemies ?char ?enemy)
  (enemies ?char2 ?enemy)

  (not (friends ?char ?char2)) ; Si ya son amigos
no tiene sentido esta accion
  (not (friends ?char2 ?char))
)
:effect (and
  (friends ?char ?char2)
  (friends ?char2 ?char)
  (not (enemies ?char ?char2)) ; Recalcamos el
hecho de que no sean enemigos
  (not (enemies ?char2 ?char)) ; Asi no pueden ser
amigos y enemigos a la vez
)
)
```

“El enemigo de mi enemigo es mi amigo” En esta acción, dos personajes establecen amistad a partir de un enemigo común que les une. Muy similar a la anterior, tan sólo cambia el motivo, que ahora no es necesario, que es cambiado por un enemigo que tengan ambos personajes.

Make-friends-friends

```
(:action make-friends-friends ; Hacerse amigo "Friends of my
friends are my friends"
:parameters (?char - character ?char2 - character
?place - place ?char3 - character)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (dead ?char3))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (= ?char3 ?char2))
  (not (= ?char ?char3))
  (not (invisible ?char))
  (not (invisible ?char2))

  (friends ?char ?char3)
  (friends ?char2 ?char3)

  (not (friends ?char ?char2)) ; Si ya son amigos
no tiene sentido esta accion
  (not (friends ?char2 ?char))
)
:effect (and
  (friends ?char ?char2)
  (friends ?char2 ?char)
  (not (enemies ?char ?char2)) ; Recalamos el
hecho de que no sean enemigos
  (not (enemies ?char2 ?char)) ; Asi no pueden ser
amigos y enemigos a la vez
)
)
```

“Los amigos de mis amigos son mis amigos” Muy similar que la acción anterior, en lugar de ser un enemigo en común, es un amigo. Esta es la última acción con la que es posible establecer amistad.

Es decir, que para que un personaje se haga amigo de otro, puede hacerlo mediante:

- Un motivo en común.
- Un enemigo en común.
- Un amigo en común.

A continuación vamos a describir las tres acciones que enfrentarán a los personajes entre sí, es decir, las acciones que crean enemigos.

Make-enemy-motive

```
(:action make-enemy-motive ; Hacerse "enemigo" de alguien por
algun motivo dispar
:parameters (?char - character ?char2 - character ?gm
- goodmotive ?em - evilmotive ?place - place)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (= ?char ?char2))

  (have-motive ?char ?gm)
  (have-motive ?char2 ?em)

  (not (enemies ?char ?char2)) ; Si ya son
enemigos no tiene sentido esta accion
  (not (enemies ?char2 ?char))
)
:effect (and
  (enemies ?char ?char2)
  (enemies ?char2 ?char)
  (not (friends ?char ?char2)) ; Recalcamos el
hecho de que no sean amigos
  (not (friends ?char2 ?char)) ; Asi no pueden ser
amigos y enemigos a la vez
)
)
```

La primera acción que crea enemistades es la que se basa en motivos dispares. Como veremos en los problemas, es la base de los enfrentamientos, ya que es la más sencilla y directa. Es una acción muy similar a la *make-friends-motive*, si los motivos de los dos personajes están en desacuerdo, esos personajes son declarados enemigos. Como podemos apreciar, es indiferente del lugar, ya que suponemos omnisciencia. Si no fuese así y se requiriese estar en el mismo lugar, muchas de las acciones de viajar no podrían ejecutarse jamás, lo detallaremos más adelante en dichas acciones.

La acción requiere de un motivo bueno y uno malo, y dos personajes (cada uno con uno de los motivos). Estos personajes no deben ser ya enemigos, al igual que hacíamos en las acciones de crear amistad. Los personajes acaban siendo enemigos y dejan de ser amigos en el efecto.

Make-enemy-friends

```
(:action make-enemy-friends ; Hacerse enemigo "Enemies of my
friends are my enemies"
:parameters (?char - character ?char2 - character
?place - place ?char3 - character)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (dead ?char3))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (= ?char3 ?char2))
  (not (= ?char ?char3))
  (not (invisible ?char))
  (not (invisible ?char2))

  (friends ?char ?char2)

  (enemies ?char2 ?char3)

  (not (enemies ?char ?char3)) ; Si ya son
enemigos no tiene sentido esta accion
  (not (enemies ?char3 ?char))
)
:effect (and
  (enemies ?char ?char3)
  (enemies ?char3 ?char)
  (not (friends ?char ?char3)) ; Recalcamos el
hecho de que no sean amigos
  (not (friends ?char3 ?char)) ; Asi no pueden ser
amigos y enemigos a la vez
)
)
```

“Los enemigos de mis amigos son mis enemigos” Una acción de gran lealtad, en la que intervienen tres personas. El personaje que ejecuta la acción, un amigo suyo, y un enemigo de este amigo. Éste último acaba convirtiéndose en enemigo del primero. Esta vez sí se requiere de un emplazamiento, ya que simula un dialogo entre los dos primeros personajes. El resto de la acción es igual a la anterior, con las comprobaciones pertinentes al tener un tercer integrante.

Make-enemy-kill-friend

(:action make-enemy-kill-friend ; Hacerse enemigo de alguien que ha matado a tu amigo

```
:parameters (?char - character ?char2 - character  
?place - place ?char3 - character)  
:precondition (and  
  (not (dead ?char))  
  (dead ?char2)  
  (at ?char ?place)  
  (at ?char2 ?place)  
  (not (= ?char ?char2))  
  (not (= ?char3 ?char2))  
  (not (= ?char ?char3))  
  (not (invisible ?char))  
  (not (invisible ?char2))  
  
  (friends ?char ?char2)  
  
  (killed-by ?char2 ?char3)  
  
  (not (enemies ?char ?char3)) ; Si ya son  
  enemigos no tiene sentido esta acción  
  (not (enemies ?char3 ?char))  
)  
:effect (and  
  (enemies ?char ?char3)  
  (enemies ?char3 ?char)  
  (not (friends ?char ?char3)) ; Recalamos el  
  hecho de que no sean amigos  
  (not (friends ?char3 ?char)) ; Asi no pueden ser  
  amigos y enemigos a la vez  
)  
)
```

Esta acción es muy similar a la anterior, pero bastante más interesante, ya que representa la venganza. Con esta acción, un personaje hará su enemigo a alguien que haya matado a algún amigo suyo. De nuevo, disponemos de tres personajes, el que realiza la acción, el amigo (muerto) de éste y el autor de la muerte. Aunque suponemos omnisciencia, de nuevo forzamos que se encuentren en el mismo lugar el personaje y su amigo muerto, para simular dramatismo (ve a su amigo muerto y quiere venganza). Sin embargo, hacemos uso de esta omnisciencia para enemistarse con el autor del asesinato, que no tiene por qué estar en ese lugar.

5.4.6 Aprendizaje y movimiento

Las siguientes acciones describirán el aprendizaje y los viajes que podrán aprender y realizar los personajes. Se trata de acciones simples, con pocas precondiciones, pero generan la estructura básica de intencionalidad que hará que los personajes decidan hacer las acciones con cierto sentido y tengan interés.

Learn-motive

```
(:action learn-motive
  :parameters (?char - character ?m - motive ?m2 -
  motive)
  :precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (implies-that ?m ?m2)
  )
  :effect (and
    (have-motive ?char ?m2)
  )
)
```

Como ya hemos hablado anteriormente, los motivos se pueden anidar entre ellos. Unos dan lugar a otros. Esta acción hace aprender al personaje que tiene un motivo, otro motivo relacionado con ese, que le ayudará a conseguir completar su motivo inicial.

Learn-motive-wish-obj

```
(:action learn-motive-wish-obj
  :parameters (?char - character ?m - motive ?obj -
  storable ?st - status)
  :precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (want-motive ?m ?obj ?st) ;Al ser storable, sea
    como sea el status como lo quiera, lo querra
    tener para hacerlo posible
  )
  :effect (and
    (wish-an-obj ?char ?obj)
  )
)
```

Para completar los motivos se necesitará que un objeto acabe de una cierta manera (destruido, adquirido...). Esta acción enseña al personaje que tiene un motivo a desear el objeto necesario para conseguir completar dicho motivo.

Learn-place-obj

```
(:action learn-place-obj
  :parameters (?char - character ?obj - storable ?place
    - place ?place2 - place)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    (wish-an-obj ?char ?obj)
    (at ?char ?place)
    (at ?obj ?place2)
  )
  :effect (and
    ;(want-to-go ?char ?place)
    (not (at ?char ?place))
    (at ?char ?place2)
  )
)
```

En esta acción, tras hacer la anterior, un personaje que desea un objeto quiere ir a donde está para conseguirlo. Si nos fijamos en la parte comentada, en un principio se desarrolló como una acción de deseo, pero como comentamos en el planteamiento, nos hemos basado en un *movimiento intencionado individual*, por tanto esta acción y las siguientes de aprendizaje se han convertido en *acciones de movimiento*.

Lo primero es verificar que el personaje no es un *NPC*, ya que si lo fuese no podría moverse. Debe desear un objeto que esté en otro lugar, entonces se realiza el viaje, se cambia el lugar inicial en el que estaba el personaje por el que está el objeto.

Como podemos observar en el efecto de la acción, en ningún momento el personaje obtiene el objeto. Se ha decidido dar libertad al planificador, para que así surjan oportunidades que puedan evitarlo.

Learn-place-obj-char

```
(:action learn-place-obj-char
  :parameters (?char - character ?char2 - character
    ?obj - storable ?place - place ?place2 - place)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    ;Puede que este muerto pero siga teniendo el
    objeto
    (wish-an-obj ?char ?obj)
    (at ?char ?place)
    (at ?char2 ?place2)
    (have-an-obj ?char2 ?obj)
  )
  :effect (and
    (not (at ?char ?place))
    (at ?char ?place2)
    ; No establecemos que sean enemigos para que
    tengan oportunidad de intercambiarlo o estar en
    deuda
  )
)
```

La acción anterior sólo sirve si el objeto no lo tiene nadie, si alguien lo posee, se debe buscar a ese alguien para interactuar con él. Como podemos leer en los comentarios, hay muchas posibilidades una vez se realice el movimiento, pero esta acción no debe inmiscuirse en lo que suceda a continuación.

Por un lado tal vez la persona que tiene el objeto esté muerta pero siga teniéndolo en su poder, así que no es necesario comprobar que no esté muerta. Por otro lado, si alguien tiene un objeto que tú deseas, la primera intención lógica sería pelear por él, pero tal vez se pueda comerciar, o bien sea un amigo y te lo quiera dar.

Learn-place-give-obj-char

(:action learn-place-give-obj-char ; Ir a darle un objeto a un amigo

```
:parameters (?char - character ?char2 - character  
?obj - storable ?place - place ?place2 - place)  
:precondition (and  
  (not (isNPC ?char))  
  (not (= ?place ?place2))  
  (not (dead ?char))  
  (not (dead ?char2))  
  (wish-an-obj ?char2 ?obj)  
  (at ?char ?place)  
  (at ?char2 ?place2)  
  (have-an-obj ?char ?obj)  
  (or  
    (friends ?char ?char2)  
    (friends ?char2 ?char)  
  )  
)  
:effect (and  
  (not (at ?char ?place))  
  (at ?char ?place2)  
)  
)
```

Esta acción ha sido originada al comprobar fallos en los problemas. Consiste en un personaje que le lleva a un amigo un objeto que éste desea. Cuando un *NPC* (que no puede moverse) necesita un objeto, es obvio que alguien debe llevárselo. Sin esta acción la única posibilidad era que un enemigo que tuviese dicho objeto fuese a “matarle” y este último fuese el asesinado, lo cual no resulta atractivo a la narración.

Con esta acción, se simboliza el querer llevarle un objeto a alguien, imprescindible para los *NPC* que no pueden moverse. Es la acción totalmente contraria a la anterior, esta vez el que tiene el objeto es quien va a viajar, y el que permanece estático es quien lo desea. Por supuesto, para que tenga sentido la acción, estos personajes deben ser amigos.

Learn-place-obj-be-destroyed

```
(:action learn-place-obj-be-destroyed
  :parameters (?char - character ?obj - storable ?place
    - place ?place2 - place ?m - motive)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    (have-an-obj ?char ?obj)
    (at ?char ?place)
    (can-be-destroyed-in ?obj ?place2)
    (have-motive ?char ?m)
    (want-motive ?m ?obj Destroyed)
  )
  :effect (and
    (not (at ?char ?place))
    (at ?char ?place2)
  )
)
```

Se podría decir que esta es una acción ajustada al problema, ya que hace uso de una variable global del tipo status “*Destroyed*”. Es necesaria para que el personaje complete su motivo, ya que le dirigirá al lugar donde puede completarlo.

El personaje tiene un objeto y quiere destruirlo. Bueno, ciertamente es su motivo quien desea que ese objeto esté destruido, así que aprenderá el lugar donde puede ser destruido e irá allí.

Learn-place-enemy

```
(:action learn-place-enemy
  :parameters (?char - character ?char2 - character
    ?place - place ?place2 - place)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    (not (dead ?char2))
    (not (weak ?char)) ; Si es debil no guerra ir a
    por el enemigo
    (enemies ?char ?char2)
    (at ?char ?place)
    (at ?char2 ?place2)
  )
  :effect (and
    (not (at ?char ?place))
    (at ?char ?place2)
  )
)
```

Con esta acción, un personaje que no sea débil puede ir a donde está un enemigo. Aunque sea raro, no habrá una acción similar que lleve a un personaje junto a su amigo, ya que carece de sentido, no tiene una finalidad definida. Ésta, por el contrario, simboliza el acto de querer ir a matar a un enemigo, de ahí que alguien débil no pueda realizarla. Para ir junto a un amigo se usarán el resto de acciones de movimiento.

Learn-place-food

```
(:action learn-place-food
  :parameters (?char - character ?animal - edible
    ?place - place ?place2 - place)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    (hurt ?char)
    (at ?char ?place)
    (at ?animal ?place2)
  )
  :effect (and
    (not (at ?char ?place))
    (at ?char ?place2)
  )
)
```

Una acción sencilla, que mueve a un personaje a un lugar donde haya un animal comestible para curarse las heridas. Por esta razón, deberá estar herido.

Learn-place-heal

```
(:action learn-place-heal
  :parameters (?char - character ?char2 - character
    ?place - place ?place2 - place ?obj - usefobj)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    (hurt ?char)
    (at ?char ?place)
    (at ?char2 ?place2)
    (have-an-obj ?char2 ?obj)
    (magic-heal ?obj)
    (can-use-magic-heal ?char2)
  )
  :effect (and
    (not (at ?char ?place))
    (at ?char ?place2)
  )
)
```

Muy similar a la anterior. En este caso en lugar de querer un animal para regenerarse, buscará un personaje que pueda curarle. Este personaje no es necesario que sea amigo suyo, ni añadimos al efecto de la acción que esté curado, ya que así dejamos a la narración continuar una vez estén juntos.

Esta ha sido la última acción que implicará desplazamiento, siendo estos todos los tipos de movimiento:

- Ir a un lugar donde está un objeto deseado.
- Ir a un lugar donde está un personaje que tiene un objeto deseado.
- Ir a un lugar a llevarle a un amigo un objeto que desea.
- Ir a un lugar donde se puede destruir un objeto.
- Ir a un lugar donde está un enemigo.
- Ir a un lugar donde hay comida.
- Ir a un lugar donde un personaje puede curar.

5.4.7 Acciones que completan motivos

Estas acciones han sido desarrolladas para completar los motivos creados en los problemas que detallaremos más adelante.

Destroy-obj

```
(:action destroy-obj
  :parameters (?char - character ?m - motive ?obj -
    object ?place - place)
  :precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (want-motive ?m ?obj Destroyed)
    (at ?char ?place)
    (have-an-obj ?char ?obj)
    (can-be-destroyed-in ?obj ?place)
    (not (invisible ?char))
  )
  :effect (and
    (motive-complete ?char ?m)
    (is ?obj Destroyed)
    (not (have-an-obj ?char ?obj))
    ; Ya no tiene el motivo
    (not (have-motive ?char ?m))
    (have-motive ?char Neutral)
  )
)
```

En nuestros problemas encontraremos motivos que querrán que cierto objeto esté destruido. Para ello, cada motivo lo especificará en su correspondiente predicado, así como el lugar donde puede ser destruido. El personaje debe tener el motivo en cuestión, al igual que el objeto y estar en el lugar en el cual puede destruirlo. Al destruirlo, habrá completado ese motivo y lo olvidará, poniéndole uno neutral como comodín. El objeto es destruido, por tanto se indica en su estado y ya ni está en un lugar ni lo tiene nadie, es imposible acceder a él en ninguna otra acción a partir de ese momento.

Acquire-obj

```
(:action acquire-obj
  :parameters (?char - character ?m - motive ?obj -
    object)
  :precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (want-motive ?m ?obj Acquired)
    (have-an-obj ?char ?obj)
  )
  :effect (and
    (motive-complete ?char ?m)
    (is ?obj Acquired)
    ; Ya no tiene el motivo
    (not (have-motive ?char ?m))
    (have-motive ?char Neutral)
  )
)
```

Esta es la acción contraria a la de destruir un objeto. Por un lado, un motivo querrá destruirlo, mientras que otro lo querrá adquirir. Adquirir un objeto no es más que confirmar que se posee, completando así el motivo correspondiente, no requiere de ningún lugar concreto.

Kill-strong

```
(:action kill-strong
  :parameters (?char - character ?char2 - character
    ?place - place ?arm - arm ?m - motive)
  :precondition (and
    (not (dead ?char))
    (not (dead ?char2))
    (not (= ?char ?char2))
    (at ?char ?place)
    (at ?char2 ?place)
    (not (weak ?char))

    (enemies ?char ?char2)

    (have-an-obj ?char ?arm)

    (not (invisible ?char2))
    (strong ?char2)
    (strong ?arm)

    (have-motive ?char ?m)
    (want-motive ?m ?arm Acquired)
  )
  :effect (and
    (dead ?char2)
    (killed-by ?char2 ?char)
  )
)
```

Esta es la última de las acciones, también como las anteriores, para complementar a uno de los *sub-goals*. En esta acción, un personaje (no débil) tiene un arma poderosa que puede matar a alguien que es muy fuerte, y tan sólo puede ser asesinado con dicha arma. Es una acción de matar que requiere además un motivo que desee adquirir el arma poderosa, ya que el que va a ser asesinado será uno de los que quieran destruirla para que no puedan matarle.

5.5 Creación de oraciones

Estas acciones necesitan de un significado más narrativo más allá del título y sus parámetros. Al igual que se han descrito paso a paso cada una de ellas, al ver los planes generados debe quedar una secuencia de acciones fáciles de leer y comprender, por tanto se han traducido a un lenguaje natural cada una de ellas cuando se muestran por pantalla:

(Para facilitar la lectura se mostrarán ejemplos con nombres de los futuros problemas)

5.5.1 Interactuar con objetos

Take-obj: *"Frodo takes the object Ring at Mordor"*.

Give-obj: *"Frodo gives the object Ring to Sam at La_Comarca"*.

Trade-obj: *"Frodo trades the object Ring to Sam for ElfMedicine at La_Comarca"*.

Loot-obj: *"Frodo loots the object Ring from Bilbo (dead) at Rivendel"*.

5.5.2 Herir y matar

Hurt-punch: *"Frodo punches and hurts Sam at La_Comarca"*.

Hurt-arm: *"Frodo hurts Sam at La_Comarca with Sword"*.

Kill-arm: *"Frodo kills Sam at La_Comarca with Sword"*.

5.5.3 Animales

Kill-and-eat-animal: *"Frodo (hurt) kills and eats Rabbit at Rohan with Sword"*.

Hurt-by-animal: *"Wolf hurts Frodo at Rohan"*.

Kill-by-animal: *"Wolf kills Frodo at Rohan"*.

Kill-animal: *"Frodo kills Wolf at Rohan with Sword"*.

5.5.4 Magia

Kill-magic: *"Gandalf kills (magic) Gollum at Rivendel with GandalfStaff"*.

Heal-magic: *"Elrond heals (magic) Frodo (hurt) at Rivendel with ElfMedicine"*.

Heal-magic-himself: *"Elrond heals (magic) himself with ElfMedicine"*.

Magic-become-invisible: *"Frodo is invisible with Ring"*.

Magic-become-visible: *"Frodo is visible with Ring"*.

5.5.5 Motivos

Same-motive-friends: *"Frodo ask his friend Sam to follow his motive Destroy_Ring".*

Make-friends-motive: *"Frodo and Sam are friends because they have the motive Destroy_Ring in common".*

Make-friends-want-kill: *"Frodo and Sam are friends because they want to kill Sauron".*

Make-friends-friends: *"Frodo and Sam are friends because they are friends of Gandalf".*

Make-enemy-motive: *"Frodo (Destroy_Ring) and Sauron (Acquire_Ring) are enemies because their motives disagree".*

Make-enemy-friends: *"Frodo and Gollum are enemies because Sam (Frodo's friend) and Gollum are enemies".*

Make-enemy-kill-friend: *"Frodo and Gollum are enemies because Sam (Frodo's friend) has been killed by Gollum".*

5.5.6 Aprendizaje y movimiento

Learn-motive: *"Frodo has the motive Destroy_Ring because he wants to complete his motive Peace".*

Learn-motive-wish-obj: *"Frodo wishes the object Ring because he has the motive Destroy_Ring".*

Learn-place-obj: *"Frodo travels to Rivendel because he wishes the object Ring".*

Learn-place-obj-char: *"Frodo travels to Rivendel because he wishes the object Ring Bilbo has".*

Learn-place-give-obj-char: *"Orc travels to Mordor because he wants to give the object Ring to Sauron".*

Learn-place-enemy: *"Aragorn travels to Mordor because he wants to kill Sauron".*

Learn-place-food: *"Frodo travels to Rohan because he wants to eat a Rabbit".*

Learn-place-heal: *"Frodo travels to Rivendel because he wants Elrond to heal him".*

Learn-place-obj-be-destroyed: *"Frodo travels to Mount_Doom because he wishes to destroy the object Ring".*

5.5.7 Acciones que completan motivos

Destroy-obj: *"Frodo destroys the object Ring at Mount_Doom".*

Acquire-obj: *"Sauron has acquired the object Ring".*

Kill-strong: *"Aragorn kills Sauron at Mordor with AndurilSword".*

6. Experimentación

El análisis de los resultados es una de las partes más importantes de este proyecto. Entrelazaremos los resultados obtenidos en las pruebas con lo que nos marca la teoría, llegando así a comprender si verdaderamente se han alcanzado los objetivos deseados.

A continuación, describiremos el plan de pruebas que se va a seguir para su evaluación:

6.1 Plan de pruebas

Para analizar el correcto funcionamiento y confirmar que se han reunido los requisitos necesarios en este proyecto, se van a realizar una serie de pruebas que verifiquen su validez. Para ello, analizaremos los datos obtenidos y los compararemos con su resultado esperado.

Se van a generar 5 problemas con un estado inicial común para todos ellos. Variando sus objetivos, conseguiremos incrementar en cada uno cada vez más la dificultad a la que se enfrenta el planificador.

Ejecutaremos cada problema con 5 variaciones de aleatoriedad, y además, en cada uno de estos casos, generará 4 soluciones. De esta manera veremos cómo afecta la aleatoriedad a la generación del plan, y dentro de esta misma aleatoriedad, estudiar las diferencias entre las narraciones. Para variar la aleatoriedad de los planes, en cada ejecución iremos cambiando la *probability-ehc* (que toma valores entre 0 y 1). Esta aleatoriedad la marcaremos en un paréntesis al lado del problema en cuestión, por ejemplo: “Problema P1 (0.3)”. En este caso, habría un 0.3 (30%) de probabilidad de ir hacia la meta, y un 0.7 (70%) de elegir una acción aleatoria. Los textos de los resultados se pueden consultar en el [Anexo III](#) de este documento.

Nos serviremos de la **longitud**, **diversidad** e **interés** para compararlas.

- La longitud describe el número de acciones totales que tiene una narración.
- La diversidad compara las acciones del plan actual con el anterior, mostrando así el número total de acciones diferentes. Se calcula mediante el número total de acciones que aparecen en uno de los dos planes pero no en el otro, siendo únicamente acciones iguales las que se correspondan en todos los parámetros.
- La longitud y la diversidad vendrán dadas por el planificador, mientras que el interés tendremos que medirlo a través de una encuesta al público de unos problemas escogidos al azar.

En ocasiones, el plan no llega a generarse porque encuentra un punto muerto en la narración en la que es imposible alcanzar los objetivos. También se tendrá en cuenta para los resultados.

A continuación, describiremos en que consiste la encuesta para medir el interés, y procederemos a analizar los resultados obtenidos y cotejarlos con los de la encuesta, de esta manera evaluaremos cada problema de una forma completa.

Encuesta realizada al público

En esta encuesta, de los 25 textos recopilados en estas pruebas, se van a exponer al público 10 (dos de cada problema, uno poco aleatorio y otro muy aleatorio). Estos encuestados desconocerán el grado de aleatoriedad de los textos, tan sólo conocerán el estado inicial de los objetos y a qué objetivo debe llegar la narración. Juzgarán, evaluando cada texto entre 1 y 5 (siendo 1 una narración con muy poco interés y 5 una narración muy interesante). En cada problema analizaremos los correspondientes resultados obtenidos de esta encuesta.

Sorprendentemente, gracias a las redes sociales, como *Twitter* y *Youtube*, se han obtenido un total de **587** encuestas que evalúan estos 10 textos. Como podemos ver, se trata de un público bastante joven, casi en su totalidad hombres.

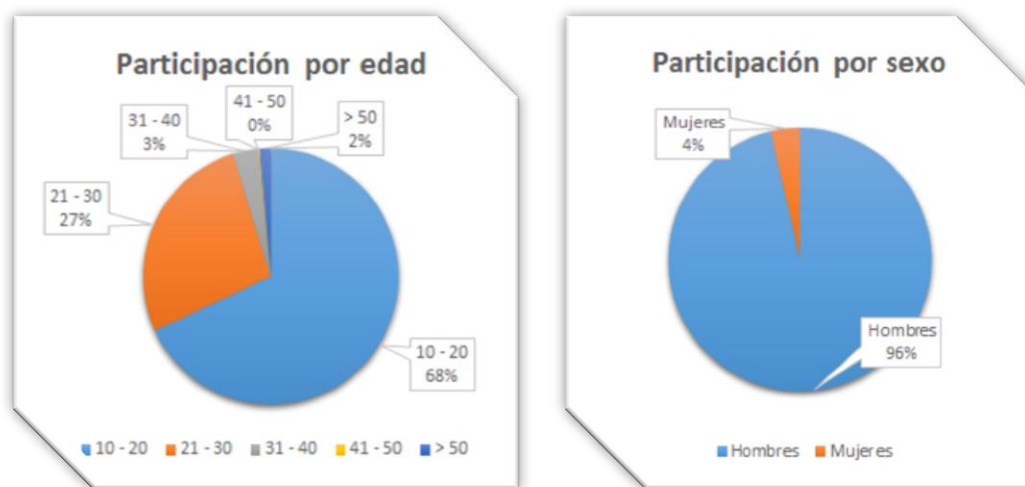


Ilustración 2 - Participación por edad y sexo

Esta encuesta se creó en *google drive*, en un formulario público. Dicho formulario constaba de 6 páginas, formado por una portada que explicaba en qué consistía el proyecto y cuál era el objetivo que se pretendía conseguir. En las siguientes 5 páginas se encontraban las narraciones generadas (a continuación en los problemas se detallarán cuales se han escogido en concreto para la encuesta). Cada una de esas páginas correspondía con cada uno de los problemas, donde se explicaba brevemente el objetivo que tenía que alcanzar la narración y se exponían las dos soluciones escogidas de ese problema (una más aleatoria que la otra siempre). Cada una se podía votar entre 1 y 5, siendo 1 una narración poco interesante y 5 una narración muy interesante. Para asegurarnos de que se evaluaran correctamente, en cada problema se han puesto de manera diferente los problemas. Es decir, si en una página de un problema la primera narración era más aleatoria que en la segunda, en todos no es así, se ha cambiado aleatoriamente en cada una. Los resultados obtenidos se han almacenado en un documento *Excel*, con el que se han creado los gráficos de cada problema en particular. Tanto la encuesta como los resultados de la misma pueden consultarse en el [Anexo IV](#) de este documento.

6.2 Creación de problemas

Como hemos mencionado anteriormente en el plan de pruebas, todos los problemas tendrán un estado inicial idéntico. Esto se ha decidido así ya que hace más sencilla la comparación entre los problemas, además del entendimiento de los mismos. Así pues, para variar la dificultad entre ellos, se presentarán distintos objetivos, cada vez más complejos, en los que se necesite de una narración más interesante.

Estado inicial de los problemas

Para el desarrollo de los problemas, nos basaremos en la temática de *El Señor de los Anillos*, de tal manera que crearemos sus personajes, sus relaciones de amistad, sus armas, lugares...

La lógica principal recae sobre los motivos y dos objetos principales en la historia (ver *ilustración 3*). Por un lado, tenemos los motivos genéricos *Peace* y *Conquer* con los que comenzarán la mayoría de los personajes. Uno simboliza el bien y otro el mal. A través de estos motivos, los personajes pueden aprender a querer completar otros motivos. Estos motivos están orientados hacia la destrucción o la adquisición de un objeto, el *Anillo* o la *Espada Anduril*:

- El motivo bueno *Destroy_ring* querrá conseguir el Anillo y destruirlo para que los que quieran adquirirlo (*Acquire_ring*) no puedan hacerlo.
- De igual manera, la Espada *Anduril* será la única arma declarada como *strong*, al igual que el personaje *Sauron*, por tanto será la única capaz de derrotarle. Los buenos querrán adquirirla (*Acquire_sword*) para matar a *Sauron*, y los malos querrán destruirla (*Destroy_sword*) para que esto no suceda.

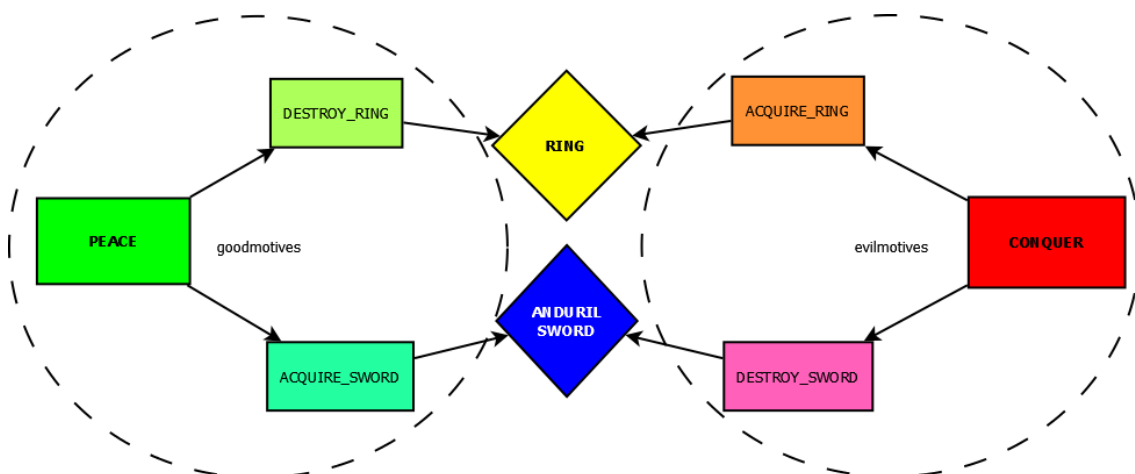


Ilustración 3 - Esquema de motivos y objetos

Una vez que los motivos quedan explicados, vamos a presentar los objetos que componen los problemas que vamos a emplear para probar el sistema.

(:objects

```
La_Comarca Rohan Rivendel Mordor Gondor Mount_Doom
Moria Bree Isengard - place
Bilbo Frodo Sam Merry Pippin - character ; Los
Hobbits
Aragorn Legolas Gimli Gandalf Boromir Elrond -
character ; Humanos, elfos y enanos
Saruman Sauron Gollum - character ; Los malos
Nazgul - character
Orc1 Orc2 Orc3 - character
Rabbit1 Rabbit2 - edible
Wolf1 Wolf2 - dangerous

Ring - usefobj
; Un anillo para gobernarlos a todos,
; un anillo para encontrarlos,
; un anillo para atraerlos a todos
; y atarlos en las tinieblas.

AndurilSword - arm
; La espada creada a partir de los trozos rotos
; de la espada Narsil, que corto el anillo
; del dedo de Sauron

GandalfStaff SarumanStaff ElfMedicine - usefobj
NazgulSword - arm
Orc1Sword Orc2Sword Orc3Sword - arm
AragornSword LegolasBow GimliAxe BoromirSword - arm
LittleElfSword - arm
Peace Destroy_ring Acquire_sword - goodmotive
Conquer Acquire_ring Destroy_sword - evilmotive
)
```

Para comenzar, declaramos los lugares más relevantes. A continuación creamos los *hobbits*, que serán declarados como débiles. Después creamos a los personajes con motivos buenos, no se hace distinción alguna entre humanos, elfos o enanos. Terminamos de crear los personajes con los que tendrán malos motivos, dos conejos comestibles y dos lobos peligrosos. Creamos los objetos principales, el Anillo y la Espada *Anduril*, los encargados de dirigir las motivaciones y crear el interés en los planes. Declaramos las armas y demás objetos mágicos, así como los motivos que explicamos anteriormente. Además, el dominio describirá las variables globales:

(:constants

```
Destroyed Acquired - status
Neutral - neutralmotive
```

)

Una vez obtenidos todos los objetos que participarán en los problemas (contando además con las variables globales del dominio), pasamos a describir los predicados iniciales que crearán el estado inicial:

; MOTIVOS

```
(want-motive Destroy_ring Ring Destroyed)
(want-motive Acquire_ring Ring Acquired)
(want-motive Acquire_sword AndurilSword Acquired)
(want-motive Destroy_sword AndurilSword Destroyed)

(implies-that Peace Destroy_ring)
; Para conseguir la paz se debe destruir el anillo
(implies-that Peace Acquire_sword)
; Para matar a Sauron se debe tener la espada
(implies-that Conquer Acquire_ring)
; Para dominar la tierra media, se debe conseguir el anillo
(implies-that Conquer Destroy_sword)
; Para evitar que maten a Sauron, deben destruirla
```

En primer lugar, definimos los *sub-goals* de los motivos, así como las relaciones entre ellos. Como podemos ver, aun estando en los problemas, podemos usar las variables constantes que se encuentran declaradas en el dominio.

; ANILLO

```
(magic-invisible Ring)
(can-be-destroyed-in Ring Mount_Doom)
```

Creamos las propiedades del anillo. Puede otorgar invisibilidad, y sólo podrá ser destruido en *Mount_Doom*. Aún no está definido su lugar, ya que lo aclararemos más adelante.

; ESPADA ANDURIL

```
(at AndurilSword Gondor)
(strong AndurilSword)
(can-be-destroyed-in AndurilSword Mordor)
```

Continuamos con la Espada *Anduril*, que la situaremos en *Gondor*. Como explicamos anteriormente, se declara como *strong* para ser la única arma capaz de derrotar a los personajes que sean también *strong*. Se ha establecido *Mordor* como el lugar capaz de destruirla, ya que es la base de los malvados.

; BILBO

```
(at Bilbo Rivendel)
(have-motive Bilbo Peace)
(wish-an-obj Bilbo Ring)
; Aunque es bueno, desea el anillo
(have-an-obj Bilbo LittleElfSword)
(weak Bilbo)
```

Bilbo está retirado en *Rivendel*. Como podemos ver, al tratarse de un personaje bueno tiene el motivo *Peace*. Aunque es bueno, desea el anillo ya que le tienta. A pesar de ser débil al ser un *hobbit*, dispone de una espada élfica.

; FRODO

```
(at Frodo La_Comarca)
(have-an-obj Frodo Ring)
(have-motive Frodo Peace)
(friends Bilbo Frodo)
(weak Frodo)
```

Frodo comienza en *La_Comarca*, con el anillo. Por supuesto, como a partir de ahora todos los personajes buenos, tiene como motivo *Peace*. Al ser un *hobbit* es débil, al igual que su tío *Bilbo*. Al ser familia son amigos.

; GANDALF

```
(at Gandalf La_Comarca)
(have-motive Gandalf Peace)
(have-motive Gandalf Destroy_ring)
(have-an-obj Gandalf GandalfStaff)
(magic-kill GandalfStaff)
(can-use-magic-kill Gandalf)
(can-use-magic-heal Gandalf)
(friends Gandalf Bilbo)
(friends Gandalf Frodo)
```

Gandalf es un mago, un viejo amigo de *Frodo* y *Bilbo*. También empieza en *La_Comarca*. Además de *Peace*, también tiene el motivo *Destroy_ring*, ya que sabe lo que es y lo que implica. Al ser un mago, domina todas las magias, tanto de daño como sanación. Sin embargo, su bastón sólo sirve para lo primero.

; SAM

```
(at Sam La_Comarca)
(have-motive Sam Peace)
(friends Sam Frodo)
(weak Sam)
```

Sam es el fiel jardinero de *Frodo*. Es un personaje muy simple.

; MERRY Y PIPPIN

```
(at Merry La_Comarca)
(have-motive Merry Peace)
(at Pippin La_Comarca)
(have-motive Pippin Peace)
(weak Merry)
(weak Pippin)
(friends Merry Pippin)
(friends Merry Frodo) ; Son primos
(friends Pippin Frodo) ; Son primos
```

Merry y *Pippin* son los últimos *hobbits* que aparecerán en estos problemas. Ambos están en *La_Comarca* y son buenos. Son amigos entre sí y de *Frodo* porque son parientes.

; GOLLUM

```
(at Gollum Mordor) ; Le estan torturando
(have-motive Gollum Peace)
(have-motive Gollum Acquire_ring)
(enemies Gollum Bilbo)
(enemies Gollum Sauron)
(wish-an-obj Gollum Ring)
(weak Gollum)
```

Gollum es sin duda el personaje más interesante. Es el único que tiene un motivo bueno y uno malo. La gracia de esto es la bipolaridad, como se muestra en los libros y las películas. A veces quiere ayudar a *Frodo*, otras matarlo, todo con el fin de conseguir el anillo. Cambiará el estado de *friends* a *enemies* y viceversa a placer. Conoce a *Bilbo* y le odia por robarle el anillo. Odia a *Sauron* por torturarlo, de ahí que esté en *Mordor*.

; SAURON

```
(at Sauron Mordor)
(isNPC Sauron)
; No puede moverse, no es un personaje en si
(have-motive Sauron Conquer)
(have-motive Sauron Acquire_ring)
(wish-an-obj Sauron Ring)
(strong Sauron)
```

Sauron es el malo principal. No es un personaje en sí, así que lo declaramos como *NPC* para que no pueda moverse. Por supuesto, quiere conquistar y adquirir el anillo. Como comentamos anteriormente, *Sauron* es el único personaje definido como *strong*.

; NAZGUL

```
(at Nazgul Mordor)
(have-motive Nazgul Conquer)
(have-an-obj Nazgul NazgulSword)
(friends Nazgul Sauron)
(wish-an-obj Nazgul Ring)
```

El *Nazgul* es un espectro con forma física cuya única proposición en la vida es conseguir el anillo. Está al servicio de *Sauron*, por lo que se establece una relación de amistad entre ellos.

; SARUMAN

```
(at Saruman Isengard)
(isNPC Saruman)
; Lo dejaremos como NPC ya que
; nunca saldra de Isengard
(have-motive Saruman Conquer)
(have-an-obj Saruman SarumanStaff)
(magic-kill SarumanStaff)
(can-use-magic-kill Saruman)
(can-use-magic-heal Saruman)
(friends Saruman Sauron)
(friends Saruman Gandalf)
```

Saruman es el mago blanco, que reside en la torre de *Isengard*. Posee los mismos poderes y bastón que *Gandalf*, y aunque son amigos, también está aliado en secreto con *Sauron*, por lo que su motivo principal es malo.

; ARAGORN

```
(at Aragorn Bree)
(have-motive Aragorn Peace)
(have-an-obj Aragorn AragornSword)
(friends Gandalf Aragorn)
```

Aragorn está en la posada de *Bree*. Es un personaje bueno que es amigo de *Gandalf*. En realidad tiene más amistades, pero se describirán a la vez que se declaran el resto de personajes.

; ORCOS

```
(at Orc1 Isengard)
(have-motive Orc1 Conquer)
(have-an-obj Orc1 Orc1Sword)
(friends Orc1 Saruman)
```

```
(at Orc2 Isengard)
(have-motive Orc2 Conquer)
(have-an-obj Orc2 Orc2Sword)
(friends Orc2 Saruman)
```

```
(at Orc3 Isengard)
(have-motive Orc3 Conquer)
(have-an-obj Orc3 Orc3Sword)
(friends Orc3 Saruman)
```

```
(friends Orc1 Orc2)
(friends Orc1 Orc3)
(friends Orc2 Orc3)
```

Hay tres orcos al servicio de *Saruman*, cada uno tiene su arma y son amigos entre sí y de *Saruman*.

; ELROND

```
(at Elrond Rivendel)
(have-motive Elrond Peace)
(can-use-magic-heal Elrond)
(have-an-obj Elrond ElfMedicine)
(magic-heal ElfMedicine)
(friends Elrond Gandalf)
(friends Elrond Aragorn)
```

El elfo *Elrond* está en *Rivendel*, ciudad de los elfos. Dispone de medicina y de los conocimientos de usarla. Es amigo de *Gandalf* y *Aragorn*.

; **LEGOLAS**

(at Legolas Rivendel)
(have-motive Legolas Peace)
(have-an-obj Legolas LegolasBow)
(friends Elrond Legolas)
(friends Legolas Aragorn)

; **GIMLI**

(at Gimli Rivendel)
(have-motive Gimli Peace)
(have-an-obj Gimli GimliAxe)
(friends Legolas Gimli)
(friends Gimli Aragorn)

; **BOROMIR**

(at Boromir Rivendel)
(have-motive Boromir Peace)
(have-an-obj Boromir BoromirSword)
(wish-an-obj Boromir Ring)
; Aunque es bueno, desea el anillo

Léglas, *Gimli* y *Boromir* (elfo, enano y humano respectivamente) están en *Rivendel*, cada uno tiene su arma y todos tienen el motivo *Peace* genérico. *Boromir* se siente tentado por el anillo, por lo que lo desea. *Léglas*, *Gimli* y *Aragorn* son amigos entre sí, mientras que a *Boromir* no se le ha asignado ninguna amistad.

; **ANIMALES**

(at Rabbit1 Rohan)
(at Rabbit2 Rohan)
(at Wolf1 Rohan)
(at Wolf2 Gondor)

Terminamos con los animales, situaremos los dos conejos en *Rohan*, al igual que un lobo, y el otro lobo estará en *Gondor*, donde se encuentra la Espada *Anduril*.

Con esto terminamos el estado inicial de los siguientes cinco problemas. Sus objetivos vendrán definidos cada uno en su apartado correspondiente, explicando la finalidad que se pretende conseguir y como aumenta la dificultad con respecto al anterior.

6.3 Problema P1

6.3.1 Objetivo del problema

Éste es el problema más simple. Tan sólo busca la resolución directa de los motivos:

```
(:goal (and
      (is Ring Destroyed)
      (dead Sauron)
    )
)
```

Para completar el problema, la narración debe conseguir que el Anillo sea destruido, da igual quien lo haga, y que *Sauron* muera, de igual forma, sin importar quien lo mate. Para ello se deberán emplear los motivos *Destroy_ring* y *Acquire_sword*, los dos motivos específicos buenos.

Al ser un problema tan básico, apenas se espera de él enfrentamiento. Se espera conseguir que varios personajes buenos sigan y aprendan de sus motivos para completar la narración.

A continuación, veremos los resultados obtenidos, los planes generados pueden verse en el [Anexo III](#) de este documento.

6.3.2 Resultados obtenidos

Este es el plan más interesante, obtenido con la aleatoriedad más baja:

```
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN kills SAURON at MORDOR with ANDURILSWORD
```

A continuación, podemos analizar todos los resultados obtenidos de este problema, generando 4 planes para cada tipo de aleatoriedad:

ALEATORIEDAD MUY ALTA (0.1)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	11	11	10	10	10,50
Diversidad		4	7	4	5,00

Tabla 3 - Resultados P1 (0.1)

ALEATORIEDAD ALTA (0.3)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	11	10	11	11	10,75
Diversidad		5	7	4	5,33

Tabla 4 - Resultados P1 (0.3)

ALEATORIEDAD MEDIA (0.5)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	10	11	10	10	10,25
Diversidad		5	5	4	4,67

Tabla 5 - Resultados P1 (0.5)

ALEATORIEDAD BAJA (0.7)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	12	11	10	11	11,00
Diversidad		4	5	5	4,67

Tabla 6 - Resultados P1 (0.7)

ALEATORIEDAD MUY BAJA (0.9)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	10	10	10	10	10,00
Diversidad		2	0	0	0,67

Tabla 7 - Resultados P1 (0.9)

Como podemos ver en los resultados, la aleatoriedad no ha marcado un cambio significativo en la longitud de los planes. En todos los casos se han generado sin problemas, y podemos ver que la diversidad sí sufre una bajada considerable en la aleatoriedad muy baja, siendo en ocasiones 0, lo que implica que son exactamente iguales.

6.3.3 Resultados de la encuesta

En estos gráficos analizaremos el interés que ha despertado en el público este problema, siendo el eje horizontal la puntuación valorada y el vertical el recuento de dichos puntos.

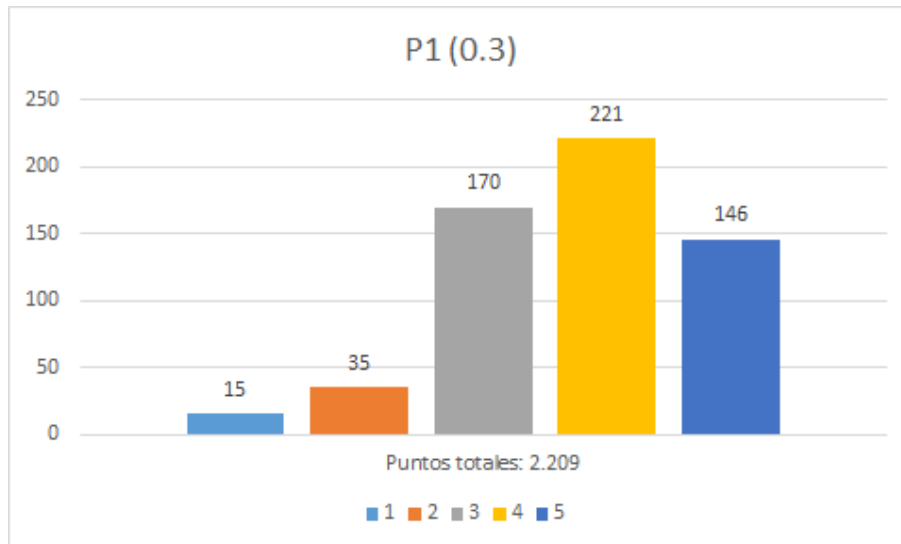


Ilustración 4 - Encuesta P1 (0.3)

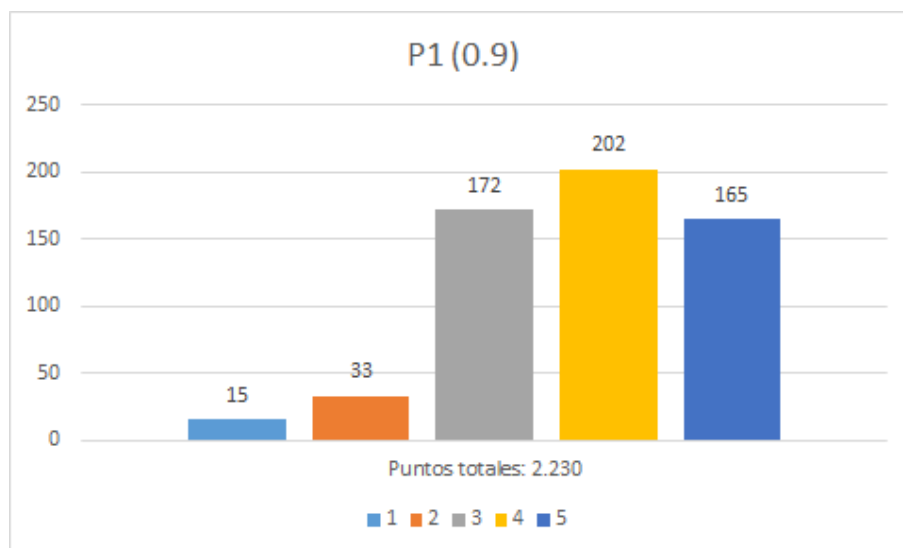


Ilustración 5 - Encuesta P1 (0.9)

La encuesta se ha realizado sobre el caso P1 (0.3) y P1 (0.9) elegidos al azar. Son unos resultados muy parecidos, y teniendo en cuenta que las narraciones son prácticamente iguales, cambiando en algunos casos el orden de las acciones, han tenido razón. Si nos fijamos en los puntos totales, calculados como el sumatorio de las votaciones, por un pequeño margen ha ganado la P1 (0.9), lo que indica que ésta ha resultado más interesante.

6.4 Problema P2

6.4.1 Objetivo del problema

En el siguiente problema, para terminar de comprobar la correcta funcionalidad de los motivos, nos ponemos de parte de los malos.

```
(:goal (and
      (dead Boromir)
      (dead Gandalf)
      (is AndurilSword Destroyed)
      (motive-complete Sauron Acquire_ring)
    )
)
```

Además, para complicarlo más que el caso anterior, *Gandalf* y *Boromir* deben morir. Los malos esta vez cobrarán vida para destruir la Espada *Anduril* y conseguir el *Anillo* para *Sauron* (ya que él no puede moverse).

A diferencia del caso anterior, ya intervienen tanto personajes con motivaciones buenas como malas, de tal manera que habrá lucha obligatoriamente o no podría haber muertos.

A continuación, veremos los resultados obtenidos, los planes generados pueden verse en el [Anexo III](#) de este documento.

6.4.2 Resultados obtenidos

Este es el plan más interesante, obtenido con la aleatoriedad más alta:

```
GANDALF (DESTROY_RING) and GOLLUM (ACQUIRE_RING) are enemies because
their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their
motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their
motives disagree
GANDALF travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SAURON and GOLLUM are friends because they have the motive
ACQUIRE_RING in common
GANDALF wishes the object RING because he has the motive DESTROY_RING
FRODO (PEACE) and SARUMAN (CONQUER) are enemies because their motives
disagree
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL has the motive DESTROY_SWORD because he wants to complete his
motive CONQUER
FRODO travels to ISENGARD because he wants to give the object RING to
GANDALF
FRODO gives the object RING to GANDALF at ISENGARD
NAZGUL wishes the object ANDURILSWORD because he has the motive
DESTROY_SWORD
```



SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
GOLLUM loots the object RING from GANDALF (dead) at ISENGARD
GOLLUM (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GOLLUM travels to MORDOR because he wants to give the object RING to SAURON
GOLLUM (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
NAZGUL takes the object ANDURILSWORD at GONDOR
GOLLUM (PEACE) and NAZGUL (DESTROY_SWORD) are enemies because their motives disagree
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
SAURON punches and hurts GOLLUM at MORDOR
NAZGUL destroys the object ANDURILSWORD at MORDOR
NAZGUL kills GOLLUM at MORDOR with NAZGULSWORD
SAURON loots the object RING from GOLLUM (dead) at MORDOR
SAURON has acquired the object RING

A continuación, podemos analizar todos los resultados obtenidos de este problema, generando 4 planes para cada tipo de aleatoriedad:

ALEATORIEDAD MUY ALTA (0.1)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	28	20	28	28	26,00
Diversidad		30	26	22	26,00

Tabla 8 - Resultados P2 (0.1)

ALEATORIEDAD ALTA (0.3)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	31	24	21	31	26,75
Diversidad		32	13	30	25,00

Tabla 9 - Resultados P2 (0.3)

ALEATORIEDAD MEDIA (0.5)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	30	25	21	31	26,75
Diversidad		19	18	32	23,00

Tabla 10 - Resultados P2 (0.5)

ALEATORIEDAD BAJA (0.7)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	35	30	23	35	30,75
Diversidad		31	27	40	32,67

Tabla 11 - Resultados P2 (0.7)

ALEATORIEDAD MUY BAJA (0.9)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	29	25	21	29	26,00
Diversidad		18	16	28	20,67

Tabla 12 - Resultados P2 (0.9)

El primer cambio que podemos apreciar es la longitud de la narración con respecto al problema anterior (se ha pasado de 10 a 26-30 líneas). El plan es más complejo, y podemos ver cómo la aleatoriedad modifica débilmente la diversidad cuando la vamos bajando. Salvo en el caso de (0.7) que debido a la creación de narraciones más extensas es mayor. Aunque su probabilidad de escoger acciones aleatoriamente es menor, pueden suceder este tipo de anomalías, ya que no deja de ser aleatorio. En cuanto a la diversidad, podemos apreciar que hay cifras que superan a la longitud del propio plan, ¿cómo es esto posible? La diversidad se calcula en función de las acciones escogidas en dicho plan y el anterior. Si muy pocas o ninguna de las acciones del plan actual no se han usado en el anterior, ya tenemos casi como mínimo la longitud del plan. A esto hay que añadir las que son diferentes del anterior, las que están en el anterior pero no en el actual. De nuevo, todos los planes se han generado sin incidentes.

6.4.3 Resultados de la encuesta

En estos gráficos analizaremos el interés que ha despertado en el público este problema, siendo el eje horizontal la puntuación valorada y el vertical el recuento de dichos puntos.

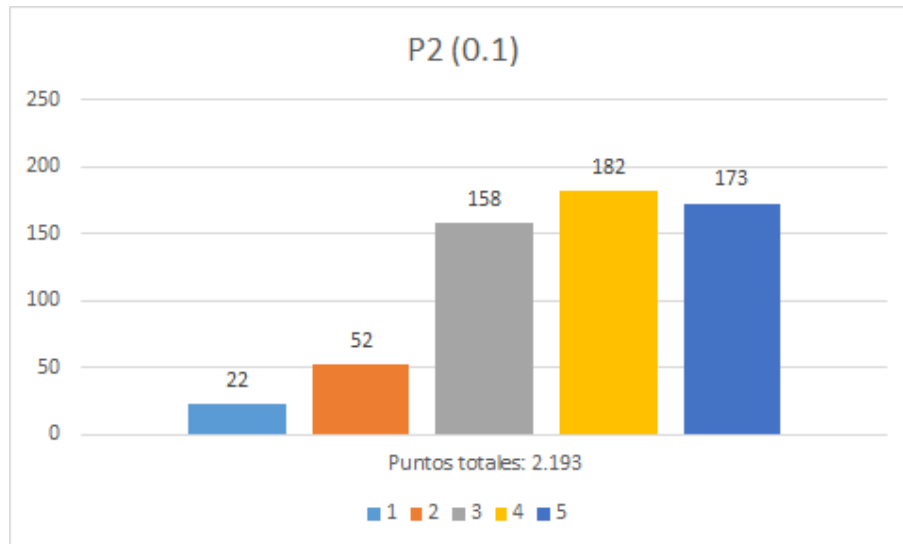


Ilustración 6 - Encuesta P2 (0.1)

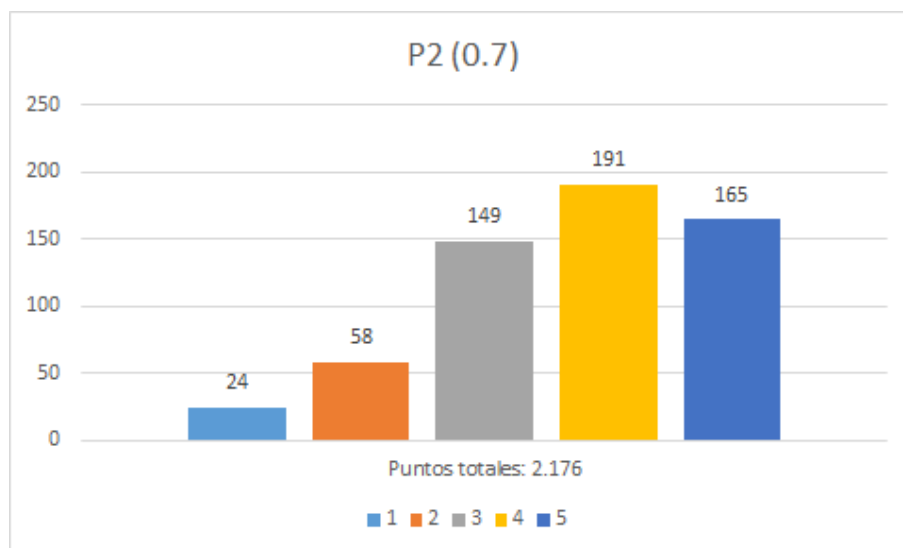


Ilustración 7 - Encuesta P2 (0.7)

Para esta encuesta se han escogido al azar las narraciones P2 (0.1) y P2 (0.7). Esta vez, aunque también son muy similares, la prueba más aleatoria ha gustado más. Analizando estos resultados con los del problema anterior, vemos que ambos muestran un interés bastante elevado.

6.5 Problema P3

6.5.1 Objetivo del problema

Con los problemas P1 y P2 hemos comprobado la funcionalidad de los motivos. Seguiremos poniéndolos como objetivo en este problema y los siguientes, pero iremos variándolos y siempre añadiendo más dificultad.

```
(:goal (and
  (dead Boromir)
  (dead Nazgul)
  (dead Saruman)
  (enemies Frodo Sam)
  (friends Gollum Aragorn)
  (is AndurilSword Destroyed)
  (is Ring Destroyed)
)
```

En honor al actor Sean Bean, que interpreta a *Boromir* en la trilogía de *El Señor de los Anillos* y famoso por morir en más de 20 ocasiones a lo largo de su carrera, incluyendo esta saga, vamos a marcar como objetivo su muerte en todos los problemas, excepto el primero que no muere nadie. Además de él, también deberán morir el *Nazgul* y *Saruman*. Recordemos que *Saruman* es un *NPC* y por tanto no puede moverse, así que han de ir a donde está a matarle. Para probar las mecánicas de amistad, vamos a establecer que *Frodo* y *Sam* acaben siendo enemigos, y que *Gollum* y *Aragorn* sean amigos. Por último, en cuanto a los motivos principales, tanto la Espada *Anduril* como el *Anillo* deben ser destruidos, de tal manera que cada bando consigue un objetivo.

A continuación, veremos los resultados obtenidos, los planes generados pueden verse en el [Anexo III](#) de este documento.

6.5.2 Resultados obtenidos

Este es el plan más interesante, obtenido con la aleatoriedad media:

```
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives
disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their
motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their
motives disagree
GANDALF (PEACE) and NAZGUL (CONQUER) are enemies because their motives
disagree
FRODO travels to MOUNT_DOOM because he wishes to destroy the object
RING
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their
motives disagree
```




NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL travels to LA_COMARCA because he wants to kill GANDALF
GOLLUM (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
SAM and NAZGUL are friends because they want to kill GOLLUM
NAZGUL asks his friend SAM to follow his motive CONQUER
FRODO (PEACE) and SAM (CONQUER) are enemies because their motives disagree
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
WOLF2 hurts NAZGUL at GONDOR
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
BOROMIR punches and hurts SARUMAN at ISENGARD
FRODO destroys the object RING at MOUNT_DOOM
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
ARAGORN (PEACE) and NAZGUL (DESTROY_SWORD) are enemies because their motives disagree
ARAGORN kills NAZGUL at MORDOR with ARAGORNSWORD
SAURON loots the object ANDURILSWORD from NAZGUL (dead) at MORDOR
SAURON has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
SAURON destroys the object ANDURILSWORD at MORDOR
GANDALF travels to ISENGARD because he wants to kill SARUMAN
GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF

A continuación, podemos analizar todos los resultados obtenidos de este problema, generando 4 planes para cada tipo de aleatoriedad:

ALEATORIEDAD MUY ALTA (0.1)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	34	38	37	22	32,75
Diversidad		63	61	53	59,00

Tabla 13 - Resultados P3 (0.1)

ALEATORIEDAD ALTA (0.3)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	30	33	33	36	33,00
Diversidad		39	16	47	34,00

Tabla 14 - Resultados P3 (0.3)

ALEATORIEDAD MEDIA (0.5)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	34	35	31	32	33,00
Diversidad		33	40	31	34,67

Tabla 15 - Resultados P3 (0.5)

ALEATORIEDAD BAJA (0.7)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	31	28	27	25	27,75
Diversidad		41	33	18	30,67

Tabla 16 - Resultados P3 (0.7)

ALEATORIEDAD MUY BAJA (0.9)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	26	26	28	29	27,25
Diversidad		8	18	25	17,00

Tabla 17 - Resultados P3 (0.9)

Con estos cambios de aleatoriedad, en este problema podemos apreciar muy fácilmente que la longitud desciende a la vez que ella, al igual que la diversidad. Los planes, en comparación con el problema anterior, son similares con respecto a longitud, pero son mucho más diversos. De nuevo, no se ha producido ningún percance por el cual no se haya generado ningún plan.

6.5.3 Resultados de la encuesta

En estos gráficos analizaremos el interés que ha despertado en el público este problema, siendo el eje horizontal la puntuación valorada y el vertical el recuento de dichos puntos.

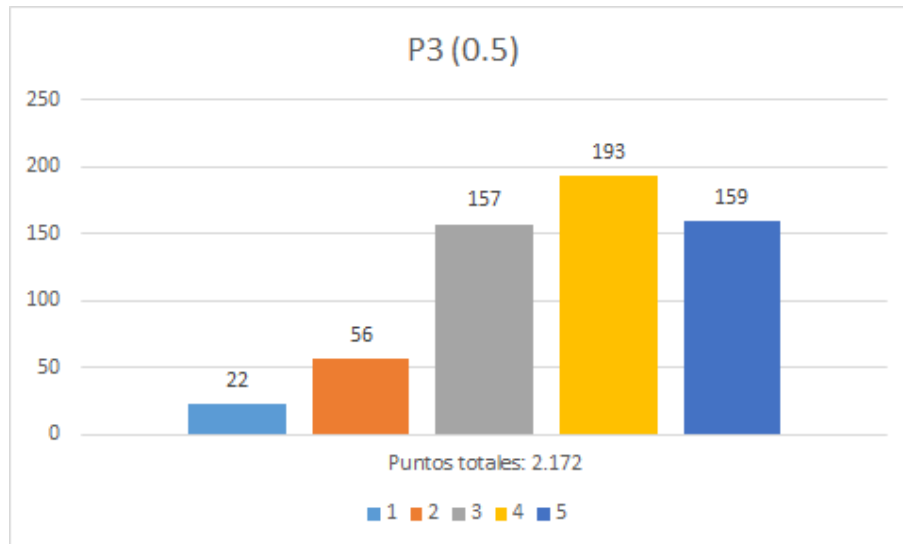


Ilustración 8 - Encuesta P3 (0.5)

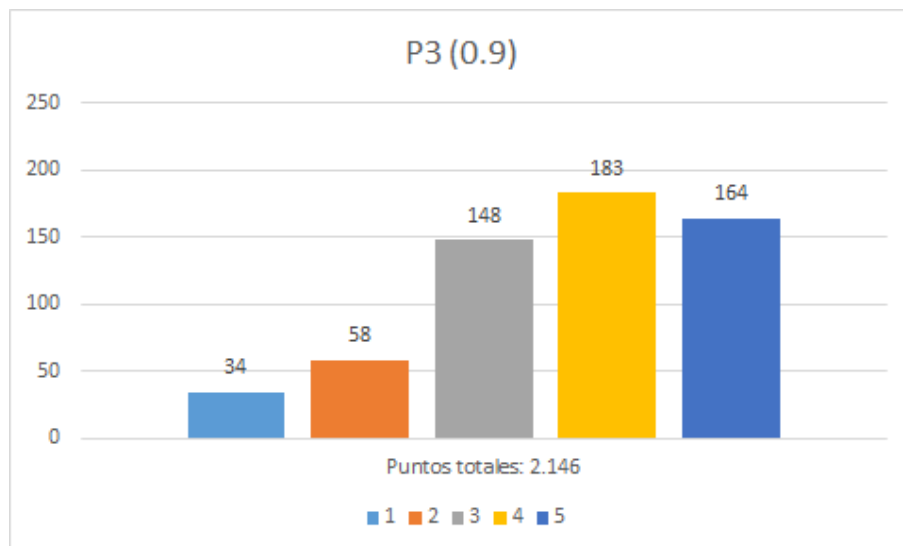


Ilustración 9 - Encuesta P3 (0.9)

Los resultados son muy similares al problema anterior. Para este caso, se han escogido el P3 (0.5) y el P3 (0.9). De nuevo, aunque muy parecidos, el caso más aleatorio ha resultado más interesante al público.

6.6 Problema P4

6.6.1 Objetivo del problema

Para complicar aún más este problema, se van a añadir más muertes, y más específicas:

```
(:goal (and
  (dead Boromir)
  (dead Orc1)
  (dead Orc2)
  (dead Rabbit1)
  (enemies Frodo Sam)
  (friends Gollum Aragorn)
  (killed-by Saruman Legolas)
  (killed-by Orc3 Frodo)
  (is Ring Acquired)
  (dead Sauron)
)
```

En este problema, además de *Boromir*, deben morir los 3 orcos. El tercero además, siendo asesinado por *Frodo*, al igual que *Saruman*, que debe matarlo *Légolas*. Una dificultad añadida es que *Frodo* no tiene ningún arma, y además, es débil, por tanto debe buscar la manera de matar al orco. Para confirmar que alguien mata algún animal para recuperarse, un conejo también debe morir. En cuanto a los motivos principales, *Sauron* debe morir y el *Anillo* debe ser adquirido. Pero en ningún momento se especifica que sea *Sauron* quien lo posea, cualquier personaje que tenga el motivo adecuado puede hacerlo.

A continuación, veremos los resultados obtenidos, los planes generados pueden verse en el [Anexo III](#) de este documento.

6.6.2 Resultados obtenidos

Este es el plan más interesante, obtenido con la aleatoriedad más alta:

```
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
FRODO (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
```



FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
GANDALF (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF punches and hurts ORC2 at LA_COMARCA
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
FRODO gives the object RING to GANDALF at LA_COMARCA
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
BOROMIR hurts SARUMAN at ISENGARD with BOROMIRSWORD
ORC3 and BOROMIR are enemies because SARUMAN (ORC3's friend) and BOROMIR are enemies
GOLLUM travels to LA_COMARCA because he wishes the object RING GANDALF has
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
ARAGORN (ACQUIRE_SWORD) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
BOROMIR hurts ORC3 at ISENGARD with BOROMIRSWORD
FRODO and GOLLUM are friends because they have the motive PEACE in common
GANDALF gives the object RING to FRODO at LA_COMARCA
GOLLUM asks his friend FRODO to follow his motive ACQUIRE_RING
ORC3 travels to ROHAN because he wants to eat a RABBIT2
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are enemies
ORC3 (hurt) kills and eats RABBIT1 at ROHAN with ORC3SWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO loots the object ORC2SWORD from ORC2 (dead) at LA_COMARCA
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
FRODO has acquired the object RING
WOLF1 hurts ORC3 at ROHAN
ARAGORN travels to LA_COMARCA because he wants to kill GOLLUM
ARAGORN and GOLLUM are friends because they want to kill SAURON
ORC3 travels to LA_COMARCA because he wants to kill FRODO
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO kills ORC3 at LA_COMARCA with ORC2SWORD
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

A continuación, podemos analizar todos los resultados obtenidos de este problema, generando 4 planes para cada tipo de aleatoriedad:

ALEATORIEDAD MUY ALTA (0.1)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	50	44	48	52	48,50
Diversidad		74	64	76	71,33

Tabla 18 - Resultados P4 (0.1)

ALEATORIEDAD ALTA (0.3)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	45	50	51	42	47,00
Diversidad		81	81	59	73,67

Tabla 19 - Resultados P4 (0.3)

ALEATORIEDAD MEDIA (0.5)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	49	46	42	40	44,25
Diversidad		59	50	52	53,67

Tabla 20 - Resultados P4 (0.5)

ALEATORIEDAD BAJA (0.7)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	45	45	47	48	46,25
Diversidad		46	60	55	53,67

Tabla 21- Resultados P4 (0.7)

ALEATORIEDAD MUY BAJA (0.9)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	43	44	43	52	45,50
Diversidad		39	35	47	40,33

Tabla 22 - Resultados P4 (0.9)

La aleatoriedad de nuevo se hace notar. Según va descendiendo, la diversidad de los planes baja también. Estas diversidades tan elevadas, como las de los primeros casos, nos muestran que entre sí no tienen casi nada que ver. Sin embargo, la longitud permanece bastante constante. Nuevamente, todos los planes han llegado al estado final sin problemas.

6.6.3 Resultados de la encuesta

En estos gráficos analizaremos el interés que ha despertado en el público este problema, siendo el eje horizontal la puntuación valorada y el vertical el recuento de dichos puntos.

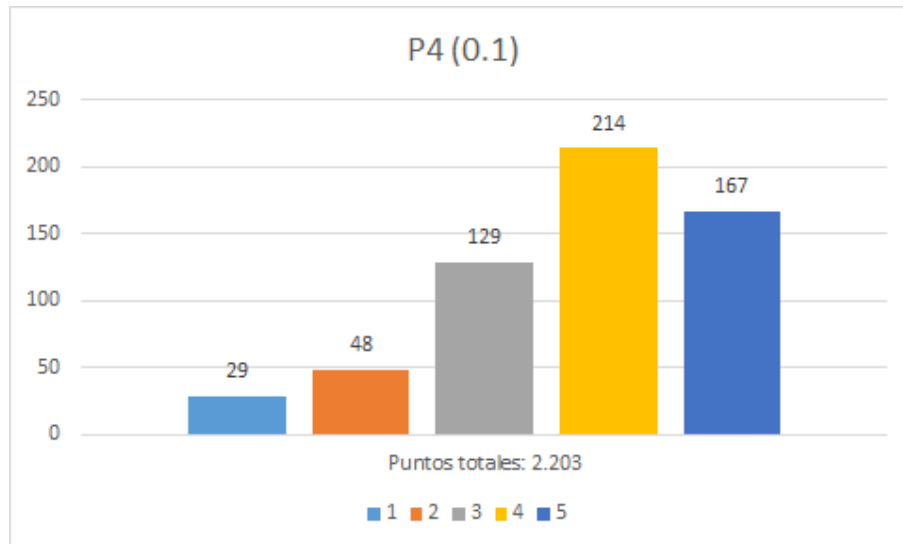


Ilustración 10 - Encuesta P4 (0.1)

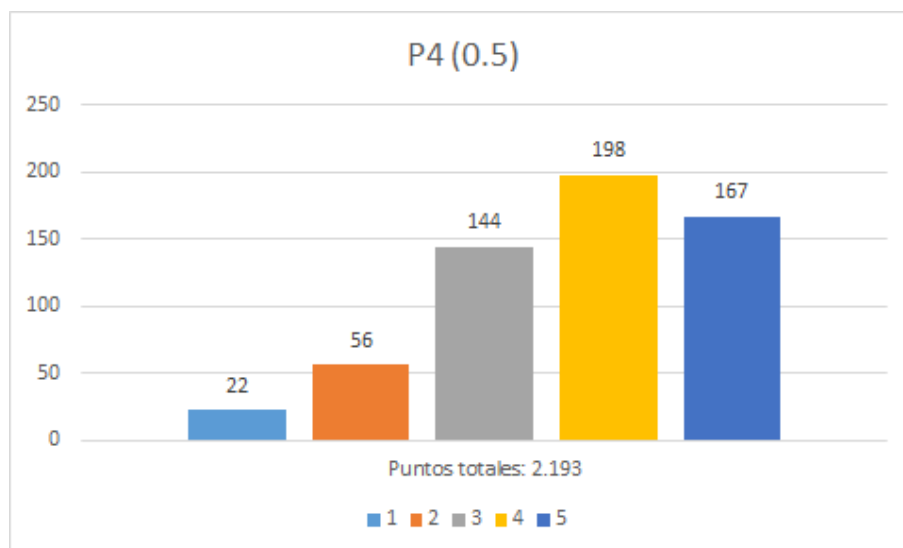


Ilustración 11 - Encuesta P4 (0.5)

Para este caso, se han empleado los resultados de P4 (0.1) y P4 (0.5). Las encuestas siguen mostrando un apoyo del público, entorno al 4, a su interés en ambas narraciones. Casi rozando el empate técnico, la narración más aleatoria vuelve a superar a la menos aleatoria.

6.7 Problema P5

6.7.1 Objetivo del problema

Llegamos al último problema y por tanto, última tanda de pruebas.

```
(:goal (and
  (dead Boromir)
  (have-an-obj Frodo LittleElfSword)
  (dead Orc1)
  (dead Orc2)
  (dead Orc3)
  (killed-by Nazgul Aragorn)
  (killed-by Saruman Legolas)
  (hurt Frodo)
  (hurt Gimli)
  (have-an-obj Sam AragornSword)
  (dead Gollum)
  (dead Rabbit1)
  (dead Wolf2)
  (is Ring Destroyed)
  (dead Sauron)
)
```

Para el estado final de este problema, se pide como novedad que algunos personajes acaben heridos y que tengan algunos objetos en concreto. Tanto *Boromir* como los tres orcos deben morir, así como el *Nazgul* a manos de *Aragorn* y *Saruman* asesinado por *Légolas*. También deberán estar muertos *Gollum*, un conejo y un lobo. *Frodo* y *Gimli* deberán acabar heridos, y en cuanto a los motivos, los buenos deben cumplir sus objetivos, el *Anillo* destruido y *Sauron* muerto.

A continuación, veremos los resultados obtenidos, los planes generados pueden verse en el [Anexo III](#) de este documento.

6.7.2 Resultados obtenidos

Este es el plan más interesante, obtenido con la aleatoriedad más alta:

```
ARAGORN (PEACE) and SARUMAN (CONQUER) are enemies because their
motives disagree
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives
disagree
GANDALF (DESTROY_RING) and ORC3 (CONQUER) are enemies because their
motives disagree
ORC2 and GANDALF are enemies because ORC3 (ORC2's friend) and GANDALF
are enemies
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
```




GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
GIMLI (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
FRODO (DESTROY_RING) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
SAM and GANDALF are friends because they have the motive PEACE in common
NAZGUL travels to RIVENDEL because he wants to kill GIMLI
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
GIMLI punches and hurts NAZGUL at RIVENDEL
NAZGUL punches and hurts GIMLI at RIVENDEL
GIMLI travels to ROHAN because he wants to eat a RABBIT2
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
FRODO travels to RIVENDEL because he wants to give the object RING to BILBO
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
NAZGUL punches and hurts FRODO at RIVENDEL
BOROMIR (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
BILBO travels to MOUNT_DOOM because he wishes the object RING FRODO has
GANDALF travels to MORDOR because he wants to kill GOLLUM
NAZGUL punches and hurts BOROMIR at RIVENDEL
ARAGORN travels to RIVENDEL because he wants to kill NAZGUL
GIMLI has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
GIMLI (hurt) kills and eats RABBIT1 at ROHAN with GIMLIAXE
SAM has the motive DESTROY_RING because he wants to complete his motive PEACE
GANDALF kills (magic) GOLLUM at MORDOR with GANDALFSTAFF
FRODO destroys the object RING at MOUNT_DOOM
WOLF1 hurts GIMLI at ROHAN
SAM (DESTROY_RING) and ORC3 (CONQUER) are enemies because their motives disagree
ARAGORN travels to ISENGARD because he wants to kill SARUMAN
ARAGORN (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ORC3 travels to LA_COMARCA because he wants to kill SAM



GIMLI travels to MORDOR because he wants to kill SAURON
ARAGORN travels to RIVENDEL because he wants to kill NAZGUL
ARAGORN travels to LA_COMARCA because he wants to kill ORC3
SAM and ARAGORN are friends because they want to kill ORC3
NAZGUL kills BOROMIR at RIVENDEL with NAZGULSWORD
LEGOLAS punches and hurts SARUMAN at ISENGARD
ARAGORN hurts ORC3 at LA_COMARCA with ARAGORNSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
GIMLI wishes the object ANDURILSWORD because he has the motive
ACQUIRE_SWORD
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
SAM kills ORC3 at LA_COMARCA with ARAGORNSWORD
NAZGUL travels to LA_COMARCA because he wants to kill ARAGORN
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
GIMLI travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
GIMLI takes the object ANDURILSWORD at GONDOR
GIMLI kills WOLF2 at GONDOR with ANDURILSWORD
GIMLI travels to MORDOR because he wants to kill SAURON
GIMLI kills SAURON at MORDOR with ANDURILSWORD
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
ARAGORN kills NAZGUL at LA_COMARCA with ARAGORNSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA

A continuación, podemos analizar todos los resultados obtenidos de este problema, generando 4 planes para cada tipo de aleatoriedad:

ALEATORIEDAD MUY ALTA (0.1)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	65	84	73	67	72,25
Diversidad		111	125	83	106,33

Tabla 23 - Resultados P5 (0.1)

ALEATORIEDAD ALTA (0.3)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	66	64	65	60	63,75
Diversidad		97	94	83	91,33

Tabla 24 - Resultados P5 (0.3)

ALEATORIEDAD MEDIA (0.5)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	58	59	59	66	60,50
Diversidad		89	86	77	84,00

Tabla 25 - Resultados P5 (0.5)

ALEATORIEDAD BAJA (0.7)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	59	54	56	61	57,50
Diversidad		55	78	93	75,33

Tabla 26 - Resultados P5 (0.7)

ALEATORIEDAD MUY BAJA (0.9)	NARRACIONES				\bar{x}
	1	2	3	4	
Genera plan	Sí	Sí	Sí	Sí	
Longitud	60	67	58	58	60,75
Diversidad		41	63	26	43,33

Tabla 27 - Resultados P5 (0.9)

La diversidad se nota mucho más que en los casos anteriores que varía según la aleatoriedad. La longitud, sin embargo, permanece muy similar. Como se ha pedido que en los objetivos les sucedan más cosas a más personajes, es decir, han intervenido personajes que a veces no han aparecido, la longitud con respecto a los planes anteriores ha crecido considerablemente. Por último, todos los planes se han generado correctamente.

6.7.3 Resultados de la encuesta

En estos gráficos analizaremos el interés que ha despertado en el público este problema, siendo el eje horizontal la puntuación valorada y el vertical el recuento de dichos puntos.

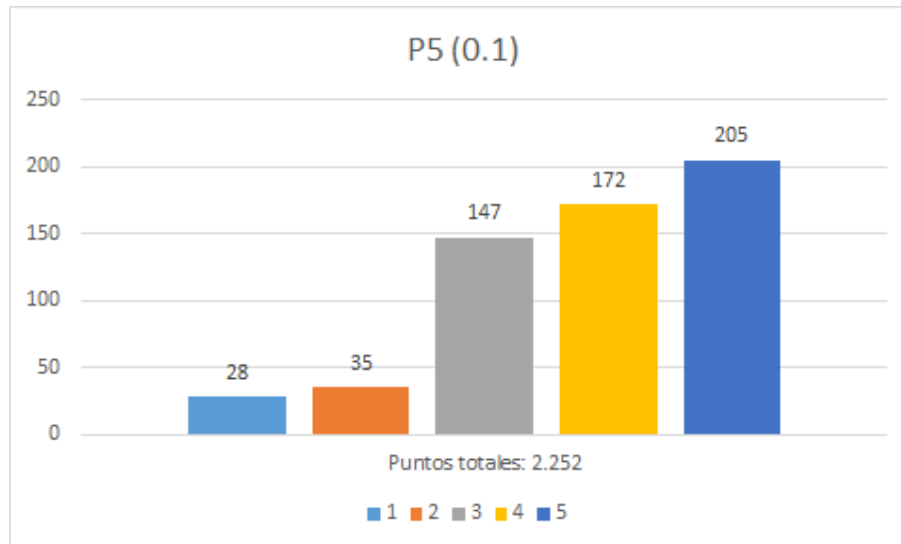


Ilustración 12 - Encuesta P5 (0.1)

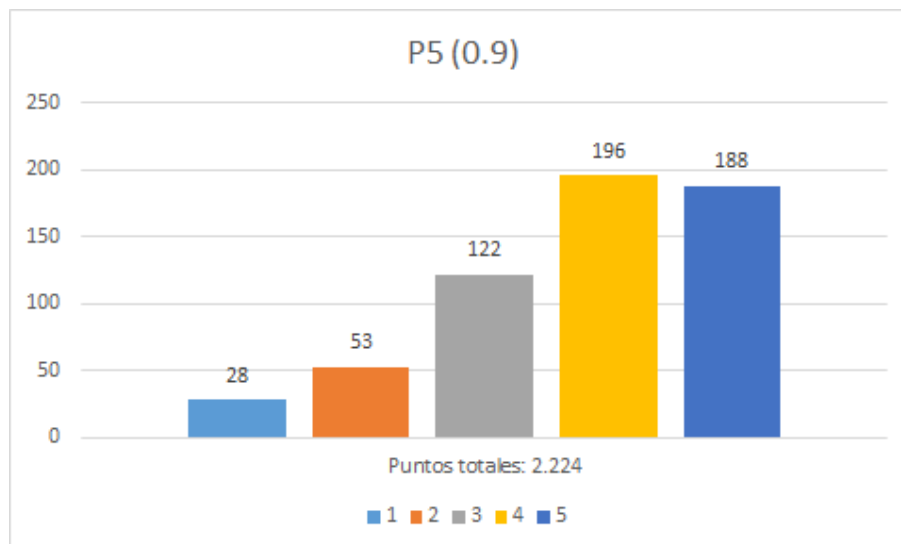


Ilustración 13 - Encuesta P5 (0.9)

Se han escogido la P5 (0.1) y la P5 (0.9). Aunque también muy similares, esta vez se ve más claramente la inclinación del público hacia el primer plan, superando la puntuación 5 al 4 por primera vez en estas encuestas. Sin embargo, si comparamos las puntuaciones, son muy similares, pero la P5 (0.1) es superior. Dejando así claro que la aleatoriedad crea un interés mayor que las acciones más prefijadas a un final.

6.8 Comentarios generales de los resultados

A lo largo de estos experimentos, además de comprobar el correcto funcionamiento del sistema, se ha buscado analizar las características de los planes generados. Estos planes, de problemas diferentes, han seguido una serie de pautas que podemos aprender:

- En cuanto a la **longitud** de las narraciones, no dependen demasiado de la aleatoriedad que tenga el planificador. Sobre todo se ve influenciada por el objetivo final, cuantos más objetos intervengan, más larga por lo general será la narración.
- En cambio, la **diversidad** sí que es totalmente dependiente de la aleatoriedad. Recordamos que la diversidad consistía en la cantidad de acciones diferentes entre una narración y la que se acababa de crear antes que ella. En algunos casos, la diversidad ha llegado a ser casi el doble que la longitud media, lo cual implica que son planes casi totalmente diferentes.
- El **interés** de los espectadores se ha mantenido prácticamente constante a lo largo de los problemas, pero en todos ellos se ha mostrado una cierta predilección por aquellos que son más aleatorios. Por tanto podemos decir que la aleatoriedad influye positivamente en el interés de las narraciones generadas.
- Por último, ¿cuál es el motivo por el cual **todas las narraciones se han generado al completo**? El secreto reside en los objetivos finales, deben ser sencillos, como: muerto, herido, tiene un objeto... Normalmente no se suelen completar cuando se añaden predicados del tipo “estar en un lugar”, ya que si ese personaje no sobrevive o bien no tiene ninguna razón para ir allí, el planificador fallará. También sucede con predicados contrarios, por ejemplo: que A mate a B y que B mate a A. Como es comprensible, sin una acción que permita la resurrección o hacer una muerte en el tiempo, es imposible.

Por último, destacar la acción *make-enemy-motive*, que como se puede leer en las narraciones, se abusa de ella al principio de las mismas. Esto es debido a que al no requerir lugar, es la manera más sencilla de declarar enemigos. En una prueba se intentó añadir lugar a la acción, pero entonces a los personajes les costaba muchísimo arrancar en la narración, además de los costes adicionales de computación que esto conllevaba.

7. Gestión del proyecto

7.1 Planificación inicial

El 10 de febrero de 2014 se comenzó con el desarrollo de este proyecto. Se planeó que se iba a invertir de 2 a 3 horas diarias de lunes a viernes desde el día del comienzo hasta su finalización.

El desarrollo del trabajo se ha organizado en diversas fases específicas:

- Consultoría inicial: se trata de las tutorías con el profesor acerca del comienzo del proyecto, así como la resolución de dudas que puedan surgir a lo largo de su desarrollo.
- Planificación del Sistema: consiste en la formación de los requisitos necesarios del sistema, así como la formación necesaria en el lenguaje PDDL y el planificador Sayphi.
- Diseño del Sistema: es la fase en la cual se determinan los aspectos de diseño del dominio, desarrollando metodologías que se implementarán para conseguir el correcto funcionamiento del sistema.
- Realización: consiste en el desarrollo en código del dominio y los problemas
- Pruebas: es la fase de experimentación, en la que se desarrolla el plan de pruebas y se obtienen los resultados.
- Documentación: Por último, la documentación final del proyecto.

En la primera semana de trabajo, se estimó que estas etapas se desarrollarían en el tiempo siguiendo el siguiente esquema:

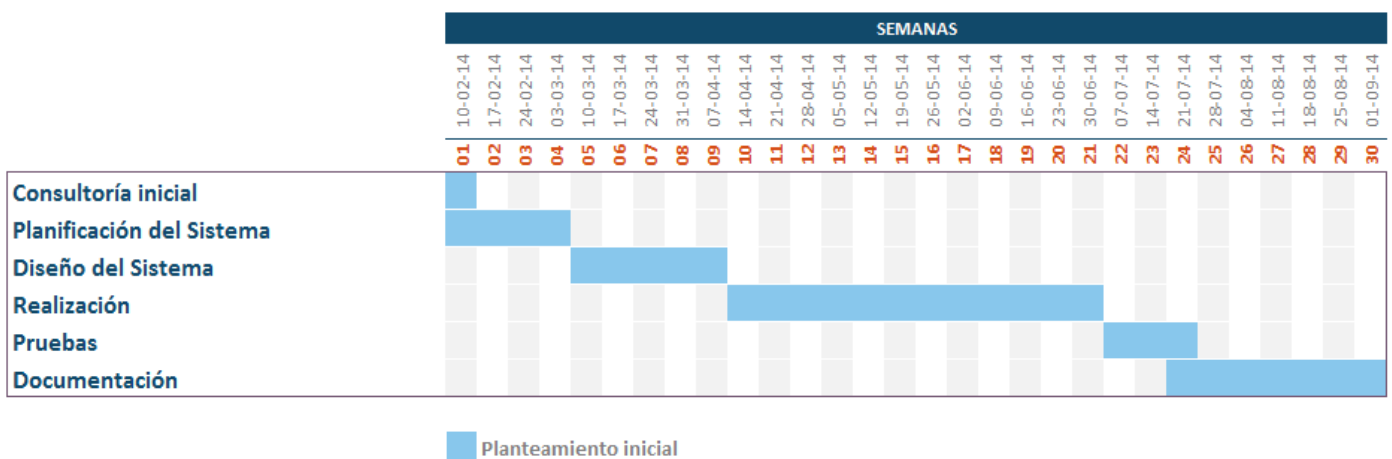


Ilustración 14 - Diagrama de Gantt Inicial

7.2 Planificación final

Sin embargo, tras su realización no todo se ha seguido según lo previsto. Debido a algunos problemas, retrasos o semanas de vacaciones ha habido algunas variaciones, siendo así el esquema resultante:

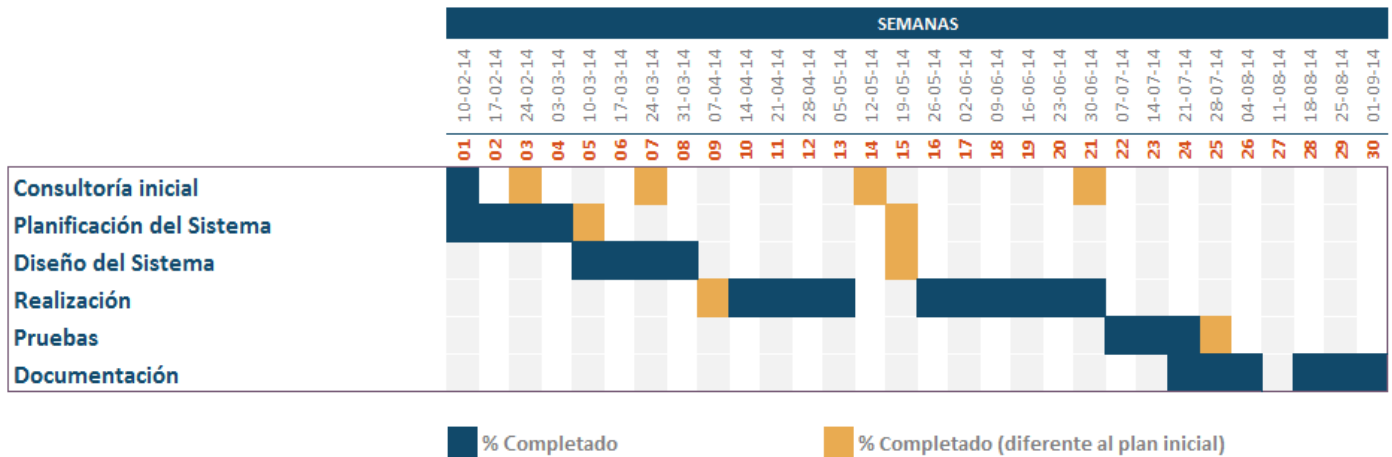


Ilustración 15 - Diagrama de Gantt Final

En primer lugar vemos que la fase de consultoría inicial han sido más de las esperadas, debido a las múltiples dudas o errores de concepto en el diseño. Por ejemplo, en la semana 14 se ve un parón en la realización del sistema, ya que el diseño que se seguía hasta la fecha no era correcto. Tras una tutoría con el profesor, se volvió a plantear y diseñar el sistema.

El resto de etapas se han seguido bastante bien, salvo en algunos casos que se ha acabado alguna después de lo previsto o se ha comenzado antes. Es de destacar la semana 27, correspondiente a una semana de vacaciones, en la que no se ha realizado ninguna hora de trabajo.

7.3 Presupuesto

Para la realización del presupuesto, se ha tenido en cuenta la implicación económica que tendrían los costes asociados al proyecto, tanto si el mismo se realiza por cuenta propia o ajena. Por tanto, es preciso analizar cuáles van a ser dichos costes y por lo tanto se pasa a explicar el análisis realizado para su obtención.

El proyecto va a tener un coste que representa un ratio importante en la partida de personal. Una premisa a considerar es que en España el salario de un programador oscila según los diferentes convenios colectivos de los diversos sectores. Para un programador junior, el coste/hora se encuentra en un intervalo entre los 15 y 25 euros/hora. Para el proyecto, el coste/hora utilizado es de 16,48€ y el resultado (según las distintas fases) sería el siguiente:

FASES	HORAS	COSTE IMPORTE (€)
Consultoría Inicial	5	82,40
Planificación del Sistema	45	741,60
Diseño del Sistema	55	906,40
Realización	150	2.472,00
Pruebas	30	494,40
Documentación	100	1.648,00
TOTALES	385,00	6.344,80

Tabla 28 - Coste de personal

Como vemos, los gastos de personal ascienden a **6.344,80 €** para las horas invertidas, no las presenciales o de convenio. Los costes de personal de estructura siempre representan el ratio más importante (y a veces el más elevado) en la Cuenta de Resultados para una empresa de servicios.

En segundo lugar, tenemos que tener en cuenta otra serie de partidas que forman parte de los gastos de estructura, tanto fijos como variables, que se engloban en “otros gastos de explotación”, como son los arrendamientos, mantenimientos, suministros (combustible, electricidad, gas, teléfono, etc...), seguros, asesoramientos exteriores, etc...

Otra partida importante a analizar, previa al Resultado de Explotación, son las dotaciones de Amortización del Inmovilizado. En el hipotético caso de que sólo consideráramos la partida de Elementos de Información (software y hardware), la amortización extrapolada según los meses del proyecto (7 meses) sería la siguiente:

ELEMENTO	DESCRIPCIÓN DEL BIEN	VALOR DEL BIEN CON IVA (€)	VALOR A AMORT.	MÍNIMO DE MESES PARA AMORT.	% MÁXIMO AMORT. ANUAL	MESES	TOTAL AMORT. (€)
Hardware	Acer AG3-605 Intel Core I7 4770 (Ampliado)	1.184,00 €	978,51 €	48	25%	7	142,70 €
Software	Microsoft Office Profesional 2013 (1 Pc)	539,00 €	445,45 €	36,36	33%	7	85,75 €
	SAYPHI	0,00 €	0,00 €	36,36	33%	7	0,00 €
	Otro software gratuito	0,00 €	0,00 €	36,36	33%	7	0,00 €
			1.423,96 €				228,45 €

Tabla 29 - Amortizaciones

Como podemos observar, la dotación de amortización a aplicar ascendería a **228,45 €**.

Si tenemos en cuenta los resultados anteriores, y el departamento administrativo-financiero nos indica que la parte de gastos generales a aplicar al proyecto (en los que se engloban un % de los “otros gastos de explotación” y un % de los resultados financieros), ascenderán a un valor de **550 €**, y que en caso de comercializar el producto, dicho departamento estima que en la primera venta se recuperan los costes, los gastos aplicados, las dotaciones, estando por encima del punto de equilibrio y, por lo tanto, alcanzando un resultado antes de impuestos equivalente a un ratio del 8% en función de ingresos, podemos determinar el presupuesto siguiente:

COSTES, GASTOS Y DOTACIONES	IMPORTE
Personal	6.344,80 €
% de Gastos generales	550,00 €
Amortización	228,45 €
TOTALES	7.123,25 €

Impuestos no incluidos

Tabla 30 - Presupuesto final

8. Conclusiones

Se ha desarrollado un sistema que genere automáticamente narrativas de interés, con factor de aleatoriedad, basado en la narrativa de aventura fantástica. A lo largo de este documento hemos aprendido el uso del lenguaje PDDL, con el que hemos creado el dominio y los problemas necesarios, así como diferentes metodologías de narración, analizando cuáles han sido más efectivas y por tanto se han implementado en este sistema.

Al ser PDDL1.2 un lenguaje muy básico y el planificador Sayphi tan sólo funciona correctamente con esta versión, se ha limitado el desarrollo de la narración, ya que con el uso de otros mecanismos ya implementados en versiones posteriores del lenguaje, se hubiesen podido conseguir unas acciones más interesantes, como sucede con el movimiento en grupos, que desgraciadamente no ha podido ser implementado.

Pese a la sencillez del lenguaje, en diversas ocasiones surgen errores si no se sigue el formato correcto, como es lógico, pero suele ser difícil encontrar el problema. Sería interesante desarrollar un proyecto que procesase el lenguaje y fuese capaz de detectar errores de compilación en tiempo de desarrollo, como hace por ejemplo la herramienta *Eclipse*.

Como mejoras para el futuro, se debería intentar migrar el sistema a PDDL3.1 y poder realizar acciones más complejas pero con un mayor sentido e interés narrativo. También sería estupendo adaptar el programa para servir de base a la televisión, cine, videojuegos... etc. Siempre, por supuesto, adecuando el programa a las necesidades requeridas.

En definitiva, el objetivo principal se ha cumplido con éxito, y tras realizar las encuestas para la fase de experimentación se ha visto que ha tenido una buena aceptación, ya que la mayoría consideraba como buenas o muy buenas las narraciones con respecto al interés generado.

9. Referencias

- [1] Ghallab, M., Howe, A., Knoblock, C., McDermott, D., Ram, A., Veloso, M., Weld, D. & Wilkins, D. (1998). PDDL - The Planning Domain Definition Language. *The AIPS-98 Planning Competition Comitee*.
- [2] Fox, M. & Long, D. (2003). PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. *Journal of Artificial Intelligence Research*, Vol. 20, pp. 61-124.
- [3] Edelkamp, S & Hoffmann, J. (2004). PDDL2.2: The Language for the Classical Part of the 4th International Planning Competition. *Journal of Artificial Intelligence Research*, Vol. 24, pp. 519-579.
- [4] Gerevini, A. & Long, D. (2006). Preferences and Soft Constraints in PDDL3. *Proceedings of the ICAPS-2006 Workshop on Preferences and Soft Constraints in Planning*. pp. 46-54.
- [5] De la Rosa, T., García, A. & Borrajo, D. (2013). A Case-Based Approach to Heuristic Planning. *Applied Intelligence journal*, Springer Verlag, Vol. 39, pp.184-201.
- [6] Haslum, P. (2012). Narrative planning: compilations to classical planning. *Journal of Artificial Intelligence Research*, Vol. 44, pp. 383-395.
- [7] Riedl, M., & Young, R. (2010). Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*, Vol. 39, pp. 217-268.
- [8] Peinado, F. & Gervás, P. (2006). Evaluation of Automatic Generation of Basic Stories. *New Generation Computing*, Vol. 24, pp. 289-302.
- [9] Porteous, J., Cavazza, M. & Charles, F. (2010). Applying Planning to Interactive Storytelling: Narrative Control using State Constraints. *ACM Transactions on Intelligent Systems and Technology*, Vol. 1, article No. 10.
- [10] Pérez, P. (2014). Generación Automática de Narrativas. *Universidad Carlos III de Madrid*, Trabajo Fin de Grado.

Anexo I: Manual de usuario

En esta sección se va a enseñar al usuario a utilizar y ejecutar el planificador Sayphi con el dominio y problemas desarrollados en este documento. Se ha empleado Windows 7 como sistema operativo y SBCL como entorno donde ejecutar el planificador. Este entorno se puede descargar gratuitamente y deberemos instalar la versión de 64 bits, muy importante ya que se requiere de una gran cantidad de memoria para ejecutar los problemas.

- Una vez tenemos instalado el SBCL en nuestra máquina, deberemos cambiar las rutas de nuestro fichero `.sbclrc` encontrado en el planificador Sayphi:

```
;; I load all the common Lisp functions
#+linux (load "/home/javi/1TFG/Compilador/narrative/.init.lisp")
#+darwin (load "/Users/javi/1TFG/Compilador/narrative/.init.lisp")
#+win32 (load "d:/1TFG/Compilador/narrative/.init.lisp")
```

Ilustración 16 - Rutas `.sbclrc`

- Por último, editaremos el archivo `.init`, donde pondremos la ruta del Sayphi:

```
(defvar *my-path*
#+(or macos darwin) (load "/Users/javi/1TFG/Compilador/narrative/")
#+linux (load "/home/javi/1TFG/Compilador/narrative/")
#+win32 (load "d:/1TFG/Compilador/narrative/"))
```

Ilustración 17 - Rutas `.init`

Para ejecutar el planificador con un dominio y un problema, es necesario seguir las siguientes instrucciones:

- En primer lugar, se debe abrir la consola de comandos (escribiendo "cmd" en el inicio).
- A continuación, ejecutaremos el SBCL. Para ello, necesitaremos indicarle la memoria necesaria. Los problemas grandes como los descritos en este proyecto pueden llegar a necesitar de 8 a 10GB o más, dependiendo de la complejidad de dichos problemas. Para ellos escribiremos, para el caso de 10GB:

```
sbcl --dynamic-space-size 10000
```

- Cargaremos el archivo `.sbclrc` que editamos anteriormente, para cargar los archivos al programa, de la siguiente manera:

```
(load "d:/1TFG/Compilador/narrative/.sbclrc")
```

- A continuación, cargaremos y compilaremos Sayphi:

```
(load-ma-sayphi)
```

- Por último, cargamos la instrucción que ejecuta un problema en un dominio. Que está compuesta por:

```
(narrative-planner 'rrt "narrative" "p1.pddl" :domain-file  
"domain.pddl" :probability-ehc 0.9 :timeout 300 :number-  
plans 4 :narrate-p t)
```

- *"narrative"*: es el nombre de la carpeta situada en directorio/planning/sayphi/domains/, donde estará el dominio. Además, también se encontrará la carpeta *probsets*, donde estarán sus correspondientes problemas.
- *"p1.pddl"*: es el nombre del problema.
- *"domain.pddl"*: es el nombre del dominio.
- *'rrt'*: es el algoritmo de planificación.
- *:probability-ehc 0.9*: marca la aleatoriedad que usará el planificador, siendo 0 lo máximo y 1 lo mínimo.
- *:timeout 300*: es el tiempo máximo para generar un plan. Si se pasa el tiempo, se abortará la ejecución.
- *:number-plans 4*: es el número de planes que se obtendrán. Calcula automáticamente la diversidad entre ellos si hay más de 1.
- *:narrate-p t*: Si se define el valor *"t"*, se generan las narrativas. De lo contrario se mostrarán las acciones con sus parámetros.

Anexo II: Código

Dominio

```
(define (domain narrative)
  (:requirements :strips :typing :equality)

  (:types
    edible dangerous - animal
    character animal - entity
    place
    goodmotive evilmotive neutralmotive - motive
    arm usefultobj - storable
    entity storable - object
    status
  )

  (:constants
    Destroyed Acquired - status
    Neutral - neutralmotive
  )

  (:predicates
    ; PLACES AND TRAVEL
    (at ?obj - object ?place - place)
    (can-be-destroyed-in ?obj - object ?place - place)

    ; CHARACTER STATUS
    (hurt ?char - character)
    (dead ?ent - entity)
    (invisible ?char - character)
    (weak ?char - character)
    (strong ?obj - object)

    (isNPC ?ent - entity)
    ; NoPlayerCharacter, como suele ser en los juegos, un
    personaje que no se mueve ni interviene (e.j: un tabernero)
    ; Los NPC podran hacer todo, excepto moverse

    (killed-by ?ent - entity ?ent2 - entity)
    ; ?ent ha sido asesinado por ?ent2 (character, animal...)

    (is ?obj - object ?st - status)

    ; FRIENDSHIP
    (friends ?char - character ?char2 - character)
    (enemies ?char - character ?char2 - character)

    ; OBJECTS
    (have-an-obj ?char - character ?obj - storable)
```



```
; Tiene un objeto Armas y utiles (p.e. un anillo)
(wish-an-obj ?char - character ?obj - storable)
; Desea un objeto

(can-use-magic-kill ?char - character)
(can-use-magic-heal ?char - character)
(magic-invisible ?obj - usefobj)
(magic-kill ?obj - usefobj)
(magic-heal ?obj - usefobj)

; MOTIVES
(have-motive ?char - character ?m - motive)
; A definir por el problema: Dinero, Salvar personas,
Poder... especificos de cada personaje
(implies-that ?m - motive ?m2 - motive)
; M se consigue mediante M2
(want-motive ?m - motive ?obj - object ?st - status)
; M quiere que OBJ esté ST
(motive-complete ?char - character ?m - motive)
)

; ooooooooooooooooooooooooooooo
; ----- ACTIONS -----
; ooooooooooooooooooooooooooooo

; //////////////////////////////////
; OBJECTS
; \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

(:action take-obj ; Coger un objeto
:parameters (?char - character ?obj - storable ?place -
place)
:precondition (and
(not (dead ?char))
(at ?char ?place)
(at ?obj ?place)
)
:effect (and
(not (at ?obj ?place))
(have-an-obj ?char ?obj)
)
)

(:action give-obj ; Dar un objeto sin nada a cambio ni estar en deuda
:parameters (?char - character ?char2 - character ?obj -
storable ?place - place)
:precondition (and
(not (dead ?char))
(not (dead ?char2))
```



```
(not (= ?char ?char2))
(at ?char ?place)
(at ?char2 ?place)
(have-an-obj ?char ?obj)
(not (invisible ?char))
(not (invisible ?char2))

; El que lo da no debe quererlo
(not (wish-an-obj ?char ?obj))

; Deben ser amigos
(or
  (friends ?char ?char2)
  (friends ?char2 ?char)
)
)
:effect (and
  (not (have-an-obj ?char ?obj))
  (have-an-obj ?char2 ?obj)
)
)

(:action trade-obj ; Comerciar con un objeto
:parameters (?char - character ?char2 - character ?obj -
  storable ?obj2 - storable ?place - place)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (= ?char ?char2))
  (at ?char ?place)
  (at ?char2 ?place)
  (have-an-obj ?char ?obj)
  (have-an-obj ?char2 ?obj2)
  (not (invisible ?char))
  (not (invisible ?char2))

  (wish-an-obj ?char2 ?obj)
  (wish-an-obj ?char ?obj2)
)
:effect (and
  (not (have-an-obj ?char ?obj))
  (not (have-an-obj ?char2 ?obj2))
  (have-an-obj ?char2 ?obj)
  (have-an-obj ?char ?obj2)
)
)

(:action loot-obj ; Despojar un objeto
:parameters (?char - character ?char2 - character ?obj -
  storable ?place - place)
:precondition (and
  (dead ?char2)
```




```

        (not (dead ?char))
        (at ?char ?place)
        (at ?char2 ?place)
        (have-an-obj ?char2 ?obj)
    )
    :effect      (and
        (not (have-an-obj ?char2 ?obj))
        (have-an-obj ?char ?obj)
    )
)

; //////////////////////////////////
; KILLING ACTIONS
; \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

(:action hurt-punch
    :parameters (?char - character ?char2 - character ?place -
        place)
    :precondition (and
        (enemies ?char ?char2)

        (not (= ?char ?char2))
        (not (dead ?char))
        (not (dead ?char2))
        (at ?char ?place)
        (at ?char2 ?place)

        (not (hurt ?char2))
        (not (invisible ?char2))

        ; No debe ser debil, si lo es,
        debe ser invisible, asi que...
        ; O bien no es debil o es invisible
        (or
            (not (weak ?char))
            (invisible ?char)
        )
        (not (strong ?char2))
    )
    :effect      (and
        (hurt ?char2)
    )
)

(:action hurt-arm
    :parameters (?char - character ?char2 - character ?place -
        place ?arm - arm)
    :precondition (and

```



```
(enemies ?char ?char2)

(not (dead ?char))
(not (dead ?char2))
(not (= ?char ?char2))
(not (hurt ?char2))

(at ?char ?place)
(at ?char2 ?place)
(have-an-obj ?char ?arm)

(not (invisible ?char2))
(not (strong ?char2))
)
:effect      (and
              (hurt ?char2)
)
)

(:action kill-arm
  :parameters (?char - character ?char2 - character ?place -
               place ?arm - arm)
  :precondition (and
                (not (dead ?char))
                (not (= ?char ?char2))
                (at ?char ?place)
                (at ?char2 ?place)

                (enemies ?char ?char2)

                (have-an-obj ?char ?arm)
                (hurt ?char2)

                (not (invisible ?char2))
                (not (strong ?char2))
)
  :effect      (and
                (not (hurt ?char2))
                (dead ?char2)
                (killed-by ?char2 ?char)
)
)

; ANIMALS

(:action kill-and-eat-animal
  :parameters (?char - character ?animal - edible ?place -
               place ?arm - arm)
  :precondition (and
                (not (dead ?char))
```

```
(not (dead ?animal))
(at ?char ?place)
(at ?animal ?place)
(have-an-obj ?char ?arm)

; Debe estar herido para comer y restaurarse
(hurt ?char)
)
:effect      (and
              (not (hurt ?char))
              (dead ?animal))
)
)

(:action hurt-by-animal
 :parameters (?char - character ?animal - dangerous ?place -
              place)
 :precondition (and
               (not (dead ?char))
               (not (dead ?animal))
               (not (hurt ?char))
               (at ?char ?place)
               (at ?animal ?place)
               (not (invisible ?char))
               (not (strong ?char))
               )
 :effect      (and
               (hurt ?char)
               )
)

(:action kill-by-animal
 :parameters (?char - character ?animal - dangerous ?place -
              place)
 :precondition (and
               (not (dead ?char))
               (not (dead ?animal))

               (hurt ?char)

               (at ?char ?place)
               (at ?animal ?place)
               (not (invisible ?char))
               (not (strong ?char))
               )
 :effect      (and
               (not (hurt ?char))
               (dead ?char)
               (killed-by ?char ?animal))
)
```




```
:parameters (?char - character ?char2 - character ?place -
place ?obj - usefobj)
:precondition (and
  (not (dead ?char))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))

  ;Para sanarle, debe conocerle,
  por tanto deben ser amigos al menos
  (or
    (friends ?char ?char2)
    (friends ?char2 ?char)
  )

  (hurt ?char2)

  (have-an-obj ?char ?obj)
  (magic-heal ?obj)
  (can-use-magic-heal ?char)

  (not (invisible ?char2))
)
:effect (and
  (not (hurt ?char2))
)
)

(:action heal-magic-himself
:parameters (?char - character ?obj - usefobj)
:precondition (and
  (not (dead ?char))

  (hurt ?char)

  (have-an-obj ?char ?obj)
  (magic-heal ?obj)
  (can-use-magic-heal ?char)
)
:effect (and
  (not (hurt ?char))
)
)

(:action magic-become-invisible
:parameters (?char - character ?obj - usefobj)
:precondition (and
  (not (dead ?char))
  (have-an-obj ?char ?obj)
  (magic-invisible ?obj)
  (not (invisible ?char))
)
)
```



```
        :effect      (and
                      (invisible ?char)
                    )
    )
)

(:action magic-become-visible
 :parameters (?char - character ?obj - usefultobj)
 :precondition (and
                (not (dead ?char))
                (invisible ?char)
                (have-an-obj ?char ?obj)
                (magic-invisible ?obj)
              )
 :effect      (and
                (not (invisible ?char))
              )
)

; //////////////////////////////////
;     MOTIVE ACTIONS
; \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

; SAME MOTIVE

(:action same-motive-friends ; Pedirle a un amigo que siga tu motivo
 :parameters (?char - character ?char2 - character ?place -
 place ?m - motive)
 :precondition (and
                (not (dead ?char))
                (not (dead ?char2))
                (at ?char ?place)
                (at ?char2 ?place)
                (not (= ?char ?char2))
                (not (invisible ?char))
                (not (invisible ?char2))

                (or
                  (friends ?char ?char2)
                  (friends ?char2 ?char)
                )

                (have-motive ?char ?m)
                (not (have-motive ?char2 ?m))
              )
 :effect      (and
                (have-motive ?char2 ?m)
              )
)

; MAKE FRIENDS
```

```
(:action make-friends-motive
; Hacerse amigo de alguien por algun motivo comun
:parameters (?char - character ?char2 - character ?place -
place ?m - motive)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (invisible ?char))
  (not (invisible ?char2))

  (have-motive ?char ?m)
  (have-motive ?char2 ?m)

  (not (friends ?char ?char2))
  ; Si ya son amigos no tiene sentido esta accion
  (not (friends ?char2 ?char))
)
:effect (and
  (friends ?char ?char2)
  (friends ?char2 ?char)
  (not (enemies ?char ?char2))
  ; Recalcamos el hecho de que no sean enemigos
  (not (enemies ?char2 ?char))
  ; Asi no pueden ser amigos y enemigos a la vez
)
)
```

```
(:action make-friends-want-kill
; Hacerse amigo de alguien por algun enemigo comun
:parameters (?char - character ?char2 - character ?place -
place ?enemy - character)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (dead ?enemy))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (invisible ?char))
  (not (invisible ?char2))

  (enemies ?char ?enemy)
  (enemies ?char2 ?enemy)

  (not (friends ?char ?char2))
  ; Si ya son amigos no tiene sentido esta accion
  (not (friends ?char2 ?char))
)
```



```
)
:effect      (and
              (friends ?char ?char2)
              (friends ?char2 ?char)
              (not (enemies ?char ?char2))
              ; Recalcamos el hecho de que no sean enemigos
              (not (enemies ?char2 ?char))
              ; Asi no pueden ser amigos y enemigos a la vez
              )
)

(:action make-friends-friends
; Hacerse amigo "Friends of my friends are my friends"
:parameters (?char - character ?char2 - character ?place -
place ?char3 - character)
:precondition (and
              (not (dead ?char))
              (not (dead ?char2))
              (not (dead ?char3))
              (at ?char ?place)
              (at ?char2 ?place)
              (not (= ?char ?char2))
              (not (= ?char3 ?char2))
              (not (= ?char ?char3))
              (not (invisible ?char))
              (not (invisible ?char2))

              (friends ?char ?char3)
              (friends ?char2 ?char3)

              (not (friends ?char ?char2))
              ; Si ya son amigos no tiene sentido esta accion
              (not (friends ?char2 ?char))
              )
:effect      (and
              (friends ?char ?char2)
              (friends ?char2 ?char)
              (not (enemies ?char ?char2))
              ; Recalcamos el hecho de que no sean enemigos
              (not (enemies ?char2 ?char))
              ; Asi no pueden ser amigos y enemigos a la vez
              )
)

; ENEMIES

(:action make-enemy-motive
; Hacerse "enemigo" de alguien por algun motivo dispar
:parameters (?char - character ?char2 - character ?gm -
goodmotive ?em - evilmotive ?place - place)
```




```
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (= ?char ?char2))

  (have-motive ?char ?gm)
  (have-motive ?char2 ?em)

  (not (enemies ?char ?char2))
  ; Si ya son enemigos no tiene sentido esta accion
  (not (enemies ?char2 ?char))
)
:effect (and
  (enemies ?char ?char2)
  (enemies ?char2 ?char)
  (not (friends ?char ?char2))
  ; Recalcamos el hecho de que no sean amigos
  (not (friends ?char2 ?char))
  ; Asi no pueden ser amigos y enemigos a la vez
)
)

(:action make-enemy-friends
; Hacerse enemigo "Enemies of my friends are my enemies"
:parameters (?char - character ?char2 - character ?place -
place ?char3 - character)
:precondition (and
  (not (dead ?char))
  (not (dead ?char2))
  (not (dead ?char3))
  (at ?char ?place)
  (at ?char2 ?place)
  (not (= ?char ?char2))
  (not (= ?char3 ?char2))
  (not (= ?char ?char3))
  (not (invisible ?char))
  (not (invisible ?char2))

  (friends ?char ?char2)

  (enemies ?char2 ?char3)

  (not (enemies ?char ?char3))
  ; Si ya son enemigos no tiene sentido esta accion
  (not (enemies ?char3 ?char))
)
:effect (and
  (enemies ?char ?char3)
  (enemies ?char3 ?char)
  (not (friends ?char ?char3))
  ; Recalcamos el hecho de que no sean amigos
  (not (friends ?char3 ?char))
)
```

```
        ; Asi no pueden ser amigos y enemigos a la vez
    )
)

(:action make-enemy-kill-friend
; Hacerse enemigo de alguien que ha matado a tu amigo
:parameters (?char - character ?char2 - character ?place -
place ?char3 - character)
:precondition (and
    (not (dead ?char))
    (dead ?char2)
    (at ?char ?place)
    (at ?char2 ?place)
    (not (= ?char ?char2))
    (not (= ?char3 ?char2))
    (not (= ?char ?char3))
    (not (invisible ?char))
    (not (invisible ?char2))

    (friends ?char ?char2)

    (killed-by ?char2 ?char3)

    (not (enemies ?char ?char3))
; Si ya son enemigos no tiene sentido esta accion
    (not (enemies ?char3 ?char))
)
:effect      (and
    (enemies ?char ?char3)
    (enemies ?char3 ?char)
    (not (friends ?char ?char3))
; Recalcamos el hecho de que no sean amigos
    (not (friends ?char3 ?char))
; Asi no pueden ser amigos y enemigos a la vez
)
)

; LEARNING

(:action learn-motive
:parameters (?char - character ?m - motive ?m2 - motive)
:precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (implies-that ?m ?m2)
)
:effect      (and
    (have-motive ?char ?m2)
)
)
```



; A partir de un motivo, querer cosas, ir a un lugar...

```
(:action learn-motive-wish-obj
  :parameters (?char - character ?m - motive ?obj - storable
    ?st - status)
  :precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (want-motive ?m ?obj ?st)
    ;Al ser storable, sea como sea el status como lo
    quiera, lo querra tener para hacerlo posible
  )
  :effect (and
    (wish-an-obj ?char ?obj)
  )
)
```

; Una vez que quieres el objeto, quieres ir a donde esta o ir donde esta su poseedor

```
(:action learn-place-obj
  :parameters (?char - character ?obj - storable ?place -
    place ?place2 - place)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    (wish-an-obj ?char ?obj)
    (at ?char ?place)
    (at ?obj ?place2)
  )
  :effect (and
    ;(want-to-go ?char ?place)
    (not (at ?char ?place))
    (at ?char ?place2)
  )
)
```

```
(:action learn-place-obj-char
  :parameters (?char - character ?char2 - character ?obj -
    storable ?place - place ?place2 - place)
  :precondition (and
    (not (isNPC ?char))
    (not (= ?place ?place2))
    (not (dead ?char))
    ;Puede que este muerto pero siga teniendo el objeto
    (wish-an-obj ?char ?obj)
    (at ?char ?place)
    (at ?char2 ?place2)
    (have-an-obj ?char2 ?obj)
  )
)
```

```
    :effect      (and
                  (not (at ?char ?place))
                  (at ?char ?place2)
                  ; No establecemos que sean enemigos para que tengan
                  ; oportunidad de intercambiarlo o estar en deuda
                )
  )
)

(:action learn-place-give-obj-char ; Ir a darle un objeto a un amigo
:parameters (?char - character ?char2 - character ?obj -
             storable ?place - place ?place2 - place)
:precondition (and
               (not (isNPC ?char))
               (not (= ?place ?place2))
               (not (dead ?char))
               (not (dead ?char2))
               (wish-an-obj ?char2 ?obj)
               (at ?char ?place)
               (at ?char2 ?place2)
               (have-an-obj ?char ?obj)
               (or
                 (friends ?char ?char2)
                 (friends ?char2 ?char)
               )
             )
)
:effect      (and
              (not (at ?char ?place))
              (at ?char ?place2)
            )
)

(:action learn-place-obj-be-destroyed
:parameters (?char - character ?obj - storable ?place -
             place ?place2 - place ?m - motive)
:precondition (and
               (not (isNPC ?char))
               (not (= ?place ?place2))
               (not (dead ?char))
               (have-an-obj ?char ?obj)
               (at ?char ?place)
               (can-be-destroyed-in ?obj ?place2)
               (have-motive ?char ?m)
               (want-motive ?m ?obj Destroyed)
             )
)
:effect      (and
              (not (at ?char ?place))
              (at ?char ?place2)
            )
)

(:action learn-place-enemy
```

```
:parameters (?char - character ?char2 - character ?place -
place ?place2 - place)
:precondition (and
  (not (isNPC ?char))
  (not (= ?place ?place2))
  (not (dead ?char))
  (not (dead ?char2))
  (not (weak ?char))
  ; Si es debil no querra ir a por el enemigo
  (enemies ?char ?char2)
  (at ?char ?place)
  (at ?char2 ?place2)
)
:effect (and
  (not (at ?char ?place))
  (at ?char ?place2)
)
)

(:action learn-place-food
:parameters (?char - character ?animal - edible ?place -
place ?place2 - place)
:precondition (and
  (not (isNPC ?char))
  (not (= ?place ?place2))
  (not (dead ?char))
  (hurt ?char)
  (at ?char ?place)
  (at ?animal ?place2)
)
:effect (and
  (not (at ?char ?place))
  (at ?char ?place2)
)
)

(:action learn-place-heal
:parameters (?char - character ?char2 - character ?place -
place ?place2 - place ?obj - usefobj)
:precondition (and
  (not (isNPC ?char))
  (not (= ?place ?place2))
  (not (dead ?char))
  (hurt ?char)
  (at ?char ?place)
  (at ?char2 ?place2)
  (have-an-obj ?char2 ?obj)
  (magic-heal ?obj)
  (can-use-magic-heal ?char2)
)
:effect (and
  (not (at ?char ?place))
```



```
        (at ?char ?place2)
      )
    )

; Acciones de destruir o adquirir un objeto para conseguir los
motivos

(:action destroy-obj
  :parameters (?char - character ?m - motive ?obj - object
    ?place - place)
  :precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (want-motive ?m ?obj Destroyed)
    (at ?char ?place)
    (have-an-obj ?char ?obj)
    (can-be-destroyed-in ?obj ?place)
    (not (invisible ?char))
  )
  :effect (and
    (motive-complete ?char ?m)
    (is ?obj Destroyed)
    (not (have-an-obj ?char ?obj))
    ; Ya no tiene el motivo
    (not (have-motive ?char ?m))
    (have-motive ?char Neutral)
  )
)

(:action acquire-obj
  :parameters (?char - character ?m - motive ?obj - object)
  :precondition (and
    (not (dead ?char))
    (have-motive ?char ?m)
    (want-motive ?m ?obj Acquired)
    (have-an-obj ?char ?obj)
  )
  :effect (and
    (motive-complete ?char ?m)
    (is ?obj Acquired)
    ; Ya no tiene el motivo
    (not (have-motive ?char ?m))
    (have-motive ?char Neutral)
  )
)

(:action kill-strong
  :parameters (?char - character ?char2 - character ?place -
    place ?arm - arm ?m - motive)
  :precondition (and
    (not (dead ?char))
    (not (dead ?char2))
```



```
(not (= ?char ?char2))
(at ?char ?place)
(at ?char2 ?place)
(not (weak ?char))

      (enemies ?char ?char2)

(have-an-obj ?char ?arm)

(not (invisible ?char2))
(strong ?char2)
(strong ?arm)

(have-motive ?char ?m)
(want-motive ?m ?arm Acquired)
)
:effect      (and
              (dead ?char2)
              (killed-by ?char2 ?char)
)
)
)
```



Problema P1.PDDL

```
(define (problem p1) ; Problema EL SEñOR DE LOS ANILLOS - Dificultad
(1/5)
(:domain narrative)
(:objects
  La_Comarca Rohan Rivendel Mordor Gondor Mount_Doom Moria
  Bree Isengard - place
  Bilbo Frodo Sam Merry Pippin - character ; Los Hobbits
  Aragorn Legolas Gimli Gandalf Boromir Elrond - character
  ; Humanos, elfos y enanos
  Saruman Sauron Gollum - character
  ; Los malos (gollum es un caso especial)
  Nazgul - character
  Orc1 Orc2 Orc3 - character
  Rabbit1 Rabbit2 - edible
  Wolf1 Wolf2 - dangerous

  Ring - usefulobj
  ; Un anillo para gobernarlos a todos,
  ; un anillo para encontrarlos,
  ; un anillo para atraerlos a todos
  ; y atarlos en las tinieblas.

  AndurilSword - arm
  ; La espada creada a partir de los trozos rotos
  ; de la espada Narsil,
  ;que corto el anillo del dedo de Sauron

  GandalfStaff SarumanStaff ElfMedicine - usefulobj
  NazgulSword - arm
  Orc1Sword Orc2Sword Orc3Sword - arm
  AragornSword LegolasBow GimliAxe BoromirSword - arm
  LittleElfSword - arm
  Peace Destroy_ring Acquire_sword - goodmotive
  Conquer Acquire_ring Destroy_sword - evilmotive

)
(:init

; MOTIVOS
(want-motive Destroy_ring Ring Destroyed)
(want-motive Acquire_ring Ring Acquired)
(want-motive Acquire_sword AndurilSword Acquired)
(want-motive Destroy_sword AndurilSword Destroyed)

(implies-that Peace Destroy_ring)
; Para conseguir la paz se debe destruir el anillo
(implies-that Peace Acquire_sword)
; Para matar a Sauron se debe tener la espada
(implies-that Conquer Acquire_ring)
; Para dominar la tierra media, se debe conseguir el anillo
```




```
(implies-that Conquer Destroy_sword)
; Para evitar que maten a Sauron, deben destruirla

; ANILLO
(magic-invisible Ring)
(can-be-destroyed-in Ring Mount_Doom)

; ESPADA ANDURIL
(at AndurilSword Gondor)
(strong AndurilSword)
(can-be-destroyed-in AndurilSword Mordor)

; BILBO
(at Bilbo Rivendel)
(have-motive Bilbo Peace)
(wish-an-obj Bilbo Ring) ; Aunque es bueno, desea el anillo
(have-an-obj Bilbo LittleElfSword)
(weak Bilbo)

; FRODO
(at Frodo La_Comarca)
(have-an-obj Frodo Ring)
(have-motive Frodo Peace)
(friends Bilbo Frodo)
(weak Frodo)

; GANDALF
(at Gandalf La_Comarca)
(have-motive Gandalf Peace)
(have-motive Gandalf Destroy_ring)
(have-an-obj Gandalf GandalfStaff)
(magic-kill GandalfStaff)
(can-use-magic-kill Gandalf)
(can-use-magic-heal Gandalf)
(friends Gandalf Bilbo)
(friends Gandalf Frodo)

; SAM
(at Sam La_Comarca)
(have-motive Sam Peace)
(friends Sam Frodo)
(weak Sam)

; MERRY Y PIPPIN
(at Merry La_Comarca)
(have-motive Merry Peace)
(at Pippin La_Comarca)
(have-motive Pippin Peace)
(weak Merry)
(weak Pippin)
```

(friends Merry Pippin)
(friends Merry Frodo) ; Son primos
(friends Pippin Frodo) ; Son primos

; GOLLUM
(at Gollum Mordor) ; Le estan torturando
(have-motive Gollum Peace)
; No es malo, solo quiere estar tranquilo con el anillo
(have-motive Gollum Acquire_ring)
; La gracia de tener tanto un motivo bueno como uno malo
; es la bipolaridad, como se muestra en los libros y las peliculas
; a veces quiere ayudar a Frodo, otras matarlo
; todo con el fin de conseguir el anillo
; Cambiara el estado de friends a enemies y enemies a friends a placer
(enemies Gollum Bilbo)
; Conoce a Bilbo y le odia por robarle el anillo
(enemies Gollum Sauron) ; Odia a Sauron por torturarlo
(wish-an-obj Gollum Ring)
(weak Gollum)

; SAURON
(at Sauron Mordor)
(isNPC Sauron) ; No puede moverse, no es un personaje en si
(have-motive Sauron Conquer)
(have-motive Sauron Acquire_ring)
(wish-an-obj Sauron Ring)
(strong Sauron)

; NAZGULS
(at Nazgul Mordor)
(have-motive Nazgul Conquer)
(have-an-obj Nazgul NazgulSword)
(friends Nazgul Sauron)
(wish-an-obj Nazgul Ring)

; SARUMAN
(at Saruman Isengard)
(isNPC Saruman)
; Lo dejaremos como NPC ya que nunca saldra de Isengard
(have-motive Saruman Conquer)
(have-an-obj Saruman SarumanStaff)
(magic-kill SarumanStaff)
(can-use-magic-kill Saruman)
(can-use-magic-heal Saruman)
(friends Saruman Sauron)
(friends Saruman Gandalf)

; ARAGORN
(at Aragorn Bree)
(have-motive Aragorn Peace)



```
(have-an-obj Aragorn AragornSword)
(friends Gandalf Aragorn)

; ORCOS
(at Orc1 Isengard)
(have-motive Orc1 Conquer)
(have-an-obj Orc1 Orc1Sword)
(friends Orc1 Saruman)

(at Orc2 Isengard)
(have-motive Orc2 Conquer)
(have-an-obj Orc2 Orc2Sword)
(friends Orc2 Saruman)

(at Orc3 Isengard)
(have-motive Orc3 Conquer)
(have-an-obj Orc3 Orc3Sword)
(friends Orc3 Saruman)

(friends Orc1 Orc2)
(friends Orc1 Orc3)
(friends Orc2 Orc3)

; ELROND
(at Elrond Rivendel)
(have-motive Elrond Peace)
(can-use-magic-heal Elrond)
(have-an-obj Elrond ElfMedicine)
(magic-heal ElfMedicine)
(friends Elrond Gandalf)
(friends Elrond Aragorn)

; LEGOLAS
(at Legolas Rivendel)
(have-motive Legolas Peace)
(have-an-obj Legolas LegolasBow)
(friends Elrond Legolas)
(friends Legolas Aragorn)

; GIMLI
(at Gimli Rivendel)
(have-motive Gimli Peace)
(have-an-obj Gimli GimliAxe)
(friends Legolas Gimli)
(friends Gimli Aragorn)

; BOROMIR
(at Boromir Rivendel)
(have-motive Boromir Peace)
(have-an-obj Boromir BoromirSword)
(wish-an-obj Boromir Ring)
```



```
; Aunque es bueno, desea el anillo

; ANIMALES
(at Rabbit1 Rohan)
(at Rabbit2 Rohan)
(at Wolf1 Rohan)
(at Wolf2 Gondor)

)
(:goal (and
      (is Ring Destroyed)
      (dead Sauron)
    )
)
)
```



Problema P2.PDDL

(define (problem p2) ; Problema EL SEñOR DE LOS ANILLOS - Dificultad
(2/5)

(:domain narrative)

(:objects

La_Comarca Rohan Rivendel Mordor Gondor Mount_Doom Moria
Bree Isengard - place
Bilbo Frodo Sam Merry Pippin - character ; Los Hobbits
Aragorn Legolas Gimli Gandalf Boromir Elrond - character
; Humanos, elfos y enanos
Saruman Sauron Gollum - character
; Los malos (gollum es un caso especial)
Nazgul - character
Orc1 Orc2 Orc3 - character
Rabbit1 Rabbit2 - edible
Wolf1 Wolf2 - dangerous

Ring - usefulobj
; Un anillo para gobernarlos a todos,
; un anillo para encontrarlos,
; un anillo para atraerlos a todos
; y atarlos en las tinieblas.

AndurilSword - arm
; La espada creada a partir de los trozos rotos
; de la espada Narsil,
; que corto el anillo del dedo de Sauron

GandalfStaff SarumanStaff ElfMedicine - usefulobj
NazgulSword - arm
Orc1Sword Orc2Sword Orc3Sword - arm
AragornSword LegolasBow GimliAxe BoromirSword - arm
LittleElfSword - arm
Peace Destroy_ring Acquire_sword - goodmotive
Conquer Acquire_ring Destroy_sword - evilmotive

)

(:init

; MOTIVOS

(want-motive Destroy_ring Ring Destroyed)
(want-motive Acquire_ring Ring Acquired)
(want-motive Acquire_sword AndurilSword Acquired)
(want-motive Destroy_sword AndurilSword Destroyed)

(implies-that Peace Destroy_ring)
; Para conseguir la paz se debe destruir el anillo
(implies-that Peace Acquire_sword)
; Para matar a Sauron se debe tener la espada
(implies-that Conquer Acquire_ring)
; Para dominar la tierra media, se debe conseguir el anillo



```
(implies-that Conquer Destroy_sword)
; Para evitar que maten a Sauron, deben destruirla

; ANILLO
(magic-invisible Ring)
(can-be-destroyed-in Ring Mount_Doom)

; ESPADA ANDURIL
(at AndurilSword Gondor)
(strong AndurilSword)
(can-be-destroyed-in AndurilSword Mordor)

; BILBO
(at Bilbo Rivendel)
(have-motive Bilbo Peace)
(wish-an-obj Bilbo Ring) ; Aunque es bueno, desea el anillo
(have-an-obj Bilbo LittleElfSword)
(weak Bilbo)

; FRODO
(at Frodo La_Comarca)
(have-an-obj Frodo Ring)
(have-motive Frodo Peace)
(friends Bilbo Frodo)
(weak Frodo)

; GANDALF
(at Gandalf La_Comarca)
(have-motive Gandalf Peace)
(have-motive Gandalf Destroy_ring)
(have-an-obj Gandalf GandalfStaff)
(magic-kill GandalfStaff)
(can-use-magic-kill Gandalf)
(can-use-magic-heal Gandalf)
(friends Gandalf Bilbo)
(friends Gandalf Frodo)

; SAM
(at Sam La_Comarca)
(have-motive Sam Peace)
(friends Sam Frodo)
(weak Sam)

; MERRY Y PIPPIN
(at Merry La_Comarca)
(have-motive Merry Peace)
(at Pippin La_Comarca)
(have-motive Pippin Peace)
(weak Merry)
(weak Pippin)
```

(friends Merry Pippin)
(friends Merry Frodo) ; Son primos
(friends Pippin Frodo) ; Son primos

; GOLLUM
(at Gollum Mordor) ; Le estan torturando
(have-motive Gollum Peace)
; No es malo, solo quiere estar tranquilo con el anillo
(have-motive Gollum Acquire_ring)
; La gracia de tener tanto un motivo bueno como uno malo
; es la bipolaridad, como se muestra en los libros y las peliculas
; a veces quiere ayudar a Frodo, otras matarlo
; todo con el fin de conseguir el anillo
; Cambiara el estado de friends a enemies y enemies a friends a placer
(enemies Gollum Bilbo)
; Conoce a Bilbo y le odia por robarle el anillo
(enemies Gollum Sauron) ; Odia a Sauron por torturarlo
(wish-an-obj Gollum Ring)
(weak Gollum)

; SAURON
(at Sauron Mordor)
(isNPC Sauron) ; No puede moverse, no es un personaje en si
(have-motive Sauron Conquer)
(have-motive Sauron Acquire_ring)
(wish-an-obj Sauron Ring)
(strong Sauron)

; NAZGULS
(at Nazgul Mordor)
(have-motive Nazgul Conquer)
(have-an-obj Nazgul NazgulSword)
(friends Nazgul Sauron)
(wish-an-obj Nazgul Ring)

; SARUMAN
(at Saruman Isengard)
(isNPC Saruman)
; Lo dejaremos como NPC ya que nunca saldra de Isengard
(have-motive Saruman Conquer)
(have-an-obj Saruman SarumanStaff)
(magic-kill SarumanStaff)
(can-use-magic-kill Saruman)
(can-use-magic-heal Saruman)
(friends Saruman Sauron)
(friends Saruman Gandalf)

; ARAGORN
(at Aragorn Bree)
(have-motive Aragorn Peace)



```
(have-an-obj Aragorn AragornSword)
(friends Gandalf Aragorn)

; ORCOS
(at Orc1 Isengard)
(have-motive Orc1 Conquer)
(have-an-obj Orc1 Orc1Sword)
(friends Orc1 Saruman)

(at Orc2 Isengard)
(have-motive Orc2 Conquer)
(have-an-obj Orc2 Orc2Sword)
(friends Orc2 Saruman)

(at Orc3 Isengard)
(have-motive Orc3 Conquer)
(have-an-obj Orc3 Orc3Sword)
(friends Orc3 Saruman)

(friends Orc1 Orc2)
(friends Orc1 Orc3)
(friends Orc2 Orc3)

; ELROND
(at Elrond Rivendel)
(have-motive Elrond Peace)
(can-use-magic-heal Elrond)
(have-an-obj Elrond ElfMedicine)
(magic-heal ElfMedicine)
(friends Elrond Gandalf)
(friends Elrond Aragorn)

; LEGOLAS
(at Legolas Rivendel)
(have-motive Legolas Peace)
(have-an-obj Legolas LegolasBow)
(friends Elrond Legolas)
(friends Legolas Aragorn)

; GIMLI
(at Gimli Rivendel)
(have-motive Gimli Peace)
(have-an-obj Gimli GimliAxe)
(friends Legolas Gimli)
(friends Gimli Aragorn)

; BOROMIR
(at Boromir Rivendel)
(have-motive Boromir Peace)
(have-an-obj Boromir BoromirSword)
(wish-an-obj Boromir Ring)
```




```
    ; Aunque es bueno, desea el anillo

    ; ANIMALES
    (at Rabbit1 Rohan)
    (at Rabbit2 Rohan)
    (at Wolf1 Rohan)
    (at Wolf2 Gondor)

)
(:goal (and
      (dead Boromir)
      (dead Gandalf)
      (is AndurilSword Destroyed)
      (motive-complete Sauron Acquire_ring)
    )
)
)
```



Problema P3.PDDL

```
(define (problem p3) ; Problema EL SEñOR DE LOS ANILLOS - Dificultad
(3/5)
(:domain narrative)
(:objects
  La_Comarca Rohan Rivendel Mordor Gondor Mount_Doom Moria
  Bree Isengard - place
  Bilbo Frodo Sam Merry Pippin - character ; Los Hobbits
  Aragorn Legolas Gimli Gandalf Boromir Elrond - character
  ; Humanos, elfos y enanos
  Saruman Sauron Gollum - character
  ; Los malos (gollum es un caso especial)
  Nazgul - character
  Orc1 Orc2 Orc3 - character
  Rabbit1 Rabbit2 - edible
  Wolf1 Wolf2 - dangerous

  Ring - usefulobj
  ; Un anillo para gobernarlos a todos,
  ; un anillo para encontrarlos,
  ; un anillo para atraerlos a todos
  ; y atarlos en las tinieblas.

  AndurilSword - arm
  ; La espada creada a partir de los trozos rotos
  ; de la espada Narsil,
  ;que corto el anillo del dedo de Sauron

  GandalfStaff SarumanStaff ElfMedicine - usefulobj
  NazgulSword - arm
  Orc1Sword Orc2Sword Orc3Sword - arm
  AragornSword LegolasBow GimliAxe BoromirSword - arm
  LittleElfSword - arm
  Peace Destroy_ring Acquire_sword - goodmotive
  Conquer Acquire_ring Destroy_sword - evilmotive

)
(:init

; MOTIVOS
(want-motive Destroy_ring Ring Destroyed)
(want-motive Acquire_ring Ring Acquired)
(want-motive Acquire_sword AndurilSword Acquired)
(want-motive Destroy_sword AndurilSword Destroyed)

(implies-that Peace Destroy_ring)
; Para conseguir la paz se debe destruir el anillo
(implies-that Peace Acquire_sword)
; Para matar a Sauron se debe tener la espada
(implies-that Conquer Acquire_ring)
; Para dominar la tierra media, se debe conseguir el anillo
```

```
(implies-that Conquer Destroy_sword)
; Para evitar que maten a Sauron, deben destruirla

; ANILLO
(magic-invisible Ring)
(can-be-destroyed-in Ring Mount_Doom)

; ESPADA ANDURIL
(at AndurilSword Gondor)
(strong AndurilSword)
(can-be-destroyed-in AndurilSword Mordor)

; BILBO
(at Bilbo Rivendel)
(have-motive Bilbo Peace)
(wish-an-obj Bilbo Ring) ; Aunque es bueno, desea el anillo
(have-an-obj Bilbo LittleElfSword)
(weak Bilbo)

; FRODO
(at Frodo La_Comarca)
(have-an-obj Frodo Ring)
(have-motive Frodo Peace)
(friends Bilbo Frodo)
(weak Frodo)

; GANDALF
(at Gandalf La_Comarca)
(have-motive Gandalf Peace)
(have-motive Gandalf Destroy_ring)
(have-an-obj Gandalf GandalfStaff)
(magic-kill GandalfStaff)
(can-use-magic-kill Gandalf)
(can-use-magic-heal Gandalf)
(friends Gandalf Bilbo)
(friends Gandalf Frodo)

; SAM
(at Sam La_Comarca)
(have-motive Sam Peace)
(friends Sam Frodo)
(weak Sam)

; MERRY Y PIPPIN
(at Merry La_Comarca)
(have-motive Merry Peace)
(at Pippin La_Comarca)
(have-motive Pippin Peace)
(weak Merry)
(weak Pippin)
```

(friends Merry Pippin)
(friends Merry Frodo) ; Son primos
(friends Pippin Frodo) ; Son primos

; GOLLUM
(at Gollum Mordor) ; Le estan torturando
(have-motive Gollum Peace)
; No es malo, solo quiere estar tranquilo con el anillo
(have-motive Gollum Acquire_ring)
; La gracia de tener tanto un motivo bueno como uno malo
; es la bipolaridad, como se muestra en los libros y las peliculas
; a veces quiere ayudar a Frodo, otras matarlo
; todo con el fin de conseguir el anillo
; Cambiara el estado de friends a enemies y enemies a friends a placer
(enemies Gollum Bilbo)
; Conoce a Bilbo y le odia por robarle el anillo
(enemies Gollum Sauron) ; Odia a Sauron por torturarlo
(wish-an-obj Gollum Ring)
(weak Gollum)

; SAURON
(at Sauron Mordor)
(isNPC Sauron) ; No puede moverse, no es un personaje en si
(have-motive Sauron Conquer)
(have-motive Sauron Acquire_ring)
(wish-an-obj Sauron Ring)
(strong Sauron)

; NAZGULS
(at Nazgul Mordor)
(have-motive Nazgul Conquer)
(have-an-obj Nazgul NazgulSword)
(friends Nazgul Sauron)
(wish-an-obj Nazgul Ring)

; SARUMAN
(at Saruman Isengard)
(isNPC Saruman)
; Lo dejaremos como NPC ya que nunca saldra de Isengard
(have-motive Saruman Conquer)
(have-an-obj Saruman SarumanStaff)
(magic-kill SarumanStaff)
(can-use-magic-kill Saruman)
(can-use-magic-heal Saruman)
(friends Saruman Sauron)
(friends Saruman Gandalf)

; ARAGORN
(at Aragorn Bree)
(have-motive Aragorn Peace)

```
(have-an-obj Aragorn AragornSword)
(friends Gandalf Aragorn)

; ORCOS
(at Orc1 Isengard)
(have-motive Orc1 Conquer)
(have-an-obj Orc1 Orc1Sword)
(friends Orc1 Saruman)

(at Orc2 Isengard)
(have-motive Orc2 Conquer)
(have-an-obj Orc2 Orc2Sword)
(friends Orc2 Saruman)

(at Orc3 Isengard)
(have-motive Orc3 Conquer)
(have-an-obj Orc3 Orc3Sword)
(friends Orc3 Saruman)

(friends Orc1 Orc2)
(friends Orc1 Orc3)
(friends Orc2 Orc3)

; ELROND
(at Elrond Rivendel)
(have-motive Elrond Peace)
(can-use-magic-heal Elrond)
(have-an-obj Elrond ElfMedicine)
(magic-heal ElfMedicine)
(friends Elrond Gandalf)
(friends Elrond Aragorn)

; LEGOLAS
(at Legolas Rivendel)
(have-motive Legolas Peace)
(have-an-obj Legolas LegolasBow)
(friends Elrond Legolas)
(friends Legolas Aragorn)

; GIMLI
(at Gimli Rivendel)
(have-motive Gimli Peace)
(have-an-obj Gimli GimliAxe)
(friends Legolas Gimli)
(friends Gimli Aragorn)

; BOROMIR
(at Boromir Rivendel)
(have-motive Boromir Peace)
(have-an-obj Boromir BoromirSword)
(wish-an-obj Boromir Ring)
```



```
; Aunque es bueno, desea el anillo

; ANIMALES
(at Rabbit1 Rohan)
(at Rabbit2 Rohan)
(at Wolf1 Rohan)
(at Wolf2 Gondor)

)
(:goal (and
      (dead Boromir)
      (dead Nazgul)
      (dead Saruman)
      (enemies Frodo Sam)
      (friends Gollum Aragorn)
      (is AndurilSword Destroyed)
      (is Ring Destroyed)
    )
)
)
```



Problema P4.PDDL

```
(define (problem p4) ; Problema EL SEñOR DE LOS ANILLOS - Dificultad
(4/5)
(:domain narrative)
(:objects
  La_Comarca Rohan Rivendel Mordor Gondor Mount_Doom Moria
  Bree Isengard - place
  Bilbo Frodo Sam Merry Pippin - character ; Los Hobbits
  Aragorn Legolas Gimli Gandalf Boromir Elrond - character
  ; Humanos, elfos y enanos
  Saruman Sauron Gollum - character
  ; Los malos (gollum es un caso especial)
  Nazgul - character
  Orc1 Orc2 Orc3 - character
  Rabbit1 Rabbit2 - edible
  Wolf1 Wolf2 - dangerous

  Ring - usefulobj
  ; Un anillo para gobernarlos a todos,
  ; un anillo para encontrarlos,
  ; un anillo para atraerlos a todos
  ; y atarlos en las tinieblas.

  AndurilSword - arm
  ; La espada creada a partir de los trozos rotos
  ; de la espada Narsil,
  ;que corto el anillo del dedo de Sauron

  GandalfStaff SarumanStaff ElfMedicine - usefulobj
  NazgulSword - arm
  Orc1Sword Orc2Sword Orc3Sword - arm
  AragornSword LegolasBow GimliAxe BoromirSword - arm
  LittleElfSword - arm
  Peace Destroy_ring Acquire_sword - goodmotive
  Conquer Acquire_ring Destroy_sword - evilmotive

)
(:init

; MOTIVOS
(want-motive Destroy_ring Ring Destroyed)
(want-motive Acquire_ring Ring Acquired)
(want-motive Acquire_sword AndurilSword Acquired)
(want-motive Destroy_sword AndurilSword Destroyed)

(implies-that Peace Destroy_ring)
; Para conseguir la paz se debe destruir el anillo
(implies-that Peace Acquire_sword)
; Para matar a Sauron se debe tener la espada
(implies-that Conquer Acquire_ring)
; Para dominar la tierra media, se debe conseguir el anillo
```



```
(implies-that Conquer Destroy_sword)
; Para evitar que maten a Sauron, deben destruirla

; ANILLO
(magic-invisible Ring)
(can-be-destroyed-in Ring Mount_Doom)

; ESPADA ANDURIL
(at AndurilSword Gondor)
(strong AndurilSword)
(can-be-destroyed-in AndurilSword Mordor)

; BILBO
(at Bilbo Rivendel)
(have-motive Bilbo Peace)
(wish-an-obj Bilbo Ring) ; Aunque es bueno, desea el anillo
(have-an-obj Bilbo LittleElfSword)
(weak Bilbo)

; FRODO
(at Frodo La_Comarca)
(have-an-obj Frodo Ring)
(have-motive Frodo Peace)
(friends Bilbo Frodo)
(weak Frodo)

; GANDALF
(at Gandalf La_Comarca)
(have-motive Gandalf Peace)
(have-motive Gandalf Destroy_ring)
(have-an-obj Gandalf GandalfStaff)
(magic-kill GandalfStaff)
(can-use-magic-kill Gandalf)
(can-use-magic-heal Gandalf)
(friends Gandalf Bilbo)
(friends Gandalf Frodo)

; SAM
(at Sam La_Comarca)
(have-motive Sam Peace)
(friends Sam Frodo)
(weak Sam)

; MERRY Y PIPPIN
(at Merry La_Comarca)
(have-motive Merry Peace)
(at Pippin La_Comarca)
(have-motive Pippin Peace)
(weak Merry)
(weak Pippin)
```




(friends Merry Pippin)
(friends Merry Frodo) ; Son primos
(friends Pippin Frodo) ; Son primos

; GOLLUM
(at Gollum Mordor) ; Le estan torturando
(have-motive Gollum Peace)
; No es malo, solo quiere estar tranquilo con el anillo
(have-motive Gollum Acquire_ring)
; La gracia de tener tanto un motivo bueno como uno malo
; es la bipolaridad, como se muestra en los libros y las peliculas
; a veces quiere ayudar a Frodo, otras matarlo
; todo con el fin de conseguir el anillo
; Cambiara el estado de friends a enemies y enemies a friends a placer
(enemies Gollum Bilbo)
; Conoce a Bilbo y le odia por robarle el anillo
(enemies Gollum Sauron) ; Odia a Sauron por torturarlo
(wish-an-obj Gollum Ring)
(weak Gollum)

; SAURON
(at Sauron Mordor)
(isNPC Sauron) ; No puede moverse, no es un personaje en si
(have-motive Sauron Conquer)
(have-motive Sauron Acquire_ring)
(wish-an-obj Sauron Ring)
(strong Sauron)

; NAZGULS
(at Nazgul Mordor)
(have-motive Nazgul Conquer)
(have-an-obj Nazgul NazgulSword)
(friends Nazgul Sauron)
(wish-an-obj Nazgul Ring)

; SARUMAN
(at Saruman Isengard)
(isNPC Saruman)
; Lo dejaremos como NPC ya que nunca saldra de Isengard
(have-motive Saruman Conquer)
(have-an-obj Saruman SarumanStaff)
(magic-kill SarumanStaff)
(can-use-magic-kill Saruman)
(can-use-magic-heal Saruman)
(friends Saruman Sauron)
(friends Saruman Gandalf)

; ARAGORN
(at Aragorn Bree)
(have-motive Aragorn Peace)



```
(have-an-obj Aragorn AragornSword)
(friends Gandalf Aragorn)

; ORCOS
(at Orc1 Isengard)
(have-motive Orc1 Conquer)
(have-an-obj Orc1 Orc1Sword)
(friends Orc1 Saruman)

(at Orc2 Isengard)
(have-motive Orc2 Conquer)
(have-an-obj Orc2 Orc2Sword)
(friends Orc2 Saruman)

(at Orc3 Isengard)
(have-motive Orc3 Conquer)
(have-an-obj Orc3 Orc3Sword)
(friends Orc3 Saruman)

(friends Orc1 Orc2)
(friends Orc1 Orc3)
(friends Orc2 Orc3)

; ELROND
(at Elrond Rivendel)
(have-motive Elrond Peace)
(can-use-magic-heal Elrond)
(have-an-obj Elrond ElfMedicine)
(magic-heal ElfMedicine)
(friends Elrond Gandalf)
(friends Elrond Aragorn)

; LEGOLAS
(at Legolas Rivendel)
(have-motive Legolas Peace)
(have-an-obj Legolas LegolasBow)
(friends Elrond Legolas)
(friends Legolas Aragorn)

; GIMLI
(at Gimli Rivendel)
(have-motive Gimli Peace)
(have-an-obj Gimli GimliAxe)
(friends Legolas Gimli)
(friends Gimli Aragorn)

; BOROMIR
(at Boromir Rivendel)
(have-motive Boromir Peace)
(have-an-obj Boromir BoromirSword)
(wish-an-obj Boromir Ring)
```



```
; Aunque es bueno, desea el anillo

; ANIMALES
(at Rabbit1 Rohan)
(at Rabbit2 Rohan)
(at Wolf1 Rohan)
(at Wolf2 Gondor)

)
(:goal (and
  (dead Boromir)
  (dead Orc1)
  (dead Orc2)
  (dead Rabbit1)
  (enemies Frodo Sam)
  (friends Gollum Aragorn)
  (killed-by Saruman Legolas)
  (killed-by Orc3 Frodo)
  (is Ring Acquired)
  (dead Sauron)
)
)
)
```

Problema P5.PDDL

```
(define (problem p5) ; Problema EL SEñOR DE LOS ANILLOS - Dificultad
(5/5)
(:domain narrative)
(:objects
  La_Comarca Rohan Rivendel Mordor Gondor Mount_Doom Moria
  Bree Isengard - place
  Bilbo Frodo Sam Merry Pippin - character ; Los Hobbits
  Aragorn Legolas Gimli Gandalf Boromir Elrond - character
  ; Humanos, elfos y enanos
  Saruman Sauron Gollum - character
  ; Los malos (gollum es un caso especial)
  Nazgul - character
  Orc1 Orc2 Orc3 - character
  Rabbit1 Rabbit2 - edible
  Wolf1 Wolf2 - dangerous

  Ring - usefulobj
  ; Un anillo para gobernarlos a todos,
  ; un anillo para encontrarlos,
  ; un anillo para atraerlos a todos
  ; y atarlos en las tinieblas.

  AndurilSword - arm
  ; La espada creada a partir de los trozos rotos
  ; de la espada Narsil,
  ;que corto el anillo del dedo de Sauron

  GandalfStaff SarumanStaff ElfMedicine - usefulobj
  NazgulSword - arm
  Orc1Sword Orc2Sword Orc3Sword - arm
  AragornSword LegolasBow GimliAxe BoromirSword - arm
  LittleElfSword - arm
  Peace Destroy_ring Acquire_sword - goodmotive
  Conquer Acquire_ring Destroy_sword - evilmotive

)
(:init

; MOTIVOS
(want-motive Destroy_ring Ring Destroyed)
(want-motive Acquire_ring Ring Acquired)
(want-motive Acquire_sword AndurilSword Acquired)
(want-motive Destroy_sword AndurilSword Destroyed)

(implies-that Peace Destroy_ring)
; Para conseguir la paz se debe destruir el anillo
(implies-that Peace Acquire_sword)
; Para matar a Sauron se debe tener la espada
(implies-that Conquer Acquire_ring)
; Para dominar la tierra media, se debe conseguir el anillo
```



```
(implies-that Conquer Destroy_sword)
; Para evitar que maten a Sauron, deben destruirla

; ANILLO
(magic-invisible Ring)
(can-be-destroyed-in Ring Mount_Doom)

; ESPADA ANDURIL
(at AndurilSword Gondor)
(strong AndurilSword)
(can-be-destroyed-in AndurilSword Mordor)

; BILBO
(at Bilbo Rivendel)
(have-motive Bilbo Peace)
(wish-an-obj Bilbo Ring) ; Aunque es bueno, desea el anillo
(have-an-obj Bilbo LittleElfSword)
(weak Bilbo)

; FRODO
(at Frodo La_Comarca)
(have-an-obj Frodo Ring)
(have-motive Frodo Peace)
(friends Bilbo Frodo)
(weak Frodo)

; GANDALF
(at Gandalf La_Comarca)
(have-motive Gandalf Peace)
(have-motive Gandalf Destroy_ring)
(have-an-obj Gandalf GandalfStaff)
(magic-kill GandalfStaff)
(can-use-magic-kill Gandalf)
(can-use-magic-heal Gandalf)
(friends Gandalf Bilbo)
(friends Gandalf Frodo)

; SAM
(at Sam La_Comarca)
(have-motive Sam Peace)
(friends Sam Frodo)
(weak Sam)

; MERRY Y PIPPIN
(at Merry La_Comarca)
(have-motive Merry Peace)
(at Pippin La_Comarca)
(have-motive Pippin Peace)
(weak Merry)
(weak Pippin)
```

(friends Merry Pippin)
(friends Merry Frodo) ; Son primos
(friends Pippin Frodo) ; Son primos

; GOLLUM
(at Gollum Mordor) ; Le estan torturando
(have-motive Gollum Peace)
; No es malo, solo quiere estar tranquilo con el anillo
(have-motive Gollum Acquire_ring)
; La gracia de tener tanto un motivo bueno como uno malo
; es la bipolaridad, como se muestra en los libros y las peliculas
; a veces quiere ayudar a Frodo, otras matarlo
; todo con el fin de conseguir el anillo
; Cambiara el estado de friends a enemies y enemies a friends a placer
(enemies Gollum Bilbo)
; Conoce a Bilbo y le odia por robarle el anillo
(enemies Gollum Sauron) ; Odia a Sauron por torturarlo
(wish-an-obj Gollum Ring)
(weak Gollum)

; SAURON
(at Sauron Mordor)
(isNPC Sauron) ; No puede moverse, no es un personaje en si
(have-motive Sauron Conquer)
(have-motive Sauron Acquire_ring)
(wish-an-obj Sauron Ring)
(strong Sauron)

; NAZGULS
(at Nazgul Mordor)
(have-motive Nazgul Conquer)
(have-an-obj Nazgul NazgulSword)
(friends Nazgul Sauron)
(wish-an-obj Nazgul Ring)

; SARUMAN
(at Saruman Isengard)
(isNPC Saruman)
; Lo dejaremos como NPC ya que nunca saldra de Isengard
(have-motive Saruman Conquer)
(have-an-obj Saruman SarumanStaff)
(magic-kill SarumanStaff)
(can-use-magic-kill Saruman)
(can-use-magic-heal Saruman)
(friends Saruman Sauron)
(friends Saruman Gandalf)

; ARAGORN
(at Aragorn Bree)
(have-motive Aragorn Peace)



```
(have-an-obj Aragorn AragornSword)
(friends Gandalf Aragorn)

; ORCOS
(at Orc1 Isengard)
(have-motive Orc1 Conquer)
(have-an-obj Orc1 Orc1Sword)
(friends Orc1 Saruman)

(at Orc2 Isengard)
(have-motive Orc2 Conquer)
(have-an-obj Orc2 Orc2Sword)
(friends Orc2 Saruman)

(at Orc3 Isengard)
(have-motive Orc3 Conquer)
(have-an-obj Orc3 Orc3Sword)
(friends Orc3 Saruman)

(friends Orc1 Orc2)
(friends Orc1 Orc3)
(friends Orc2 Orc3)

; ELROND
(at Elrond Rivendel)
(have-motive Elrond Peace)
(can-use-magic-heal Elrond)
(have-an-obj Elrond ElfMedicine)
(magic-heal ElfMedicine)
(friends Elrond Gandalf)
(friends Elrond Aragorn)

; LEGOLAS
(at Legolas Rivendel)
(have-motive Legolas Peace)
(have-an-obj Legolas LegolasBow)
(friends Elrond Legolas)
(friends Legolas Aragorn)

; GIMLI
(at Gimli Rivendel)
(have-motive Gimli Peace)
(have-an-obj Gimli GimliAxe)
(friends Legolas Gimli)
(friends Gimli Aragorn)

; BOROMIR
(at Boromir Rivendel)
(have-motive Boromir Peace)
(have-an-obj Boromir BoromirSword)
(wish-an-obj Boromir Ring)
```



```
; Aunque es bueno, desea el anillo

; ANIMALES
(at Rabbit1 Rohan)
(at Rabbit2 Rohan)
(at Wolf1 Rohan)
(at Wolf2 Gondor)

)
(:goal (and
  (dead Boromir)
  (have-an-obj Frodo LittleElfSword)
  (dead Orc1)
  (dead Orc2)
  (dead Orc3)
  (killed-by Nazgul Aragorn)
  (killed-by Saruman Legolas)
  (hurt Frodo)
  (hurt Gimli)
  (have-an-obj Sam AragornSword)
  (dead Gollum)
  (dead Rabbit1)
  (dead Wolf2)
  (is Ring Destroyed)
  (dead Sauron)
)
)
)
```


Anexo III: Resultados

Problema P1.PDDL

Problema P1.PDDL (0.1)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (ACQUIRE_RING) are enemies because their motives disagree
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Problema P1.PDDL (0.3)

ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN (ACQUIRE_SWORD) and SAURON (ACQUIRE_RING) are enemies because their motives disagree
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN kills SAURON at MORDOR with ANDURILSWORD
FRODO destroys the object RING at MOUNT_DOOM



Problema P1.PDDL (0.5)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (ACQUIRE_RING) are enemies because their motives disagree
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Problema P1.PDDL (0.7)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Problema P1.PDDL (0.9)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN kills SAURON at MORDOR with ANDURILSWORD



Problema P2.PDDL

Problema P2.PDDL (0.1)

GANDALF (DESTROY_RING) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SAURON and GOLLUM are friends because they have the motive ACQUIRE_RING in common
GANDALF wishes the object RING because he has the motive DESTROY_RING
FRODO (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
FRODO travels to ISENGARD because he wants to give the object RING to GANDALF
FRODO gives the object RING to GANDALF at ISENGARD
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
GOLLUM loots the object RING from GANDALF (dead) at ISENGARD
GOLLUM (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GOLLUM travels to MORDOR because he wants to give the object RING to SAURON
GOLLUM (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
NAZGUL takes the object ANDURILSWORD at GONDOR
GOLLUM (PEACE) and NAZGUL (DESTROY_SWORD) are enemies because their motives disagree
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
SAURON punches and hurts GOLLUM at MORDOR
NAZGUL destroys the object ANDURILSWORD at MORDOR
NAZGUL kills GOLLUM at MORDOR with NAZGULSWORD
SAURON loots the object RING from GOLLUM (dead) at MORDOR
SAURON has acquired the object RING



Problema P2.PDDL (0.3)

GANDALF (DESTROY_RING) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
SAURON and GOLLUM are friends because they have the motive ACQUIRE_RING in common
FRODO gives the object RING to GANDALF at LA_COMARCA
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
GOLLUM asks his friend SAURON to follow his motive PEACE
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
GANDALF travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
NAZGUL destroys the object ANDURILSWORD at MORDOR
GOLLUM travels to ISENGARD because he wishes the object RING GANDALF has
SARUMAN kills (magic) GOLLUM at ISENGARD with SARUMANSTAFF
NAZGUL travels to ISENGARD because he wishes the object RING GANDALF has
SAURON (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ORC1 loots the object RING from GANDALF (dead) at ISENGARD
ORC1 travels to MORDOR because he wants to kill SAURON
SAURON and ORC1 are friends because they have the motive CONQUER in common
ORC1 gives the object RING to SAURON at MORDOR
SAURON has acquired the object RING



Problema P2.PDDL (0.5)

GANDALF (DESTROY_RING) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SAURON and GOLLUM are friends because they have the motive ACQUIRE_RING in common
FRODO gives the object RING to GANDALF at LA_COMARCA
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GOLLUM asks his friend SAURON to follow his motive PEACE
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
GANDALF travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
GOLLUM travels to ISENGARD because he wishes the object RING
GANDALF has
NAZGUL takes the object ANDURILSWORD at GONDOR
SARUMAN kills (magic) GOLLUM at ISENGARD with SARUMANSTAFF
WOLF2 hurts NAZGUL at GONDOR
WOLF2 kills NAZGUL at GONDOR
SAURON (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ORC1 has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
ORC1 loots the object RING from GANDALF (dead) at ISENGARD
ORC1 travels to MORDOR because he wants to kill SAURON
SAURON and ORC1 are friends because they have the motive CONQUER in common
ORC1 gives the object RING to SAURON at MORDOR
ORC1 wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
SAURON has acquired the object RING
ORC1 travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL has
ORC1 loots the object ANDURILSWORD from NAZGUL (dead) at GONDOR
ORC1 travels to MORDOR because he wishes to destroy the object ANDURILSWORD
ORC1 destroys the object ANDURILSWORD at MORDOR



Problema P2.PDDL (0.7)

GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
FRODO gives the object RING to GANDALF at LA_COMARCA
GANDALF (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF travels to MORDOR because he wants to kill SAURON
GANDALF travels to ISENGARD because he wants to kill SARUMAN
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
BOROMIR travels to ISENGARD because he wishes the object RING GANDALF has
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
NAZGUL destroys the object ANDURILSWORD at MORDOR
GOLLUM travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL loots the object RING from GANDALF (dead) at ISENGARD
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GOLLUM and NAZGUL are friends because they want to kill BILBO
NAZGUL and SARUMAN are enemies because GOLLUM (NAZGUL's friend) and SARUMAN are enemies
SARUMAN kills (magic) NAZGUL at ISENGARD with SARUMANSTAFF
ELROND (PEACE) and SAURON (ACQUIRE_RING) are enemies because their motives disagree
SARUMAN punches and hurts GOLLUM at ISENGARD
GOLLUM loots the object RING from NAZGUL (dead) at ISENGARD
SARUMAN kills (magic) GOLLUM at ISENGARD with SARUMANSTAFF
ORC1 loots the object RING from GOLLUM (dead) at ISENGARD
ELROND (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ELROND travels to MORDOR because he wants to kill SAURON
ORC1 travels to MORDOR because he wants to kill ELROND
ORC1 and SAURON are friends because they want to kill ELROND
ORC1 gives the object RING to SAURON at MORDOR
SAURON has acquired the object RING



Problema P2.PDDL (0.9)

GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
FRODO gives the object RING to GANDALF at LA_COMARCA
GANDALF (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
GANDALF travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wishes the object RING GANDALF has
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
NAZGUL destroys the object ANDURILSWORD at MORDOR
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM travels to ISENGARD because he wishes the object RING GANDALF has
BILBO travels to ISENGARD because he wishes the object RING GANDALF has
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
NAZGUL travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL loots the object RING from GANDALF (dead) at ISENGARD
BILBO hurts NAZGUL at ISENGARD with LITTLEELFSWORD
NAZGUL travels to MORDOR because he wants to give the object RING to SAURON
BILBO travels to MORDOR because he wishes the object RING NAZGUL has
BILBO kills NAZGUL at MORDOR with LITTLEELFSWORD
SAURON loots the object RING from NAZGUL (dead) at MORDOR
SAURON has acquired the object RING



Problema P3.PDDL

Problema P3.PDDL (0.1)

GANDALF (DESTROY_RING) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (ACQUIRE_RING) are enemies because their motives disagree
GANDALF (DESTROY_RING) and NAZGUL (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
GANDALF travels to ISENGARD because he wants to kill SARUMAN
GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF
ARAGORN travels to MORDOR because he wants to kill SAURON
BOROMIR (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GOLLUM travels to MOUNT_DOOM because he wishes the object RING FRODO has
BOROMIR travels to MORDOR because he wants to kill NAZGUL
ARAGORN (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
BOROMIR hurts NAZGUL at MORDOR with BOROMIRSWORD
FRODO and GOLLUM are friends because they have the motive PEACE in common
ARAGORN travels to MOUNT_DOOM because he wants to kill GOLLUM
NAZGUL punches and hurts BOROMIR at MORDOR
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are enemies
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
NAZGUL kills BOROMIR at MORDOR with NAZGULSWORD
FRODO destroys the object RING at MOUNT_DOOM
NAZGUL travels to ROHAN because he wants to eat a RABBIT1
WOLF1 kills NAZGUL at ROHAN
ARAGORN (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ORC1 has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
ARAGORN travels to MORDOR because he wants to kill SAURON
ORC1 travels to MORDOR because he wants to kill ARAGORN



ORC1 wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
ORC1 travels to ROHAN because he wishes the object ANDURILSWORD NAZGUL has
ORC1 loots the object ANDURILSWORD from NAZGUL (dead) at ROHAN
ORC1 travels to MORDOR because he wishes to destroy the object ANDURILSWORD
ORC1 destroys the object ANDURILSWORD at MORDOR

Problema P3.PDDL (0.3)

BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
GANDALF (DESTROY_RING) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
ARAGORN travels to MORDOR because he wants to kill SAURON
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
GANDALF (DESTROY_RING) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
GOLLUM travels to MOUNT_DOOM because he wishes the object RING FRODO has
NAZGUL travels to LA_COMARCA because he wants to kill GANDALF
FRODO and GOLLUM are friends because they have the motive PEACE in common
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN (PEACE) and NAZGUL (DESTROY_SWORD) are enemies because their motives disagree
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are enemies
NAZGUL takes the object ANDURILSWORD at GONDOR
GANDALF travels to ISENGARD because he wants to kill SARUMAN
WOLF2 hurts NAZGUL at GONDOR
WOLF2 kills NAZGUL at GONDOR
GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF
ARAGORN (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ORC1 has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
ORC1 travels to MORDOR because he wants to kill ARAGORN



ORC1 wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
ORC1 travels to GONDOR because he wishes the object ANDURILSWORD NAZGUL has
ORC1 loots the object ANDURILSWORD from NAZGUL (dead) at GONDOR
ORC1 travels to MORDOR because he wants to kill ARAGORN
ORC1 destroys the object ANDURILSWORD at MORDOR

Problema P3.PDDL (0.5)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL travels to LA_COMARCA because he wants to kill GANDALF
GOLLUM (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
SAM and NAZGUL are friends because they want to kill GOLLUM
NAZGUL asks his friend SAM to follow his motive CONQUER
FRODO (PEACE) and SAM (CONQUER) are enemies because their motives disagree
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
WOLF2 hurts NAZGUL at GONDOR
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
BOROMIR punches and hurts SARUMAN at ISENGARD
FRODO destroys the object RING at MOUNT_DOOM
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
ARAGORN (PEACE) and NAZGUL (DESTROY_SWORD) are enemies because their motives disagree
ARAGORN kills NAZGUL at MORDOR with ARAGORNSWORD
SAURON loots the object ANDURILSWORD from NAZGUL (dead) at MORDOR
SAURON has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
SAURON destroys the object ANDURILSWORD at MORDOR
GANDALF travels to ISENGARD because he wants to kill SARUMAN



GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF

Problema P3.PDDL (0.7)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL travels to LA_COMARCA because he wishes the object RING FRODO has
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
GANDALF (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF kills (magic) NAZGUL at LA_COMARCA with GANDALFSTAFF
SAURON has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
SAURON and GOLLUM are friends because they have the motive ACQUIRE_RING in common
SAURON asks his friend GOLLUM to follow his motive DESTROY_SWORD
GOLLUM wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
GOLLUM travels to GONDOR because he wishes the object ANDURILSWORD
GOLLUM takes the object ANDURILSWORD at GONDOR
GOLLUM travels to LA_COMARCA because he wishes the object RING FRODO has
FRODO and GOLLUM are friends because they have the motive PEACE in common
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are enemies
GANDALF travels to ISENGARD because he wants to kill SARUMAN
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
GOLLUM travels to MORDOR because he wishes to destroy the object ANDURILSWORD
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF
FRODO destroys the object RING at MOUNT_DOOM
GOLLUM destroys the object ANDURILSWORD at MORDOR



Problema P3.PDDL (0.9)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their
motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives
disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their
motives disagree
GANDALF (PEACE) and NAZGUL (CONQUER) are enemies because their motives
disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their
motives disagree
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in
common
NAZGUL has the motive DESTROY_SWORD because he wants to complete his
motive CONQUER
NAZGUL wishes the object ANDURILSWORD because he has the motive
DESTROY_SWORD
GOLLUM travels to LA_COMARCA because he wishes the object RING FRODO
has
FRODO and GOLLUM are friends because they have the motive PEACE in
common
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are
enemies
FRODO travels to MOUNT_DOOM because he wishes to destroy the object
RING
FRODO destroys the object RING at MOUNT_DOOM
GANDALF travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF
GANDALF travels to MORDOR because he wants to kill NAZGUL
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
WOLF2 hurts NAZGUL at GONDOR
NAZGUL travels to MORDOR because he wishes to destroy the object
ANDURILSWORD
NAZGUL destroys the object ANDURILSWORD at MORDOR
GANDALF kills (magic) NAZGUL at MORDOR with GANDALFSTAFF



Problema P4.PDDL

Problema P4.PDDL (0.1)

BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
FRODO (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
GANDALF (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF punches and hurts ORC2 at LA_COMARCA
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
FRODO gives the object RING to GANDALF at LA_COMARCA
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
BOROMIR hurts SARUMAN at ISENGARD with BOROMIRSWORD
ORC3 and BOROMIR are enemies because SARUMAN (ORC3's friend) and BOROMIR are enemies
GOLLUM travels to LA_COMARCA because he wishes the object RING GANDALF has
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
ARAGORN (ACQUIRE_SWORD) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
BOROMIR hurts ORC3 at ISENGARD with BOROMIRSWORD
FRODO and GOLLUM are friends because they have the motive PEACE in common
GANDALF gives the object RING to FRODO at LA_COMARCA
GOLLUM asks his friend FRODO to follow his motive ACQUIRE_RING
ORC3 travels to ROHAN because he wants to eat a RABBIT2
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are enemies
ORC3 (hurt) kills and eats RABBIT1 at ROHAN with ORC3SWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO loots the object ORC2SWORD from ORC2 (dead) at LA_COMARCA
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD



FRODO has acquired the object RING
WOLF1 hurts ORC3 at ROHAN
ARAGORN travels to LA_COMARCA because he wants to kill GOLLUM
ARAGORN and GOLLUM are friends because they want to kill SAURON
ORC3 travels to LA_COMARCA because he wants to kill FRODO
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO kills ORC3 at LA_COMARCA with ORC2SWORD
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Problema P4.PDDL (0.3)

GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
FRODO gives the object RING to GANDALF at LA_COMARCA
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
BILBO travels to LA_COMARCA because he wishes the object RING GANDALF has
GANDALF (DESTROY_RING) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN travels to MORDOR because he wants to kill SAURON
BILBO gives the object LITTLEELFSWORD to FRODO at LA_COMARCA
SAURON punches and hurts ARAGORN at MORDOR
GANDALF (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF travels to ISENGARD because he wants to kill ORC1
GANDALF punches and hurts SARUMAN at ISENGARD
GANDALF punches and hurts ORC3 at ISENGARD
SAM (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
BOROMIR travels to ISENGARD because he wishes the object RING GANDALF has
BILBO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
SARUMAN loots the object RING from GANDALF (dead) at ISENGARD
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree



GOLLUM (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
BOROMIR punches and hurts ORC2 at ISENGARD
BOROMIR (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
NAZGUL travels to LA_COMARCA because he wants to kill BILBO
BOROMIR punches and hurts ORC1 at ISENGARD
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SARUMAN has the motive ACQUIRE_RING because he wants to complete his motive CONQUER
ARAGORN travels to ROHAN because he wants to eat a RABBIT1
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
FRODO and NAZGUL are friends because they want to kill GOLLUM
BOROMIR kills ORC1 at ISENGARD with BOROMIRSWORD
ARAGORN (hurt) kills and eats RABBIT1 at ROHAN with ARAGORNSWORD
SARUMAN has acquired the object RING
BOROMIR kills ORC2 at ISENGARD with BOROMIRSWORD
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
FRODO and SAM are enemies because NAZGUL (FRODO's friend) and SAM are enemies
ARAGORN travels to MORDOR because he wants to kill SAURON
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ORC3 travels to LA_COMARCA because he wants to kill BILBO
FRODO kills ORC3 at LA_COMARCA with LITTLEELFSWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Problema P4.PDDL (0.5)

BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
FRODO gives the object RING to GANDALF at LA_COMARCA
GANDALF (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
NAZGUL travels to RIVENDEL because he wants to kill BILBO
NAZGUL punches and hurts BILBO at RIVENDEL



LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ORC2 and GANDALF are enemies because ORC3 (ORC2's friend) and GANDALF are enemies
BOROMIR (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
NAZGUL punches and hurts BOROMIR at RIVENDEL
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
NAZGUL kills BOROMIR at RIVENDEL with NAZGULSWORD
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ORC3 travels to LA_COMARCA because he wants to kill FRODO
ARAGORN takes the object ANDURILSWORD at GONDOR
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
GOLLUM (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
BILBO travels to ROHAN because he wants to eat a RABBIT1
LEGOLAS punches and hurts SARUMAN at ISENGARD
GANDALF punches and hurts ORC3 at LA_COMARCA
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
BILBO (hurt) kills and eats RABBIT1 at ROHAN with LITTLEELFSWORD
SAM and ORC3 are friends because they want to kill GOLLUM
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
FRODO loots the object ORC2SWORD from ORC2 (dead) at LA_COMARCA
FRODO kills ORC3 at LA_COMARCA with ORC2SWORD
SAM and ORC3 are enemies because FRODO (SAM's friend) has been killed by ORC3
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
GOLLUM travels to LA_COMARCA because he wishes the object RING
GANDALF has
GANDALF and GOLLUM are friends because they have the motive PEACE in common
GOLLUM asks his friend GANDALF to follow his motive ACQUIRE_RING
GANDALF has acquired the object RING
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Problema P4.PDDL (0.7)

GANDALF (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
FRODO gives the object RING to GANDALF at LA_COMARCA
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and SARUMAN (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree



FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC3 (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
ORC3 travels to LA_COMARCA because he wants to kill FRODO
GANDALF punches and hurts ORC3 at LA_COMARCA
GOLLUM (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
SAM and GANDALF are friends because they have the motive PEACE in common
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC2 (CONQUER) are enemies because their motives disagree
GANDALF gives the object RING to SAM at LA_COMARCA
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SAM and ORC3 are friends because they want to kill GOLLUM
SAM gives the object RING to ORC3 at LA_COMARCA
SAM and FRODO are enemies because ORC3 (SAM's friend) and FRODO are enemies
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ORC3 has the motive ACQUIRE_RING because he wants to complete his motive CONQUER
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
FRODO loots the object ORC2SWORD from ORC2 (dead) at LA_COMARCA
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ORC3 travels to ROHAN because he wants to eat a RABBIT2
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
ORC3 (hurt) kills and eats RABBIT1 at ROHAN with ORC3SWORD
ORC3 travels to LA_COMARCA because he wants to kill FRODO
ARAGORN takes the object ANDURILSWORD at GONDOR
LEGOLAS punches and hurts SARUMAN at ISENGARD
GANDALF punches and hurts ORC3 at LA_COMARCA
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
FRODO kills ORC3 at LA_COMARCA with ORC2SWORD
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
GOLLUM travels to LA_COMARCA because he wishes the object RING ORC3 has
GOLLUM loots the object RING from ORC3 (dead) at LA_COMARCA
GOLLUM has acquired the object RING
ARAGORN kills SAURON at MORDOR with ANDURILSWORD



Problema P4.PDDL (0.9)

BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
FRODO (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
WOLF2 hurts ARAGORN at GONDOR
ARAGORN travels to ISENGARD because he wants to kill SARUMAN
ARAGORN punches and hurts SARUMAN at ISENGARD
ARAGORN punches and hurts ORC3 at ISENGARD
FRODO travels to RIVENDEL because he wants to give the object RING to BILBO
ARAGORN (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ORC2 and ARAGORN are enemies because ORC3 (ORC2's friend) and ARAGORN are enemies
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
ARAGORN punches and hurts ORC1 at ISENGARD
ARAGORN punches and hurts ORC2 at ISENGARD
ARAGORN kills ORC1 at ISENGARD with ARAGORNSWORD
ARAGORN travels to ROHAN because he wants to eat a RABBIT1
ARAGORN (hurt) kills and eats RABBIT1 at ROHAN with ARAGORNSWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
GOLLUM travels to RIVENDEL because he wishes the object RING FRODO has
FRODO and GOLLUM are friends because they have the motive PEACE in common
GOLLUM asks his friend FRODO to follow his motive ACQUIRE_RING



SAM (PEACE) and FRODO (ACQUIRE_RING) are enemies because their motives disagree
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ORC2 travels to RIVENDEL because he wants to kill FRODO
FRODO kills ORC2 at RIVENDEL with LITTLEELFSWORD
ORC3 travels to RIVENDEL because he wants to kill FRODO
FRODO kills ORC3 at RIVENDEL with LITTLEELFSWORD
FRODO has acquired the object RING
ARAGORN kills SAURON at MORDOR with ANDURILSWORD



Problema P5.PDDL

Problema P5.PDDL (0.1)

ARAGORN (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC3 (CONQUER) are enemies because their motives disagree
ORC2 and GANDALF are enemies because ORC3 (ORC2's friend) and GANDALF are enemies
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
GIMLI (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
FRODO (DESTROY_RING) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
SAM and GANDALF are friends because they have the motive PEACE in common
NAZGUL travels to RIVENDEL because he wants to kill GIMLI
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
GIMLI punches and hurts NAZGUL at RIVENDEL
NAZGUL punches and hurts GIMLI at RIVENDEL
GIMLI travels to ROHAN because he wants to eat a RABBIT2
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
FRODO travels to RIVENDEL because he wants to give the object RING to BILBO
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
NAZGUL punches and hurts FRODO at RIVENDEL
BOROMIR (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
BILBO travels to MOUNT_DOOM because he wishes the object RING FRODO has
GANDALF travels to MORDOR because he wants to kill GOLLUM
NAZGUL punches and hurts BOROMIR at RIVENDEL
ARAGORN travels to RIVENDEL because he wants to kill NAZGUL
GIMLI has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE



LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
GIMLI (hurt) kills and eats RABBIT1 at ROHAN with GIMLIAXE
SAM has the motive DESTROY_RING because he wants to complete his
motive PEACE
GANDALF kills (magic) GOLLUM at MORDOR with GANDALFSTAFF
FRODO destroys the object RING at MOUNT_DOOM
WOLF1 hurts GIMLI at ROHAN
SAM (DESTROY_RING) and ORC3 (CONQUER) are enemies because their
motives disagree
ARAGORN travels to ISENGARD because he wants to kill SARUMAN
ARAGORN (PEACE) and ORC3 (CONQUER) are enemies because their motives
disagree
ORC3 travels to LA_COMARCA because he wants to kill SAM
GIMLI travels to MORDOR because he wants to kill SAURON
ARAGORN travels to RIVENDEL because he wants to kill NAZGUL
ARAGORN travels to LA_COMARCA because he wants to kill ORC3
SAM and ARAGORN are friends because they want to kill ORC3
NAZGUL kills BOROMIR at RIVENDEL with NAZGULSWORD
LEGOLAS punches and hurts SARUMAN at ISENGARD
ARAGORN hurts ORC3 at LA_COMARCA with ARAGORNSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
GIMLI wishes the object ANDURILSWORD because he has the motive
ACQUIRE_SWORD
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
SAM kills ORC3 at LA_COMARCA with ARAGORNSWORD
NAZGUL travels to LA_COMARCA because he wants to kill ARAGORN
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
GIMLI travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
GIMLI takes the object ANDURILSWORD at GONDOR
GIMLI kills WOLF2 at GONDOR with ANDURILSWORD
GIMLI travels to MORDOR because he wants to kill SAURON
GIMLI kills SAURON at MORDOR with ANDURILSWORD
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
ARAGORN kills NAZGUL at LA_COMARCA with ARAGORNSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA

Problema P5.PDDL (0.3)

ARAGORN (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their
motives disagree
GANDALF (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their
motives disagree
GANDALF (PEACE) and ORC3 (CONQUER) are enemies because their motives
disagree
GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives
disagree
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
SAM and GANDALF are friends because they have the motive PEACE in
common



ARAGORN (PEACE) and SAURON (ACQUIRE_RING) are enemies because their motives disagree
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
GANDALF punches and hurts ORC1 at LA_COMARCA
GANDALF (DESTROY_RING) and SARUMAN (CONQUER) are enemies because their motives disagree
ORC2 and GANDALF are enemies because ORC3 (ORC2's friend) and GANDALF are enemies
FRODO (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF travels to ISENGARD because he wants to kill ORC3
GANDALF travels to MORDOR because he wants to kill GOLLUM
ORC1 punches and hurts FRODO at LA_COMARCA
GANDALF (DESTROY_RING) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF kills (magic) GOLLUM at MORDOR with GANDALFSTAFF
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ORC1 travels to ROHAN because he wants to eat a RABBIT2
GIMLI travels to MORDOR because he wants to kill NAZGUL
GIMLI punches and hurts NAZGUL at MORDOR
NAZGUL punches and hurts GIMLI at MORDOR
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO travels to ROHAN because he wants to eat a RABBIT1
GIMLI kills NAZGUL at MORDOR with GIMLIAXE
FRODO travels to RIVENDEL because he wants ELROND heals him
ORC1 (hurt) kills and eats RABBIT1 at ROHAN with ORC1SWORD
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
ARAGORN punches and hurts NAZGUL at MORDOR
ARAGORN kills NAZGUL at MORDOR with ARAGORNSWORD
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
ORC3 travels to MORDOR because he wants to kill GANDALF
GANDALF kills (magic) ORC3 at MORDOR with GANDALFSTAFF
ARAGORN gives the object ARAGORNSWORD to GANDALF at MORDOR
WOLF1 hurts ORC1 at ROHAN
LEGOLAS hurts SARUMAN at ISENGARD with LEGOLASBOW
WOLF1 kills ORC1 at ROHAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
BOROMIR (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
FRODO destroys the object RING at MOUNT_DOOM



SAM (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
FRODO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
BOROMIR punches and hurts ORC2 at ISENGARD
ORC2 punches and hurts BOROMIR at ISENGARD
ORC2 kills BOROMIR at ISENGARD with ORC2SWORD
LEGOLAS (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
ORC2 travels to LA_COMARCA because he wants to kill SAM
LEGOLAS travels to LA_COMARCA because he wants to kill ORC2
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
GANDALF travels to LA_COMARCA because he wants to kill ORC2
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
GANDALF gives the object ARAGORNSWORD to SAM at LA_COMARCA
ARAGORN takes the object ANDURILSWORD at GONDOR
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
ARAGORN kills WOLF2 at GONDOR with ANDURILSWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Problema P5.PDDL (0.5)

GANDALF (DESTROY_RING) and NAZGUL (CONQUER) are enemies because their motives disagree
BILBO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
GANDALF (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
GANDALF (DESTROY_RING) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
SAM and GANDALF are friends because they have the motive PEACE in common
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC3 (CONQUER) are enemies because their motives disagree
GANDALF punches and hurts ORC2 at LA_COMARCA
ORC2 punches and hurts FRODO at LA_COMARCA
GANDALF (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
GANDALF travels to MORDOR because he wants to kill GOLLUM
FRODO travels to RIVENDEL because he wants to give the object RING to BILBO



ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF punches and hurts NAZGUL at MORDOR
NAZGUL travels to ROHAN because he wants to eat a RABBIT2
GANDALF kills (magic) GOLLUM at MORDOR with GANDALFSTAFF
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ORC1 travels to MORDOR because he wants to kill GANDALF
GANDALF kills (magic) ORC1 at MORDOR with GANDALFSTAFF
GANDALF has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN travels to ROHAN because he wants to kill NAZGUL
NAZGUL travels to RIVENDEL because he wishes the object RING FRODO has
NAZGUL punches and hurts GIMLI at RIVENDEL
WOLF1 hurts ARAGORN at ROHAN
ORC3 travels to MORDOR because he wants to kill GANDALF
GANDALF kills (magic) ORC3 at MORDOR with GANDALFSTAFF
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
LEGOLAS punches and hurts SARUMAN at ISENGARD
ARAGORN (hurt) kills and eats RABBIT1 at ROHAN with ARAGORNSWORD
BILBO wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
NAZGUL travels to ROHAN because he wants to eat a RABBIT2
ARAGORN kills NAZGUL at ROHAN with ARAGORNSWORD
GANDALF wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN travels to LA_COMARCA because he wants to kill ORC2
ARAGORN kills ORC2 at LA_COMARCA with ARAGORNSWORD
FRODO travels to RIVENDEL because he wants ELROND heals him
SAM and ARAGORN are friends because they have the motive PEACE in common
BILBO travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
WOLF2 hurts BILBO at GONDOR
BILBO kills WOLF2 at GONDOR with LITTLEELFSWORD
GANDALF travels to GONDOR because he wishes the object ANDURILSWORD
GANDALF takes the object ANDURILSWORD at GONDOR
GANDALF travels to MORDOR because he wants to kill SAURON
BILBO travels to RIVENDEL because he wants ELROND heals him
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
GANDALF kills SAURON at MORDOR with ANDURILSWORD



Problema P5.PDDL (0.7)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
FRODO (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
SAM and GANDALF are friends because they have the motive PEACE in common
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC2 (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ORC3 travels to LA_COMARCA because he wants to kill FRODO
GANDALF punches and hurts ORC3 at LA_COMARCA
ORC3 punches and hurts FRODO at LA_COMARCA
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
FRODO travels to RIVENDEL because he wants ELROND heals him
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
GANDALF kills (magic) ORC3 at LA_COMARCA with GANDALFSTAFF
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
ARAGORN kills WOLF2 at GONDOR with ARAGORNSWORD
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN travels to ISENGARD because he wants to kill SARUMAN
ARAGORN punches and hurts SARUMAN at ISENGARD
ARAGORN travels to MORDOR because he wants to kill SAURON



ARAGORN punches and hurts GOLLUM at MORDOR
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
ARAGORN punches and hurts NAZGUL at MORDOR
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
GIMLI travels to MORDOR because he wants to kill NAZGUL
NAZGUL punches and hurts GIMLI at MORDOR
ARAGORN kills NAZGUL at MORDOR with ARAGORNSWORD
GANDALF travels to MORDOR because he wants to kill GOLLUM
ARAGORN kills SAURON at MORDOR with ANDURILSWORD
ARAGORN gives the object ARAGORNSWORD to GANDALF at MORDOR
SAM has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SAM wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN kills GOLLUM at MORDOR with ANDURILSWORD
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
FRODO travels to ROHAN because he wants to eat a RABBIT1
SAM travels to MORDOR because he wishes the object ANDURILSWORD
ARAGORN has
GANDALF gives the object ARAGORNSWORD to SAM at MORDOR
ORC2 travels to MORDOR because he wants to kill GANDALF
GANDALF kills (magic) ORC2 at MORDOR with GANDALFSTAFF
FRODO (hurt) kills and eats RABBIT1 at ROHAN with LITTLEELFSWORD
WOLF1 hurts FRODO at ROHAN
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
FRODO destroys the object RING at MOUNT_DOOM

Problema P5.PDDL (0.9)

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
SAM and GANDALF are friends because they have the motive PEACE in common
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
FRODO (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree



LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
BILBO wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
WOLF2 hurts ARAGORN at GONDOR
ARAGORN kills WOLF2 at GONDOR with ARAGORNSWORD
GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GOLLUM travels to LA_COMARCA because he wishes the object RING FRODO has
GANDALF kills (magic) GOLLUM at LA_COMARCA with GANDALFSTAFF
FRODO travels to RIVENDEL because he wants to give the object RING to BILBO
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
NAZGUL travels to RIVENDEL because he wishes the object RING FRODO has
GIMLI punches and hurts NAZGUL at RIVENDEL
NAZGUL punches and hurts FRODO at RIVENDEL
NAZGUL punches and hurts GIMLI at RIVENDEL
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN travels to ISENGARD because he wants to kill SARUMAN
ARAGORN punches and hurts SARUMAN at ISENGARD
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
GANDALF travels to ISENGARD because he wants to kill ORC1
ARAGORN gives the object ARAGORNSWORD to GANDALF at ISENGARD
GANDALF kills (magic) ORC1 at ISENGARD with GANDALFSTAFF
GANDALF kills (magic) ORC2 at ISENGARD with GANDALFSTAFF
GANDALF kills (magic) ORC3 at ISENGARD with GANDALFSTAFF
SAM has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
SAM wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
SAM travels to ISENGARD because he wishes the object ANDURILSWORD
ARAGORN has
GANDALF gives the object ARAGORNSWORD to SAM at ISENGARD



ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN travels to RIVENDEL because he wants ELROND heals him
ARAGORN kills NAZGUL at RIVENDEL with ANDURILSWORD
FRODO travels to ROHAN because he wants to eat a RABBIT2
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN kills SAURON at MORDOR with ANDURILSWORD
FRODO (hurt) kills and eats RABBIT1 at ROHAN with LITTLEELFSWORD
WOLF1 hurts FRODO at ROHAN

Anexo IV: Encuesta realizada al público

Página 1: Introducción

Trabajo Fin de Grado: Generación Automática de Narrativas: Aventura Fantástica

Muchas gracias por participar en esta pequeña encuesta sobre mi proyecto de fin de carrera.

POR FAVOR LEED ATENTAMENTE ESTA PÁGINA

***Obligatorio**

¿Cual es tu nombre? (opcional, es solo para ver si te conozco)

¿Cual es tu sexo? *

Hombre

Mujer

¿Cuál es tu edad? *

< 10

10 - 20

21 - 30

31 - 40

41 - 50

> 50

Información sobre el proyecto

El proyecto consiste en la generación automática de narraciones. En mi lugar, lo he desarrollado sobre El Señor de los Anillos. No esperéis una novela digna de un escritor, ya que es generada aleatoriamente, y tan solo marca las etapas o hitos de la narración.

He creado un conjunto de acciones (ir a un lugar, matar, herir, coger un objeto, hacerse enemigo de alguien...) y marcandole un objetivo desarrolla una historia que llegue a ese fin.

En los siguientes 5 problemas (pondré 2 resultados de cada uno, por tanto, 10 narraciones) os pediré que evalúeis su "interés".

Espero que os guste y muchas gracias por participar, ¡los resultados saldrán en una gráfica en mi proyecto!

Ilustración 18 - Encuesta pág. 1 parte 1

Pequeñas aclaraciones

Antes de que me infléis a preguntas, aquí os resolveré unas cuantas que puede que os surjan:


- Los personajes se dirigen por "motivos" u objetivos (PEACE, CONQUER) básicamente para identificarlos como buenos o malos o lo que quieren hacer, como DESTROY_RING. Dichos motivos están unidos entre si, si tienes PEACE podrás DESTROY_RING con el que buscarás destruir el anillo.
 - Por norma general, para matar a alguien primero le hieres (hurt) y luego lo matas (kill), pero hay excepciones como la magia que mata de un golpe.
 - Gollum es especialito, para crear su bipolaridad se le han asignado tanto un motivo bueno (PEACE) como uno malo (ACQUIRE_RING). Ya veréis que la lia parda.
 - A Sauron solo se le puede matar con la espada ANDURIL, así que por un lado los buenos querrán conseguirla y los malos destruirla.
 - En cada problema os pondré cual es el objetivo que se pretende conseguir.
 - Los que no comprendáis bien el inglés, coged la narración y pasarla por google traductor, que lo traduce "decentemente".
- (para mas dudas preguntarme por el vídeo).

Estado inicial de los personajes

Para todos los problemas, este es el estado inicial:

- Bilbo está en Rivendel y tiene la espada LittleElfSword
- Frodo está en La Comarca y tiene el Anillo.
- Gandalf está en La Comarca, tiene su bastón, con el que puede matar mágicamente.
- Sam está en La Comarca, al igual que Merry y Pippin.
- Gollum está en Mordor (recién torturado) y como ya dije, desea tanto la Paz como el Anillo.
- Sauron está en Mordor, y no puede moverse de allí (es un NPC)
- Saruman está en Isengard y tampoco puede moverse de allí (tambien es un NPC)
- Aragorn está en Bree y posee su espada.
- Hay 3 orcos cuyos nombres he tardado horas en pensar: Orco1, Orco2, Orco3.
- Un Nazgul está en Mordor, con su espada.
- Elrond está en Rivendel, y tiene Medicina Elfica, con la que puede curar.
- Légolas, Gimli y Boromir también están en Rivendel y tienen sus respectivas armas.
- Hay 2 Conejitos en Rohan (comestibles, si comes uno te sanas las heridas)
- Hay 2 Lobos, uno en Rohan con los conejitos y otro en Gondor junto a la espada Anduril.

[Continuar »](#)

 16% completado

Con la tecnología de

Este contenido no ha sido creado ni aprobado por Google.

[Informar sobre abusos](#) - [Condiciones del servicio](#) - [Otros términos](#)

Ilustración 19 - Encuesta pág. 1 parte 2

Página 2: Problema 1

Problema 1

Este es el problema más sencillito.

El objetivo consiste en:

- El anillo debe ser destruido.
- Sauron debe estar muerto.

Como veréis son muy parecidos ya que tiene poca cosa

Narración 1.1

ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN (ACQUIRE_SWORD) and SAURON (ACQUIRE_RING) are enemies because their
motives disagree

ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD

GANDALF asks his friend FRODO to follow his motive DESTROY_RING

ARAGORN travels to MORDOR because he wants to kill SAURON

ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD

ARAGORN takes the object ANDURILSWORD at GONDOR

ARAGORN travels to MORDOR because he wants to kill SAURON

FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING

ARAGORN kills SAURON at MORDOR with ANDURILSWORD

FRODO destroys the object RING at MOUNT_DOOM

Narración 1.1 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

Narración 1.2

GANDALF asks his friend FRODO to follow his motive DESTROY_RING

ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree

ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE

ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD

ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD

ARAGORN takes the object ANDURILSWORD at GONDOR

FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING

ARAGORN travels to MORDOR because he wants to kill SAURON

FRODO destroys the object RING at MOUNT_DOOM

ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Narración 1.2 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

« Atrás Continuar »


 33% completado

Ilustración 20 - Encuesta pág. 2

Página 3: Problema 2

*Obligatorio

Problema 2

La cosa comienza a complicarse.

El objetivo consiste en:

- Boromir debe estar muerto
- Gandalf debe estar muerto
- La espada Anduril debe ser destruida
- Sauron debe adquirir el anillo.

Narración 2.1

GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
FRODO gives the object RING to GANDALF at LA_COMARCA
GANDALF (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF travels to MORDOR because he wants to kill SAURON
GANDALF travels to ISENGARD because he wants to kill SARUMAN
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
BOROMIR travels to ISENGARD because he wishes the object RING GANDALF has
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
NAZGUL destroys the object ANDURILSWORD at MORDOR
GOLLUM travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL loots the object RING from GANDALF (dead) at ISENGARD
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GOLLUM and NAZGUL are friends because they want to kill BILBO
NAZGUL and SARUMAN are enemies because GOLLUM (NAZGUL's friend) and SARUMAN
are enemies
SARUMAN kills (magic) NAZGUL at ISENGARD with SARUMANSTAFF
ELROND (PEACE) and SAURON (ACQUIRE_RING) are enemies because their motives disagree
SARUMAN punches and hurts GOLLUM at ISENGARD
GOLLUM loots the object RING from NAZGUL (dead) at ISENGARD
SARUMAN kills (magic) GOLLUM at ISENGARD with SARUMANSTAFF
ORC1 loots the object RING from GOLLUM (dead) at ISENGARD
ELROND (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ELROND travels to MORDOR because he wants to kill SAURON
ORC1 travels to MORDOR because he wants to kill ELROND
ORC1 and SAURON are friends because they want to kill ELROND
ORC1 gives the object RING to SAURON at MORDOR
SAURON has acquired the object RING

Narración 2.1 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

Ilustración 21 - Encuesta pág. 3 parte 1

Narración 2.2

GANDALF (DESTROY_RING) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SAURON and GOLLUM are friends because they have the motive ACQUIRE_RING in common
GANDALF wishes the object RING because he has the motive DESTROY_RING
FRODO (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
FRODO travels to ISENGARD because he wants to give the object RING to GANDALF
FRODO gives the object RING to GANDALF at ISENGARD
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
SARUMAN kills (magic) GANDALF at ISENGARD with SARUMANSTAFF
GOLLUM travels to ISENGARD because he wishes the object RING GANDALF has
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
GOLLUM loots the object RING from GANDALF (dead) at ISENGARD
GOLLUM (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GOLLUM travels to MORDOR because he wants to give the object RING to SAURON
GOLLUM (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
NAZGUL takes the object ANDURILSWORD at GONDOR
GOLLUM (PEACE) and NAZGUL (DESTROY_SWORD) are enemies because their motives disagree
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
SAURON punches and hurts GOLLUM at MORDOR
NAZGUL destroys the object ANDURILSWORD at MORDOR
NAZGUL kills GOLLUM at MORDOR with NAZGULSWORD
SAURON loots the object RING from GOLLUM (dead) at MORDOR
SAURON has acquired the object RING

Narración 2.2 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

« Atrás Continuar »

50% completado

Ilustración 22 - Encuesta pág. 3 parte 2

Página 4: Problema 3

*Obligatorio

Problema 3

Complicamos la cosa un poco más, los objetivos son los siguientes:

- Boromir muerto.
- Nazgul muerto.
- Saruman muerto.
- Frodo y Sam deben ser enemigos.
- Gollum y Aragorn deben ser amigos.
- La espada Anduril debe ser destruida (se puede destruir solo en Mordor)
- El anillo debe ser destruido (solo se puede destruir en Mount_Doom)

Narración 3.1

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
GOLLUM travels to LA_COMARCA because he wishes the object RING FRODO has
FRODO and GOLLUM are friends because they have the motive PEACE in common
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are enemies
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
FRODO destroys the object RING at MOUNT_DOOM
GANDALF travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF
GANDALF travels to MORDOR because he wants to kill NAZGUL
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
WOLF2 hurts NAZGUL at GONDOR
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
NAZGUL destroys the object ANDURILSWORD at MORDOR
GANDALF kills (magic) NAZGUL at MORDOR with GANDALFSTAFF

Narración 3.1 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

Ilustración 23 - Encuesta pág. 4 parte 1

Narración 3.2

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
NAZGUL has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
NAZGUL travels to LA_COMARCA because he wants to kill GANDALF
GOLLUM (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
SAM and NAZGUL are friends because they want to kill GOLLUM
NAZGUL asks his friend SAM to follow his motive CONQUER
FRODO (PEACE) and SAM (CONQUER) are enemies because their motives disagree
NAZGUL wishes the object ANDURILSWORD because he has the motive DESTROY_SWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
NAZGUL travels to GONDOR because he wishes the object ANDURILSWORD
NAZGUL takes the object ANDURILSWORD at GONDOR
WOLF2 hurts NAZGUL at GONDOR
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
BOROMIR punches and hurts SARUMAN at ISENGARD
FRODO destroys the object RING at MOUNT_DOOM
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
NAZGUL travels to MORDOR because he wishes to destroy the object ANDURILSWORD
ARAGORN (PEACE) and NAZGUL (DESTROY_SWORD) are enemies because their motives disagree
ARAGORN kills NAZGUL at MORDOR with ARAGORNSWORD
SAURON loots the object ANDURILSWORD from NAZGUL (dead) at MORDOR
SAURON has the motive DESTROY_SWORD because he wants to complete his motive CONQUER
SAURON destroys the object ANDURILSWORD at MORDOR
GANDALF travels to ISENGARD because he wants to kill SARUMAN
GANDALF kills (magic) SARUMAN at ISENGARD with GANDALFSTAFF

Narración 3.2 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

« Atrás

Continuar »

66% completado

Ilustración 24 - Encuesta pág. 4 parte 2

Página 5: Problema 4

*Obligatorio

Problema 4

Más complicado aún, los objetivos son:

- Boromir muerto (por si no queda claro, quiero que muera en todos)
- Orcos 1 y 2 muertos.
- Conejo1 muerto. (alguien debe comérselo)
- Frodo y Sam enemigos.
- Gollum y Aragorn amigos.
- Saruman debe ser asesinado por Légolas.
- Frodo debe matar al Orco3.
- El anillo debe ser "adquirido" (cualquiera, no solo Sauron)
- Sauron debe morir.

Ilustración 25 - Encuesta pág. 5 parte 1

Narración 4.1

BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
FRODO (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
GANDALF (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF punches and hurts ORC2 at LA_COMARCA
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
FRODO gives the object RING to GANDALF at LA_COMARCA
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
BOROMIR hurts SARUMAN at ISENGARD with BOROMIRSWORD
ORC3 and BOROMIR are enemies because SARUMAN (ORC3's friend) and BOROMIR are enemies
GOLLUM travels to LA_COMARCA because he wishes the object RING GANDALF has
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
ARAGORN (ACQUIRE_SWORD) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
BOROMIR hurts ORC3 at ISENGARD with BOROMIRSWORD
FRODO and GOLLUM are friends because they have the motive PEACE in common
GANDALF gives the object RING to FRODO at LA_COMARCA
GOLLUM asks his friend FRODO to follow his motive ACQUIRE_RING
ORC3 travels to ROHAN because he wants to eat a RABBIT2
FRODO and SAM are enemies because GOLLUM (FRODO's friend) and SAM are enemies
ORC3 (hurt) kills and eats RABBIT1 at ROHAN with ORC3SWORD
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO loots the object ORC2SWORD from ORC2 (dead) at LA_COMARCA
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
FRODO has acquired the object RING
WOLF1 hurts ORC3 at ROHAN
ARAGORN travels to LA_COMARCA because he wants to kill GOLLUM
ARAGORN and GOLLUM are friends because they want to kill SAURON
ORC3 travels to LA_COMARCA because he wants to kill FRODO
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO kills ORC3 at LA_COMARCA with ORC2SWORD
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Narración 4.1 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

Ilustración 26 - Encuesta pág. 5 parte 2

Narración 4.2

BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
SAM (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
FRODO gives the object RING to GANDALF at LA_COMARCA
GANDALF (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
NAZGUL travels to RIVENDEL because he wants to kill BILBO
NAZGUL punches and hurts BILBO at RIVENDEL
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ORC2 and GANDALF are enemies because ORC3 (ORC2's friend) and GANDALF are enemies
BOROMIR (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
NAZGUL punches and hurts BOROMIR at RIVENDEL
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
NAZGUL kills BOROMIR at RIVENDEL with NAZGULSWORD
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ORC3 travels to LA_COMARCA because he wants to kill FRODO
ARAGORN takes the object ANDURILSWORD at GONDOR
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
GOLLUM (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
BILBO travels to ROHAN because he wants to eat a RABBIT1
LEGOLAS punches and hurts SARUMAN at ISENGARD
GANDALF punches and hurts ORC3 at LA_COMARCA
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
BILBO (hurt) kills and eats RABBIT1 at ROHAN with LITTLEELFSWORD
SAM and ORC3 are friends because they want to kill GOLLUM
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
FRODO loots the object ORC2SWORD from ORC2 (dead) at LA_COMARCA
FRODO kills ORC3 at LA_COMARCA with ORC2SWORD
SAM and ORC3 are enemies because FRODO (SAM's friend) has been killed by ORC3
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN and GOLLUM are friends because they have the motive PEACE in common
GOLLUM travels to LA_COMARCA because he wishes the object RING
GANDALF and GOLLUM are friends because they have the motive PEACE in common
GOLLUM asks his friend GANDALF to follow his motive ACQUIRE_RING
GANDALF has acquired the object RING
ARAGORN kills SAURON at MORDOR with ANDURILSWORD

Narración 4.2 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

« Atrás

Continuar »

83% completado

Ilustración 27 - Encuesta pág. 5 parte 3

Página 6: Problema 5

*Obligatorio

Problema 5

El ultimo ya. Se que debéis estar hartos de tanta gilipollez sin sentido, pero queda poquito para que me hagáis un enorme favor!

Como es el último... también es el más difícil. Objetivos:

- Boromir muerto, por supuesto.
- Frodo debe tener la Espadita elfica (LittleElfSword)
- Orcos 1, 2 y 3 muertos.
- Aragorn debe matar al Nazgul.
- Legolas debe matar a Saruman.
- Frodo y Gimli deben estar heridos (hurt)
- Sam debe tener la espada de Aragorn (no preguntéis porqué, se me ha ocurrido a lo random)
- Gollum, el Conejo1 y el Lobo1 deben estar muertos.

y por último...

- Anillo destruido
- Sauron muerto.

Ilustración 28 - Encuesta pág. 6 parte 1

Narración 5.1

GANDALF asks his friend FRODO to follow his motive DESTROY_RING
SAM and GANDALF are friends because they have the motive PEACE in common
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
FRODO (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
FRODO (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
ARAGORN (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC2 (CONQUER) are enemies because their motives disagree
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and ORC1 (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
BOROMIR (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
ARAGORN has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
BILBO wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
ARAGORN travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN takes the object ANDURILSWORD at GONDOR
WOLF2 hurts ARAGORN at GONDOR
ARAGORN kills WOLF2 at GONDOR with ARAGORNSWORD
GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GOLLUM travels to LA_COMARCA because he wishes the object RING FRODO has
GANDALF kills (magic) GOLLUM at LA_COMARCA with GANDALFSTAFF
FRODO travels to RIVENDEL because he wants to give the object RING to BILBO
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
NAZGUL travels to RIVENDEL because he wishes the object RING FRODO has
GIMLI punches and hurts NAZGUL at RIVENDEL
NAZGUL punches and hurts FRODO at RIVENDEL
NAZGUL punches and hurts GIMLI at RIVENDEL
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
ARAGORN travels to ISENGARD because he wants to kill SARUMAN
ARAGORN punches and hurts SARUMAN at ISENGARD
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
BOROMIR travels to ISENGARD because he wants to kill SARUMAN
SARUMAN kills (magic) BOROMIR at ISENGARD with SARUMANSTAFF
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
GANDALF travels to ISENGARD because he wants to kill ORC1
ARAGORN gives the object ARAGORNSWORD to GANDALF at ISENGARD
GANDALF kills (magic) ORC1 at ISENGARD with GANDALFSTAFF
GANDALF kills (magic) ORC2 at ISENGARD with GANDALFSTAFF
GANDALF kills (magic) ORC3 at ISENGARD with GANDALFSTAFF
SAM has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
SAM wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
SAM travels to ISENGARD because he wishes the object ANDURILSWORD ARAGORN has
GANDALF gives the object ARAGORNSWORD to SAM at ISENGARD
ARAGORN travels to MORDOR because he wants to kill SAURON
FRODO destroys the object RING at MOUNT_DOOM
ARAGORN travels to RIVENDEL because he wants ELROND heals him
ARAGORN kills NAZGUL at RIVENDEL with ANDURILSWORD
FRODO travels to ROHAN because he wants to eat a RABBIT2
ARAGORN travels to MORDOR because he wants to kill SAURON
ARAGORN kills SAURON at MORDOR with ANDURILSWORD
FRODO (hurt) kills and eats RABBIT1 at ROHAN with LITTLEELFSWORD
WOLF1 hurts FRODO at ROHAN

Narración 5.1 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

Ilustración 29 - Encuesta pág. 6 parte 2

Narración 5.2

ARAGORN (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC3 (CONQUER) are enemies because their motives disagree
ORC2 and GANDALF are enemies because ORC3 (ORC2's friend) and GANDALF are enemies
GANDALF asks his friend FRODO to follow his motive DESTROY_RING
GIMLI (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (DESTROY_RING) and ORC1 (CONQUER) are enemies because their motives disagree
GIMLI (PEACE) and SAURON (CONQUER) are enemies because their motives disagree
FRODO (DESTROY_RING) and NAZGUL (CONQUER) are enemies because their motives disagree
GANDALF (PEACE) and GOLLUM (ACQUIRE_RING) are enemies because their motives disagree
SAM and GANDALF are friends because they have the motive PEACE in common
NAZGUL travels to RIVENDEL because he wants to kill GIMLI
ORC1 travels to LA_COMARCA because he wants to kill GANDALF
GIMLI punches and hurts NAZGUL at RIVENDEL
NAZGUL punches and hurts GIMLI at RIVENDEL
GIMLI travels to ROHAN because he wants to eat a RABBIT2
LEGOLAS (PEACE) and SARUMAN (CONQUER) are enemies because their motives disagree
BILBO has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
GANDALF kills (magic) ORC1 at LA_COMARCA with GANDALFSTAFF
FRODO travels to RIVENDEL because he wants to give the object RING to BILBO
BILBO gives the object LITTLEELFSWORD to FRODO at RIVENDEL
ARAGORN (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
NAZGUL punches and hurts FRODO at RIVENDEL
BOROMIR (PEACE) and NAZGUL (CONQUER) are enemies because their motives disagree
ORC2 travels to LA_COMARCA because he wants to kill GANDALF
FRODO travels to MOUNT_DOOM because he wishes to destroy the object RING
GANDALF kills (magic) ORC2 at LA_COMARCA with GANDALFSTAFF
BILBO travels to MOUNT_DOOM because he wishes the object RING FRODO has
GANDALF travels to MORDOR because he wants to kill GOLLUM
NAZGUL punches and hurts BOROMIR at RIVENDEL
ARAGORN travels to RIVENDEL because he wants to kill NAZGUL
GIMLI has the motive ACQUIRE_SWORD because he wants to complete his motive PEACE
LEGOLAS travels to ISENGARD because he wants to kill SARUMAN
GIMLI (hurt) kills and eats RABBIT1 at ROHAN with GIMLIAXE
SAM has the motive DESTROY_RING because he wants to complete his motive PEACE
GANDALF kills (magic) GOLLUM at MORDOR with GANDALFSTAFF
FRODO destroys the object RING at MOUNT_DOOM
WOLF1 hurts GIMLI at ROHAN
SAM (DESTROY_RING) and ORC3 (CONQUER) are enemies because their motives disagree
ARAGORN travels to ISENGARD because he wants to kill SARUMAN
ARAGORN (PEACE) and ORC3 (CONQUER) are enemies because their motives disagree
ORC3 travels to LA_COMARCA because he wants to kill SAM
GIMLI travels to MORDOR because he wants to kill SAURON
ARAGORN travels to RIVENDEL because he wants to kill NAZGUL
ARAGORN travels to LA_COMARCA because he wants to kill ORC3
SAM and ARAGORN are friends because they want to kill ORC3
NAZGUL kills BOROMIR at RIVENDEL with NAZGULSWORD
LEGOLAS punches and hurts SARUMAN at ISENGARD
ARAGORN hurts ORC3 at LA_COMARCA with ARAGORNSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
GIMLI wishes the object ANDURILSWORD because he has the motive ACQUIRE_SWORD
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
LEGOLAS kills SARUMAN at ISENGARD with LEGOLASBOW
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
SAM kills ORC3 at LA_COMARCA with ARAGORNSWORD
NAZGUL travels to LA_COMARCA because he wants to kill ARAGORN
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
GIMLI travels to GONDOR because he wishes the object ANDURILSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA
GIMLI takes the object ANDURILSWORD at GONDOR
GIMLI kills WOLF2 at GONDOR with ANDURILSWORD
GIMLI travels to MORDOR because he wants to kill SAURON
GIMLI kills SAURON at MORDOR with ANDURILSWORD
SAM gives the object ARAGORNSWORD to ARAGORN at LA_COMARCA
ARAGORN kills NAZGUL at LA_COMARCA with ARAGORNSWORD
ARAGORN gives the object ARAGORNSWORD to SAM at LA_COMARCA

Narración 5.2 ¿Qué te ha parecido? *

1 2 3 4 5

Es poco interesante Es muy interesante

Resultados de la encuesta

A continuación se expondrán los resultados obtenidos en esta encuesta. En la siguiente tabla se muestran las valoraciones del público para cada narración, así como su sexo y edad. Las narraciones han sido enumeradas para la encuesta siguiendo un patrón aleatorio, sin tener nada que ver con la verdadera aleatoriedad de las narraciones generadas. Dichas enumeraciones se corresponden con los siguientes problemas y aleatoriedad:

Enumeración encuesta	Problema (Aleatoriedad)
Narración 1.1	P1 (0.3)
Narración 1.2	P1 (0.9)
Narración 2.1	P2 (0.7)
Narración 2.2	P2 (0.1)
Narración 3.1	P3 (0.9)
Narración 3.2	P3 (0.5)
Narración 4.1	P4 (0.1)
Narración 4.2	P4 (0.5)
Narración 5.1	P5 (0.9)
Narración 5.2	P5 (0.1)

Tabla 31 - Enumeración de narraciones

De esta manera, el encuestado no supo en ningún momento la relación de la enumeración con la aleatoriedad de la narración, consiguiendo así unos resultados más fiables.

El sexo de los participantes se ha almacenado de forma booleana, siendo hombres el 1 y mujeres el 0. La edad viene marcada por el intervalo marcado por la encuesta. Estos han sido los resultados:



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	> 50	4	3	4	1	5	2	4	3	4	1
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	2	4	5	3	5	2	4	4	3	4
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	3	2	3	3	3	2	2	2	4
1	10 - 20	4	3	4	2	5	4	5	3	5	4
1	10 - 20	3	4	2	3	3	4	5	5	5	5
1	10 - 20	5	5	5	5	5	5	5	4	5	5
1	10 - 20	4	5	4	5	5	3	4	5	2	4
1	10 - 20	4	3	4	3	3	4	4	4	3	4
1	21 - 30	4	3	5	5	4	5	4	4	5	5
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	4	3	5	4	2	2	1	5	3
1	10 - 20	4	5	5	4	3	5	5	5	5	5
1	21 - 30	5	3	2	3	4	2	4	5	4	4
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	10 - 20	3	4	5	5	5	4	4	5	4	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	4	4	4	3	4	3	3	4	4	4
1	10 - 20	4	4	3	5	5	3	4	4	5	5
1	10 - 20	2	4	5	3	3	4	3	3	5	3
1	21 - 30	3	3	3	3	3	3	3	3	1	3
1	21 - 30	5	4	4	2	3	2	5	5	5	4
1	10 - 20	3	2	4	5	3	4	4	4	1	3
1	21 - 30	3	3	4	5	4	3	4	4	4	4
1	10 - 20	4	5	5	4	3	3	4	3	5	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	3	4	3	2	3	4	4	3	4	3
1	10 - 20	3	3	4	2	5	2	4	4	3	4
1	10 - 20	4	1	4	3	4	3	4	3	4	3
1	21 - 30	1	1	1	1	1	1	1	1	1	1
1	10 - 20	3	4	3	3	1	4	4	4	3	4
0	21 - 30	4	5	3	4	4	3	4	5	3	5
1	10 - 20	5	5	3	3	5	4	5	5	5	5
1	21 - 30	4	2	2	3	4	3	4	4	4	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	3	4	4	5	4	4	4	5	4
1	21 - 30	5	5	4	3	4	5	4	4	3	4
1	10 - 20	4	4	2	5	5	5	1	1	3	3
1	10 - 20	5	4	5	5	5	5	4	4	5	3
1	10 - 20	5	5	4	3	5	4	4	5	5	3
1	21 - 30	4	3	4	2	2	5	3	4	4	3
0	> 50	1	1	1	1	1	1	1	1	1	1
1	10 - 20	3	4	4	3	4	4	4	3	2	5
1	10 - 20	3	2	4	5	2	1	4	4	3	4
1	10 - 20	4	5	4	5	4	5	4	5	3	4
1	21 - 30	4	3	3	2	3	4	4	2	3	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	3	5	3	2	5	5	4	4	4	3



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	2	3	2	2	2	1	3	2	4
1	10 - 20	4	4	5	5	5	5	4	4	4	5
1	31 - 40	5	5	4	3	3	2	2	1	3	1
1	10 - 20	2	5	5	3	5	3	5	3	4	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	3	4	2	3	2	4	2	3	2	3
1	10 - 20	4	3	3	4	4	5	1	4	4	4
1	21 - 30	5	5	5	5	5	5	5	5	5	5
1	10 - 20	2	4	4	4	4	4	4	5	4	4
1	10 - 20	4	3	4	5	2	4	4	4	5	4
1	21 - 30	4	4	3	4	4	4	4	4	3	3
1	10 - 20	4	4	3	5	4	3	2	3	4	5
1	21 - 30	4	3	3	4	2	4	3	4	2	4
1	10 - 20	4	3	5	4	4	4	5	3	4	5
1	21 - 30	5	5	3	3	2	2	1	1	3	2
1	10 - 20	3	3	5	5	3	4	4	5	5	5
1	10 - 20	4	4	4	4	3	3	3	3	3	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	10 - 20	5	4	3	2	3	3	1	1	2	1
1	10 - 20	3	4	4	4	3	4	5	4	4	5
1	21 - 30	3	3	4	4	4	4	4	4	4	4
1	10 - 20	3	1	2	5	1	3	4	3	1	4
1	21 - 30	2	5	5	2	1	4	5	3	3	5
1	10 - 20	3	4	4	5	5	5	4	5	3	4
1	10 - 20	2	4	2	3	1	2	3	3	4	4
1	21 - 30	5	3	5	5	4	1	4	3	4	4
1	10 - 20	4	3	5	2	1	4	1	4	2	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	5	4	3	5	4	5	3	2	2	4
1	10 - 20	4	5	5	5	5	5	3	4	5	4
1	10 - 20	4	4	5	3	5	4	5	5	4	4
1	10 - 20	4	4	4	5	4	3	1	2	1	5
0	10 - 20	4	5	3	5	5	1	3	4	4	2
1	21 - 30	3	4	3	2	4	4	3	3	4	3
1	10 - 20	2	4	1	3	3	4	2	4	3	3
1	10 - 20	3	3	4	1	4	5	4	3	4	4
1	10 - 20	4	3	4	5	2	5	2	5	3	4
1	10 - 20	2	3	3	3	3	3	3	3	3	3
1	21 - 30	3	3	2	2	3	5	3	4	5	5
1	10 - 20	4	2	4	3	2	4	5	4	4	5
0	10 - 20	4	3	3	4	3	4	4	4	4	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	5	3	3	4	3	4	5	4	3
1	21 - 30	4	5	3	4	4	4	4	4	3	4
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	10 - 20	5	4	4	5	4	5	5	5	5	3
1	10 - 20	4	4	4	5	5	4	5	4	4	5
1	21 - 30	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	5	5	5	5	5	5	5	5	5
1	10 - 20	4	4	4	4	3	3	3	3	1	2
1	10 - 20	4	4	3	4	3	5	4	2	4	5



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	3	4	2	3	3	3	5	4	5	2
1	10 - 20	3	4	5	4	3	2	5	5	5	5
1	10 - 20	5	5	5	1	1	5	1	5	5	1
1	10 - 20	5	4	5	4	4	5	5	5	4	4
1	10 - 20	4	5	4	5	2	3	1	5	3	5
1	21 - 30	2	3	1	5	3	4	1	4	2	1
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	2	3	3	4	4	2	3	4	3	1
1	21 - 30	5	4	5	5	5	5	5	5	2	4
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	21 - 30	4	4	4	4	4	4	4	4	4	4
1	10 - 20	5	5	4	4	5	5	3	3	5	5
1	10 - 20	3	3	4	4	3	3	3	3	3	3
1	10 - 20	3	2	1	4	1	3	1	2	1	3
1	21 - 30	5	5	4	4	5	4	4	5	5	5
1	10 - 20	5	4	4	2	4	5	5	2	3	5
1	10 - 20	3	3	3	3	3	3	4	3	4	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	4	4	3	5	2	3	2	4	3	3
1	10 - 20	4	5	5	4	5	5	4	5	5	5
1	21 - 30	4	5	5	4	5	4	4	3	5	3
1	21 - 30	3	5	2	2	5	4	4	3	3	3
1	21 - 30	3	5	4	3	4	4	4	4	4	4
1	10 - 20	5	5	5	1	1	1	1	1	1	1
1	10 - 20	4	3	4	2	4	2	4	5	3	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	3	3	4	4	4	3	5	4	4	3
0	21 - 30	4	3	4	5	4	3	5	5	5	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	5	5	4	5	4	5	5	5	5	5
1	10 - 20	4	4	5	4	4	4	4	5	4	4
1	10 - 20	2	5	5	5	5	5	5	5	5	5
1	10 - 20	5	4	3	2	5	5	5	5	5	2
1	10 - 20	4	4	3	4	4	3	4	4	3	4
1	10 - 20	3	3	4	2	3	4	4	4	3	3
1	10 - 20	3	4	4	5	5	4	3	4	4	4
1	10 - 20	4	4	4	5	4	5	4	5	5	4
1	21 - 30	5	5	5	3	4	3	4	5	4	4
1	10 - 20	3	3	5	5	4	4	4	4	4	4
1	21 - 30	4	2	4	2	2	4	4	4	4	4
1	10 - 20	2	3	3	2	4	2	2	2	2	3
1	10 - 20	5	4	4	3	4	3	4	5	4	4
1	10 - 20	3	5	4	5	5	3	2	3	4	3
1	21 - 30	4	3	4	3	3	3	4	2	2	3
1	10 - 20	3	5	4	5	5	3	2	3	4	3
1	10 - 20	5	2	5	5	5	5	4	5	5	4
1	10 - 20	3	3	3	3	3	3	3	3	3	4
1	> 50	1	1	1	1	1	1	1	1	1	1
1	10 - 20	3	3	4	2	3	4	2	2	3	4
1	10 - 20	4	5	3	4	5	5	4	5	3	5
1	21 - 30	5	3	4	5	3	4	5	4	4	5
1	10 - 20	3	3	3	3	3	3	3	3	3	3



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	21 - 30	4	3	4	2	3	5	3	3	4	2
1	10 - 20	3	4	3	4	3	2	4	5	5	4
0	21 - 30	4	3	5	4	3	2	5	5	2	5
1	10 - 20	5	2	4	1	4	5	3	3	5	1
1	10 - 20	5	4	3	4	2	3	3	4	5	3
1	10 - 20	4	5	3	4	4	3	3	4	4	4
1	10 - 20	4	3	5	5	5	4	3	5	4	5
1	10 - 20	4	3	3	4	4	3	4	4	3	4
1	10 - 20	5	5	5	5	4	5	4	3	4	5
1	10 - 20	5	3	4	3	4	2	4	5	2	3
1	21 - 30	4	3	3	2	4	2	3	2	3	1
1	10 - 20	4	5	5	4	4	3	4	3	4	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	21 - 30	4	4	4	4	4	4	4	4	5	4
1	10 - 20	4	5	3	5	5	4	4	5	4	5
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	5	5	4	4	4	4	4	4	4	4
1	10 - 20	4	3	1	3	2	3	3	2	2	4
1	10 - 20	5	3	5	4	1	3	5	2	3	5
1	21 - 30	3	3	3	3	3	3	3	3	5	4
1	10 - 20	4	4	4	3	3	4	3	3	3	4
1	> 50	4	3	3	4	4	5	5	4	2	3
1	21 - 30	4	5	5	5	4	5	5	5	4	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	5	3	4	4	3	2	1	4	3	1
1	10 - 20	5	5	5	5	3	4	3	3	3	5
1	10 - 20	4	4	5	4	3	4	4	5	4	4
1	31 - 40	2	3	4	4	3	5	2	3	4	5
1	10 - 20	5	4	4	5	3	4	3	3	4	4
1	21 - 30	3	2	3	3	2	2	3	2	2	4
1	21 - 30	4	4	5	4	5	3	5	4	4	5
1	10 - 20	1	1	2	4	5	3	5	4	5	4
0	10 - 20	3	2	4	5	3	4	2	4	5	5
1	21 - 30	5	4	4	4	4	4	4	4	4	4
1	21 - 30	2	4	4	2	4	2	3	3	1	3
1	10 - 20	3	4	4	5	3	5	5	3	5	2
1	10 - 20	4	3	4	5	5	4	2	3	4	2
1	10 - 20	4	1	1	1	3	3	2	3	5	3
1	10 - 20	5	3	1	2	4	3	4	5	1	5
1	10 - 20	3	3	2	4	3	4	3	2	4	4
1	10 - 20	3	3	4	4	3	3	4	3	3	3
1	10 - 20	2	2	2	4	4	2	3	3	2	3
1	10 - 20	4	4	3	3	2	4	4	3	5	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	31 - 40	1	3	5	3	5	5	5	4	1	3
1	10 - 20	3	5	3	5	4	5	4	5	3	2
1	10 - 20	4	4	5	5	4	2	3	4	5	4
1	10 - 20	3	4	3	5	4	3	3	4	5	4
1	10 - 20	3	4	3	4	3	3	3	3	3	3
1	21 - 30	5	4	2	3	4	2	3	3	2	3
1	21 - 30	4	4	4	4	4	4	4	5	5	5



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	4	5	5	4	5	5	5	3	4	5
1	10 - 20	3	4	5	4	3	3	5	4	5	4
1	10 - 20	5	4	4	4	5	5	5	4	4	5
1	21 - 30	2	1	5	5	2	4	5	3	4	3
1	21 - 30	5	5	5	4	5	5	5	5	5	5
1	21 - 30	3	2	3	3	3	4	3	3	3	3
1	10 - 20	4	4	4	3	3	5	5	5	4	4
1	10 - 20	5	4	4	4	3	4	3	4	4	3
1	10 - 20	5	4	3	4	5	4	3	5	5	3
1	21 - 30	5	5	3	3	4	2	4	3	4	3
1	21 - 30	5	3	5	3	3	1	2	4	3	5
1	21 - 30	3	4	3	3	3	3	3	3	3	3
1	31 - 40	3	3	3	3	3	3	3	3	3	2
1	21 - 30	4	4	4	4	2	3	5	5	5	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	2	2	2	2	3	5	3	1	2	4
1	10 - 20	5	4	5	5	5	5	5	4	5	5
1	10 - 20	4	5	4	5	5	3	2	3	4	3
1	10 - 20	4	4	5	3	4	4	5	3	5	4
1	10 - 20	4	3	3	4	3	3	4	5	3	3
1	10 - 20	3	4	3	4	5	3	4	5	4	2
1	10 - 20	3	3	4	2	2	3	1	4	3	5
1	10 - 20	2	3	2	3	3	4	5	3	4	2
1	31 - 40	2	1	1	1	2	2	1	1	1	1
1	10 - 20	3	4	4	2	3	5	4	4	5	5
1	21 - 30	5	5	2	3	2	4	4	3	4	3
1	21 - 30	5	5	4	4	4	4	4	4	4	4
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	10 - 20	4	4	5	4	4	3	5	4	2	5
1	31 - 40	4	3	4	4	2	3	5	2	5	3
1	21 - 30	4	5	4	4	5	3	4	5	5	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	4	4	5	5	4	5	4	5	5	5
0	31 - 40	5	3	3	3	3	3	3	3	3	3
1	10 - 20	3	2	5	3	2	3	4	3	3	4
1	21 - 30	4	3	4	2	4	3	2	4	2	4
1	10 - 20	1	3	2	2	3	4	2	2	3	2
1	10 - 20	4	5	3	4	4	5	5	3	4	5
1	21 - 30	3	5	1	5	5	4	4	4	4	4
1	21 - 30	3	2	3	2	4	4	3	3	4	3
1	21 - 30	4	5	4	5	5	4	4	5	5	4
1	10 - 20	5	1	2	2	5	2	4	5	5	1
1	10 - 20	4	3	4	3	4	3	4	2	5	4
1	10 - 20	4	3	5	5	5	4	4	5	3	5
1	21 - 30	5	4	5	5	3	4	5	3	5	5
1	10 - 20	4	2	4	4	2	4	3	2	2	3
1	21 - 30	3	4	2	3	3	4	4	3	4	3
1	10 - 20	4	3	2	4	5	3	4	4	3	3
1	21 - 30	4	4	5	4	4	4	4	4	4	4
1	10 - 20	4	3	5	3	2	5	4	5	5	5
1	31 - 40	5	5	2	4	5	4	2	3	2	2
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	3	4	4	4	3	4	5	5	4	5



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	5	5	4	5	5	4	5	4	5	5
1	21 - 30	3	3	3	3	3	3	3	3	3	3
1	21 - 30	5	4	4	4	4	4	4	3	4	4
1	10 - 20	4	4	3	4	4	4	3	4	5	4
1	10 - 20	4	5	3	5	4	4	4	4	3	5
0	21 - 30	1	5	5	1	5	1	1	5	1	5
1	10 - 20	4	3	3	4	5	4	4	5	4	4
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	10 - 20	3	4	2	2	4	4	4	4	4	4
1	21 - 30	1	1	1	1	1	1	1	1	1	1
1	10 - 20	3	4	4	5	4	3	4	5	5	4
1	21 - 30	4	4	3	4	4	3	4	4	4	4
1	10 - 20	4	3	3	4	3	3	3	3	3	3
0	10 - 20	3	3	4	4	1	5	5	4	4	5
1	21 - 30	3	5	2	4	5	1	5	2	3	2
1	10 - 20	3	4	3	5	5	5	5	5	5	5
1	21 - 30	5	4	5	5	4	4	3	4	5	5
1	10 - 20	3	5	3	5	2	4	4	4	5	5
1	10 - 20	4	5	5	4	3	5	4	4	4	4
1	21 - 30	5	4	5	5	4	5	5	5	4	4
1	10 - 20	5	4	5	5	4	5	5	4	5	5
1	10 - 20	4	3	2	4	3	4	2	2	2	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	4	5	4	3	5	4	4	3	5	5
1	10 - 20	4	3	2	4	5	5	5	4	4	5
1	10 - 20	3	4	2	2	2	2	2	2	2	2
1	10 - 20	3	3	1	2	2	2	2	2	2	2
1	10 - 20	4	4	5	5	5	5	5	5	5	5
1	10 - 20	5	4	2	3	4	4	4	5	4	5
1	10 - 20	3	4	2	2	5	3	4	4	4	5
1	10 - 20	4	5	5	5	5	5	5	5	5	5
1	10 - 20	5	4	5	4	5	5	4	4	5	5
1	21 - 30	4	5	3	3	4	3	4	3	5	4
1	10 - 20	4	5	4	4	3	4	3	3	5	5
1	10 - 20	2	2	1	1	1	1	2	1	1	2
1	10 - 20	4	4	5	4	3	1	5	5	4	5
1	10 - 20	3	5	3	1	4	1	5	2	5	3
1	10 - 20	4	5	5	4	4	4	3	4	5	5
1	10 - 20	5	4	5	5	5	5	3	4	5	5
1	10 - 20	4	3	3	3	4	3	3	4	3	5
1	21 - 30	3	3	4	4	4	5	4	4	5	4
1	21 - 30	4	3	4	4	2	2	3	3	5	5
1	10 - 20	4	4	4	3	3	3	4	3	4	3
1	10 - 20	3	3	4	4	4	4	4	5	5	5
1	21 - 30	4	3	5	5	4	2	4	3	4	4
1	10 - 20	4	4	3	4	4	3	4	3	5	4
1	10 - 20	2	3	3	4	3	4	5	4	4	5
1	10 - 20	4	3	5	3	1	5	3	5	2	5
1	21 - 30	3	3	4	3	1	3	4	4	3	3
1	10 - 20	4	5	5	4	4	5	3	4	4	5
1	10 - 20	4	3	4	4	4	4	4	4	3	4
1	21 - 30	2	3	4	3	5	5	3	4	3	3
1	10 - 20	5	5	4	3	4	5	1	2	5	5



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	21 - 30	4	4	5	3	5	5	4	2	5	5
1	10 - 20	4	5	4	3	3	4	4	4	3	4
1	10 - 20	4	3	2	4	3	2	4	4	2	2
1	10 - 20	5	4	3	4	3	3	3	2	4	3
1	21 - 30	3	3	1	2	1	2	2	1	1	1
1	10 - 20	1	3	5	2	4	3	1	4	4	3
1	21 - 30	3	4	4	5	3	5	4	4	5	5
1	10 - 20	3	5	2	2	3	4	2	3	4	3
1	21 - 30	5	5	4	5	5	5	5	5	5	4
1	10 - 20	3	3	2	3	2	4	3	3	3	2
1	21 - 30	4	4	3	4	2	2	3	3	3	3
1	10 - 20	5	4	4	5	5	4	4	5	5	4
1	21 - 30	4	4	3	3	4	3	3	4	4	3
1	10 - 20	3	3	2	1	4	2	2	5	1	3
1	10 - 20	4	4	2	3	4	3	3	4	5	3
1	10 - 20	1	3	3	3	4	2	4	3	4	5
1	21 - 30	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	5	3	3	3	4	3	2	4	3
1	10 - 20	3	4	3	3	4	3	3	4	4	4
1	21 - 30	2	3	2	3	2	2	3	2	3	3
1	10 - 20	5	5	4	4	4	5	5	5	5	5
1	10 - 20	4	4	4	4	4	4	4	3	4	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	3	4	4	4	5	4	5	4	4	5
1	10 - 20	5	4	5	5	4	5	5	4	5	4
1	10 - 20	4	3	5	3	1	3	4	3	4	2
1	41 - 50	3	2	1	3	2	4	4	4	4	3
1	21 - 30	5	5	5	1	5	5	5	3	5	2
1	21 - 30	4	3	4	3	5	4	3	2	4	3
1	31 - 40	3	3	3	3	4	4	3	4	5	3
1	10 - 20	3	3	2	4	3	5	3	4	3	2
1	21 - 30	5	5	4	5	5	4	4	4	5	5
1	10 - 20	5	5	4	5	4	5	5	4	4	5
1	10 - 20	5	5	4	4	4	4	4	4	4	4
1	10 - 20	3	2	4	4	2	3	4	3	3	5
1	10 - 20	3	2	2	4	2	2	4	2	3	2
1	10 - 20	3	5	5	4	3	4	5	5	3	5
1	10 - 20	5	5	4	4	5	3	4	5	4	4
1	21 - 30	4	4	2	5	1	5	3	4	2	1
1	10 - 20	5	3	3	3	3	3	3	3	3	3
1	21 - 30	3	5	1	5	5	1	1	5	1	5
1	10 - 20	5	4	3	5	5	5	5	4	5	5
1	10 - 20	4	3	4	2	3	2	2	4	3	2
1	21 - 30	4	2	4	3	2	3	4	3	4	4
1	10 - 20	5	4	4	4	5	5	4	5	4	5
1	10 - 20	4	3	3	3	4	2	4	3	4	4
1	10 - 20	4	3	3	3	4	2	4	3	4	3
1	10 - 20	4	5	3	5	2	4	3	4	3	5
1	21 - 30	4	4	3	4	2	4	5	4	5	2
1	21 - 30	3	4	2	4	1	3	4	2	4	2
1	10 - 20	1	2	3	4	5	4	4	5	5	5
1	10 - 20	3	4	4	3	5	3	4	5	4	2
1	10 - 20	4	3	2	3	5	3	5	2	2	5



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	10 - 20	4	4	5	4	5	4	4	4	4	5
1	21 - 30	4	3	3	4	4	4	4	4	4	4
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	10 - 20	5	5	5	3	5	4	5	5	5	4
1	10 - 20	3	4	3	5	5	3	2	4	5	4
1	10 - 20	3	3	4	3	5	3	4	4	3	3
1	21 - 30	3	3	4	3	4	4	3	2	4	3
1	10 - 20	5	4	4	4	4	4	4	4	4	4
1	10 - 20	2	3	4	4	2	3	3	3	3	3
1	10 - 20	2	3	3	4	4	3	5	2	4	3
1	10 - 20	5	5	5	5	4	5	4	5	5	5
1	10 - 20	4	3	3	2	1	3	1	4	2	2
1	10 - 20	3	3	4	3	2	3	5	3	1	4
1	10 - 20	4	4	4	3	5	5	4	5	4	5
1	10 - 20	4	4	4	3	2	1	4	5	5	4
1	21 - 30	5	5	1	2	1	5	3	3	4	4
1	21 - 30	4	4	4	4	5	3	3	4	5	4
1	10 - 20	3	4	4	3	4	4	3	5	4	3
1	10 - 20	3	3	4	4	3	3	3	3	3	3
1	21 - 30	4	4	3	3	5	5	5	4	5	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
0	10 - 20	5	5	5	5	5	5	5	5	5	5
0	10 - 20	5	5	5	5	5	5	5	5	5	5
1	> 50	4	3	2	4	5	3	3	2	4	5
1	21 - 30	3	4	2	4	5	4	3	5	4	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	3	4	4	4	3	4	2	3	2	3
1	10 - 20	5	5	3	2	5	1	4	4	4	1
1	10 - 20	4	4	5	5	5	5	5	5	4	4
1	10 - 20	4	5	3	5	4	4	5	3	4	5
1	10 - 20	3	5	4	2	4	5	4	2	3	5
1	10 - 20	4	2	4	2	4	4	5	4	3	4
1	31 - 40	4	4	3	5	5	5	5	5	5	5
1	10 - 20	4	3	4	3	5	3	4	2	3	3
1	10 - 20	4	5	5	5	5	5	5	5	5	5
1	10 - 20	3	4	4	3	2	4	5	5	4	3
1	10 - 20	4	4	4	4	4	4	4	4	4	4
1	10 - 20	3	3	4	4	3	5	3	4	4	5
1	31 - 40	3	3	4	3	3	3	3	3	3	4
1	21 - 30	2	2	2	4	1	5	2	1	3	1
1	31 - 40	5	5	5	4	4	4	4	4	5	5
1	21 - 30	4	4	3	5	3	4	4	4	5	5
1	21 - 30	2	4	1	4	4	5	3	3	3	3
1	10 - 20	3	3	3	3	3	3	2	2	2	2
1	10 - 20	2	3	3	4	3	3	3	3	3	4
1	10 - 20	4	4	3	5	4	4	5	4	4	5
1	10 - 20	4	3	4	4	4	3	3	4	5	3
1	10 - 20	3	4	3	3	1	4	4	4	2	4
1	10 - 20	4	4	5	5	3	3	5	4	4	5
1	10 - 20	5	5	5	4	5	5	5	5	5	5
1	10 - 20	4	3	2	3	3	2	2	3	4	1

Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	4	3	4	5	3	5	3	5	1	1
1	10 - 20	3	2	4	3	3	4	4	4	5	3
1	21 - 30	4	4	5	5	3	4	3	3	5	4
1	31 - 40	5	5	4	4	4	4	4	4	4	4
0	> 50	3	4	4	4	4	4	4	4	4	4
1	10 - 20	5	4	3	5	4	4	3	5	5	5
1	21 - 30	4	4	5	5	4	4	4	4	5	5
1	10 - 20	3	5	5	3	4	3	4	5	2	3
0	10 - 20	5	5	5	4	5	5	5	5	5	5
1	10 - 20	4	5	3	3	3	4	4	4	3	4
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	21 - 30	3	3	2	4	4	5	3	3	4	5
1	21 - 30	3	3	3	3	3	3	3	3	3	3
1	10 - 20	5	5	5	5	4	4	4	5	5	5
1	10 - 20	3	3	3	3	3	3	3	3	3	3
0	31 - 40	3	4	5	5	2	3	3	4	5	5
1	10 - 20	3	4	4	3	5	3	3	4	4	3
1	10 - 20	3	3	2	4	2	4	3	1	4	3
1	10 - 20	5	5	4	3	1	1	2	2	3	3
1	10 - 20	4	3	5	2	4	4	4	5	2	4
1	21 - 30	3	4	4	4	4	4	4	4	4	4
1	> 50	4	3	5	4	3	5	2	5	4	2
1	10 - 20	3	3	5	4	4	5	3	5	3	5
1	10 - 20	5	5	5	4	3	2	5	3	4	3
1	10 - 20	4	4	4	4	4	4	5	3	2	3
1	21 - 30	4	3	5	4	5	4	4	4	4	4
1	10 - 20	3	2	5	5	3	3	5	3	5	3
1	10 - 20	3	3	4	3	3	3	3	3	3	3
1	10 - 20	3	5	4	5	3	5	2	5	4	5
1	10 - 20	3	4	4	3	3	4	5	4	4	5
1	21 - 30	4	3	4	4	4	3	4	3	4	3
1	10 - 20	3	4	5	3	5	2	4	2	1	1
1	< 10	3	3	4	2	3	2	2	4	2	3
1	10 - 20	4	5	4	4	4	3	4	4	4	4
1	10 - 20	4	3	4	3	3	5	4	3	2	4
1	10 - 20	3	4	5	5	1	1	1	3	3	4
1	21 - 30	4	5	4	3	5	4	4	5	3	4
1	21 - 30	3	4	3	4	4	4	2	4	3	2
1	10 - 20	4	5	3	5	4	3	5	5	5	5
1	21 - 30	3	3	3	3	3	3	5	3	3	3
1	10 - 20	4	4	4	5	5	4	5	4	5	5
0	10 - 20	4	3	3	3	1	4	3	2	2	3
1	10 - 20	4	4	3	4	4	4	5	3	4	5
1	10 - 20	1	2	2	1	3	2	4	4	4	5
1	21 - 30	3	4	4	4	4	3	5	5	2	5
1	21 - 30	5	4	5	3	4	5	4	4	5	4
1	21 - 30	4	5	4	4	5	2	3	3	3	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	21 - 30	4	3	5	4	4	4	3	5	3	4
1	21 - 30	4	3	4	3	2	3	4	4	3	4
1	10 - 20	3	3	4	3	3	3	3	3	3	3
1	10 - 20	4	4	3	3	3	4	4	4	3	4
1	10 - 20	4	4	3	4	3	2	4	4	5	3



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	5	4	3	4	4	3	5	5	4	4
1	21 - 30	3	3	3	3	3	3	3	3	3	3
1	21 - 30	5	3	5	3	5	5	5	5	5	5
1	21 - 30	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	4	3	4	3	4	4	3	5	4
0	10 - 20	5	3	3	5	2	4	4	4	3	5
1	21 - 30	4	3	3	3	3	3	3	3	3	3
1	10 - 20	3	2	4	3	2	5	2	3	5	4
1	21 - 30	3	4	3	2	4	4	4	3	2	3
1	21 - 30	4	3	5	5	4	4	5	3	4	4
1	10 - 20	3	4	3	2	5	3	3	2	4	1
1	21 - 30	5	5	5	5	5	3	5	5	5	1
1	10 - 20	3	1	2	4	1	5	4	2	3	2
1	21 - 30	3	4	4	4	3	5	5	4	4	3
1	10 - 20	3	3	4	3	1	3	2	4	1	3
1	10 - 20	4	4	5	5	3	4	5	5	4	5
1	10 - 20	5	5	1	1	1	3	4	4	5	5
1	10 - 20	4	4	5	3	3	5	5	4	5	5
1	21 - 30	4	4	4	4	4	4	3	4	4	3
1	10 - 20	5	5	5	5	5	5	4	4	4	4
1	21 - 30	1	1	1	1	1	1	1	1	1	1
1	21 - 30	3	2	4	3	3	4	4	3	3	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	1	5	1	3	4	5	5	5	5
1	10 - 20	4	4	5	4	4	5	4	4	4	4
1	10 - 20	3	3	4	5	5	4	3	3	3	3
1	10 - 20	3	4	3	3	2	4	3	4	5	4
1	10 - 20	5	5	4	4	5	5	5	5	5	5
1	10 - 20	4	4	2	4	4	2	5	2	4	3
1	10 - 20	3	3	2	3	3	3	4	4	4	3
1	10 - 20	4	4	3	3	4	4	5	4	3	4
1	10 - 20	4	5	3	4	4	5	4	3	4	5
1	10 - 20	2	3	3	4	4	4	2	2	4	4
1	10 - 20	4	3	5	4	4	3	5	4	3	4
1	10 - 20	3	4	3	5	4	4	3	4	5	5
1	31 - 40	3	4	3	4	3	4	4	3	4	3
1	10 - 20	4	5	3	4	5	4	3	4	4	5
1	10 - 20	5	5	5	5	5	5	5	1	5	5
1	21 - 30	3	4	5	3	3	3	4	2	1	5
1	21 - 30	1	5	1	3	5	1	2	4	4	1
1	21 - 30	3	3	2	2	2	1	1	1	1	1
1	21 - 30	4	3	1	4	4	2	1	1	3	5
1	10 - 20	4	4	3	5	2	4	5	4	4	3
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	10 - 20	4	5	4	3	4	5	4	2	5	4
1	10 - 20	3	5	4	2	4	3	3	5	1	5
1	10 - 20	1	2	3	5	1	3	4	3	2	3
1	10 - 20	4	4	4	4	4	4	4	4	5	4
1	10 - 20	4	3	3	4	3	3	4	4	5	3
1	10 - 20	4	4	5	5	4	5	5	5	5	5
0	21 - 30	3	4	3	3	4	5	4	5	4	5
1	10 - 20	4	5	3	5	3	2	5	3	4	3
1	21 - 30	3	5	2	3	4	3	4	4	2	2



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	3	3	3	3	3	3	3	3	3	3
1	10 - 20	5	2	4	3	2	3	4	2	2	4
1	31 - 40	3	3	5	4	3	5	4	5	5	4
1	10 - 20	5	5	3	3	3	3	4	2	4	3
1	10 - 20	5	5	5	5	5	5	5	5	5	5
1	31 - 40	5	3	4	4	4	4	4	4	5	5
1	21 - 30	3	3	3	3	3	3	3	3	3	3
1	21 - 30	2	4	3	4	5	3	2	4	4	5
1	10 - 20	3	5	4	4	5	4	4	4	4	5
1	10 - 20	3	4	3	4	4	4	5	3	4	5
1	21 - 30	5	5	4	5	3	4	2	5	3	3
0	< 10	5	5	5	5	5	5	5	5	5	5
1	10 - 20	4	3	4	5	4	4	2	2	4	4
1	21 - 30	4	5	5	3	5	4	5	5	5	4
1	10 - 20	4	5	2	5	4	2	4	3	3	3
1	10 - 20	5	4	3	5	4	3	5	4	4	3
0	21 - 30	4	2	4	5	1	5	5	5	4	4
1	21 - 30	4	4	4	4	4	4	4	4	4	4
1	10 - 20	5	5	5	5	5	5	5	5	5	5
0	> 50	4	4	4	4	4	5	4	4	4	3
1	10 - 20	3	3	3	3	3	4	5	5	5	5
1	21 - 30	3	4	4	4	5	3	3	4	4	4
1	10 - 20	3	4	2	5	2	4	1	2	2	4
1	10 - 20	4	5	1	3	2	5	5	4	5	3
1	21 - 30	4	4	2	5	3	4	3	4	4	5
1	21 - 30	3	3	3	3	4	3	4	4	4	4
1	10 - 20	2	1	4	5	2	3	4	1	2	4
1	21 - 30	3	4	2	3	3	4	3	2	4	4
1	10 - 20	4	4	4	4	4	4	4	4	5	4
1	10 - 20	3	4	5	5	4	3	5	4	4	3
1	10 - 20	3	5	5	5	4	4	4	4	4	5
1	21 - 30	2	3	3	3	2	3	2	2	2	2
1	10 - 20	4	3	5	3	2	4	5	3	4	5
1	10 - 20	4	3	5	3	2	4	5	3	4	5
1	21 - 30	3	4	4	2	3	3	4	2	3	4
1	10 - 20	4	5	4	3	5	4	3	4	5	3
1	10 - 20	3	3	3	3	5	5	5	5	5	5
1	10 - 20	2	3	3	1	2	3	2	3	2	3
1	10 - 20	4	4	5	3	4	4	4	4	4	4
1	21 - 30	3	4	3	3	3	4	4	4	4	4
1	10 - 20	3	4	4	5	5	2	4	5	4	5
1	10 - 20	2	3	4	5	5	5	5	5	5	5
1	21 - 30	5	4	4	4	4	4	4	5	4	4
1	10 - 20	2	3	4	1	3	2	4	1	2	3
1	31 - 40	3	4	3	4	3	2	3	3	2	3
1	21 - 30	5	4	4	3	3	4	4	3	3	4
1	10 - 20	4	4	5	4	4	4	5	4	4	5
1	10 - 20	4	4	5	4	4	4	5	4	4	5
1	21 - 30	3	4	4	5	4	2	4	4	3	5
1	21 - 30	4	4	4	4	4	3	4	2	4	4
1	31 - 40	3	3	3	3	3	3	3	3	3	3
1	21 - 30	3	4	2	5	4	3	4	5	5	5
1	10 - 20	5	5	5	5	5	3	3	4	5	4



Sexo	Edad	1.1 P1(0.3)	1.2 P1(0.9)	2.1 P2(0.7)	2.2 P2(0.1)	3.1 P3(0.9)	3.2 P3(0.5)	4.1 P4(0.1)	4.2 P4(0.5)	5.1 P5(0.9)	5.2 P5(0.1)
1	10 - 20	4	4	5	5	3	5	5	5	5	5
1	10 - 20	5	4	3	3	3	3	3	3	3	3
1	10 - 20	4	5	5	3	3	3	5	5	4	4
1	10 - 20	3	4	3	4	4	4	3	3	4	5

Tabla 32 - Resultados de la encuesta