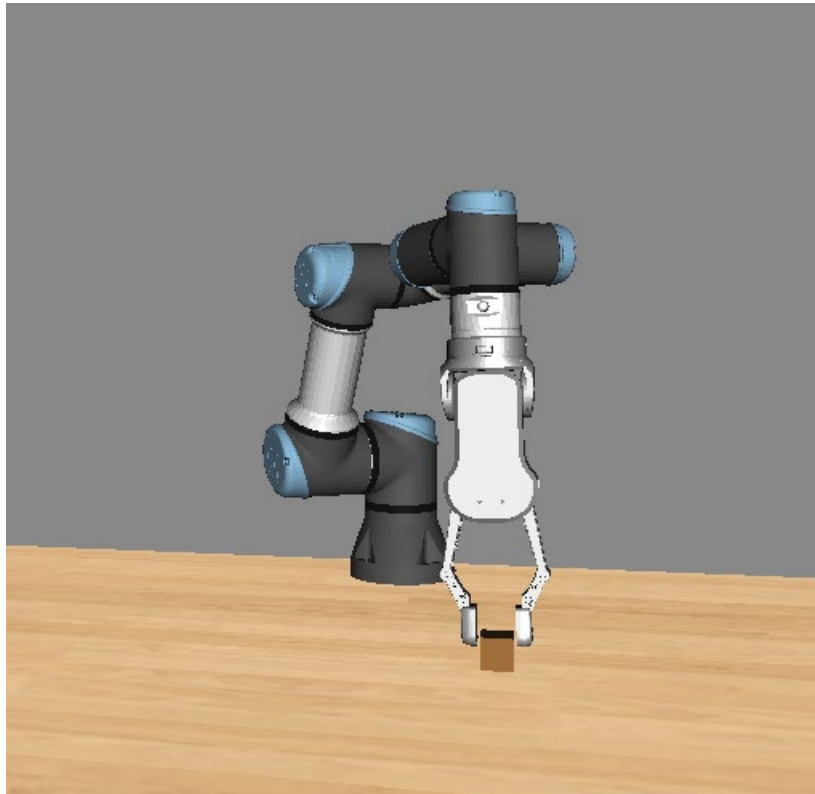


Proyecto Final:

Parte de manipuladores robóticos



Información

- El peso de esta parte del proyecto final en la nota final de la asignatura es de un 15% (50% del 30%).
- La evaluación será mediante una demo final durante el mes de junio (en horario a convenir con los estudiantes). Dicha demo incluirá un turno de preguntas en las que los profesores pueden preguntar a cualquier miembro del grupo sobre cualquier aspecto de la propuesta hecha por los estudiantes.
- El mismo día de la evaluación habrá que entregar todo el código (y, opcionalmente videos de la demo) a través de Atenea.
- A modo orientativo, si P es la nota total de la parte de manipuladores robóticos del proyecto final, $P = 0.6 * D + 0.4 * TP$, donde D es la nota de la demo real (y en caso de que falle, de los videos y código subidos el mismo día) y TP es la nota de cada alumno en el turno de preguntas.

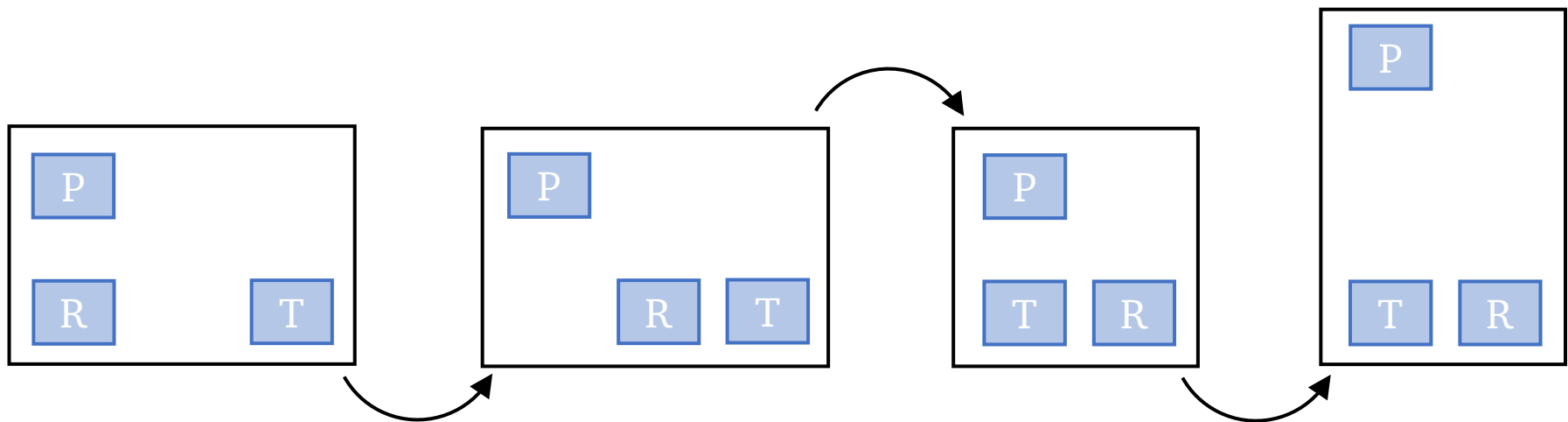
Información

El proyecto final consiste en programar un robot manipulador para ejecute una cierta tarea de manipulación. Los pasos a seguir son los siguientes:

1. Escoger una de las siguientes 10 tareas (uno por grupo final).
2. Crear los *domain* y *problem files* (PDDL) asociados a la tarea de manipulación escogida.
3. Resolver el problema de planificación de tareas y movimientos asociado al PDDL *problem file* tal como se ha visto en las distintas sesiones prácticas. La parte de planificación de movimientos se tiene que resolver con distintos planificadores. Comparar en simulación las soluciones obtenidas usando *The Kautam Project*.
4. BONUS (Opcional) → Hacer un benchmarking con los planificadores usados.
5. Implementar y ejecutar la solución obtenida en el robot real (UR3e).

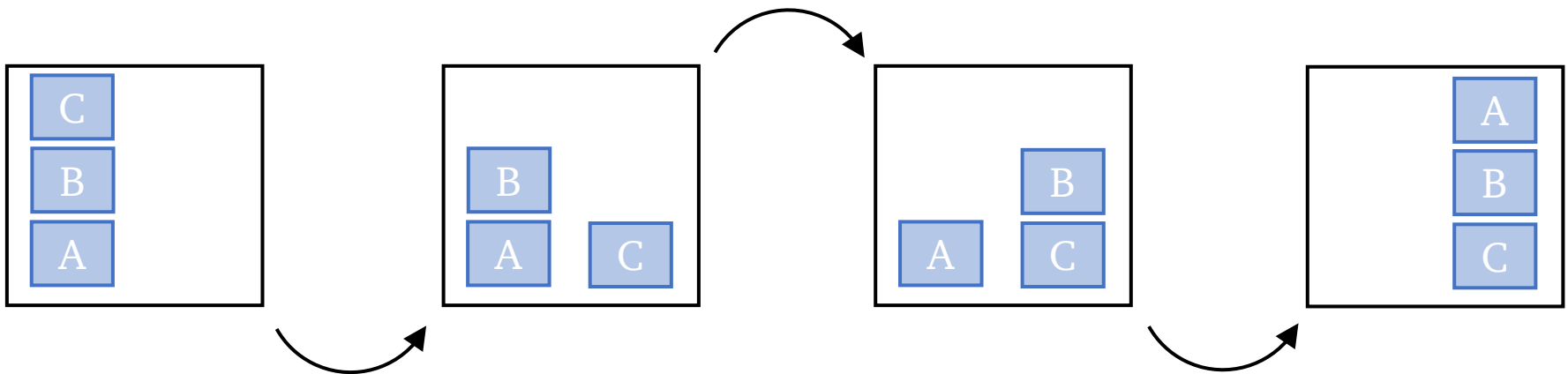
Tarea 1

Hacer un enroque con dos piezas (que simulan la torre y el rey) y mover un tercera que simule un peón.



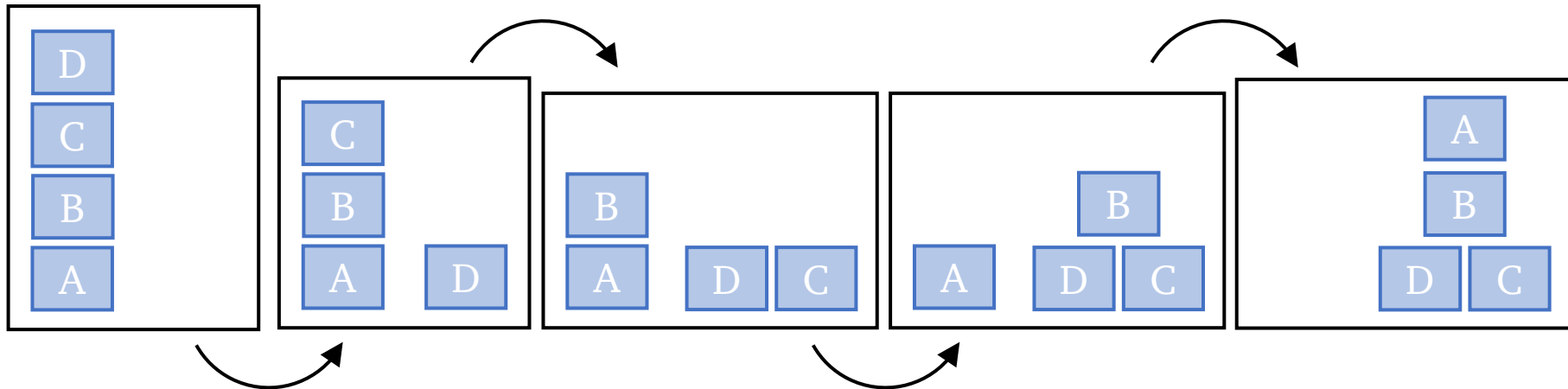
Tarea 2

Resolver el problema de las torres de Hanoi simplificado con tres piezas.



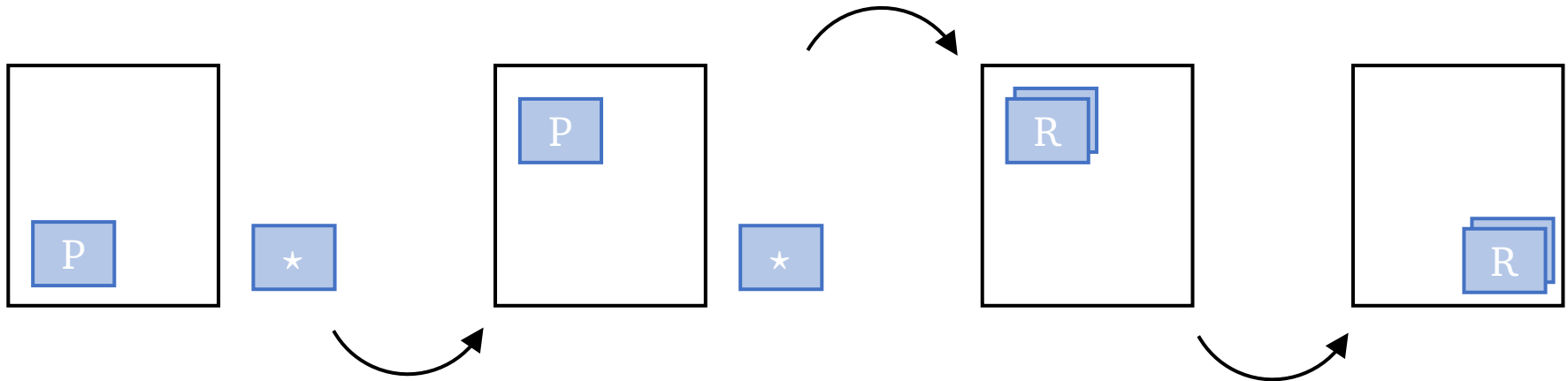
Tarea 3

Montar un triángulo con 4 piezas.



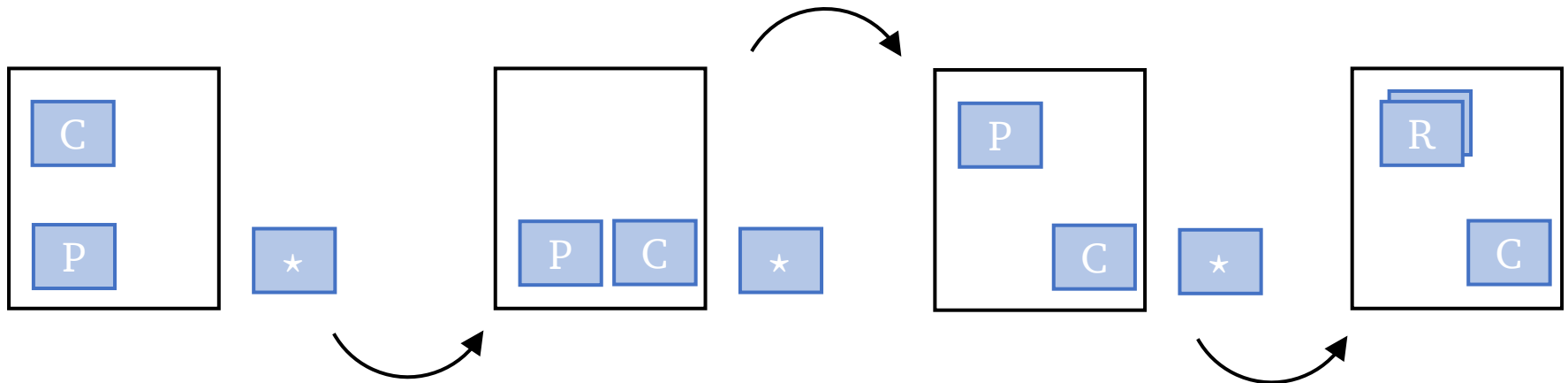
Tarea 4

Convierte una pieza simulando un peón en una reina y muévete en diagonal.



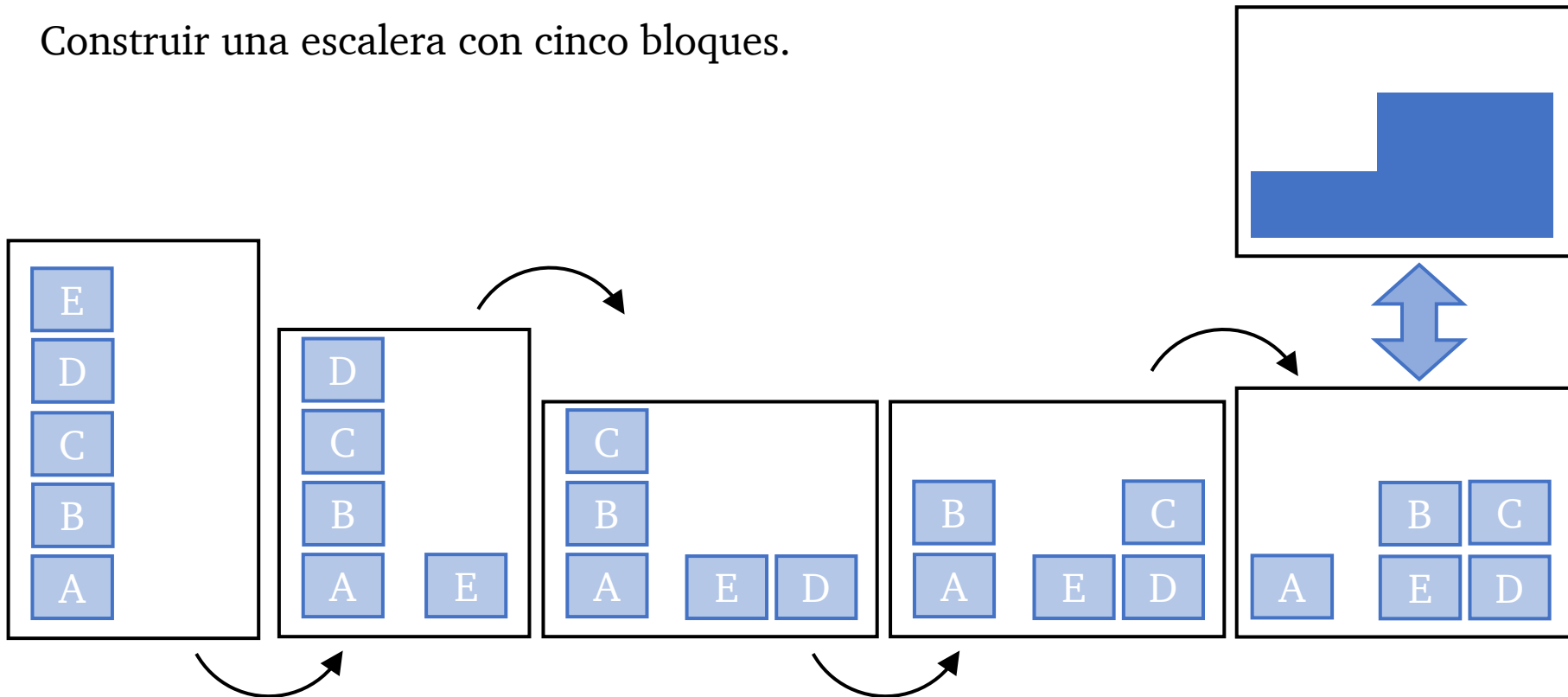
Tarea 5

Convierte una pieza simulando un peón en una reina con un pieza simulando un caballo en medio de su camino.



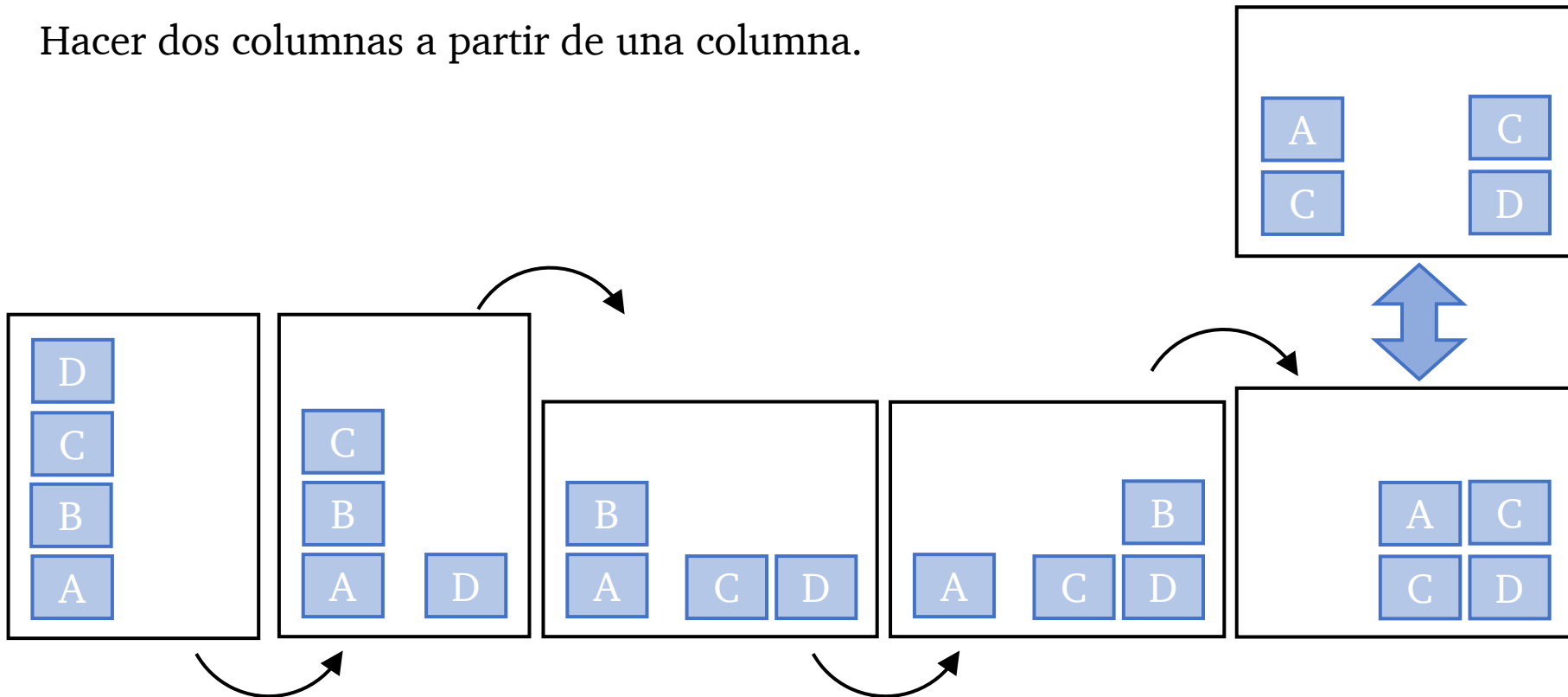
Tarea 6

Construir una escalera con cinco bloques.



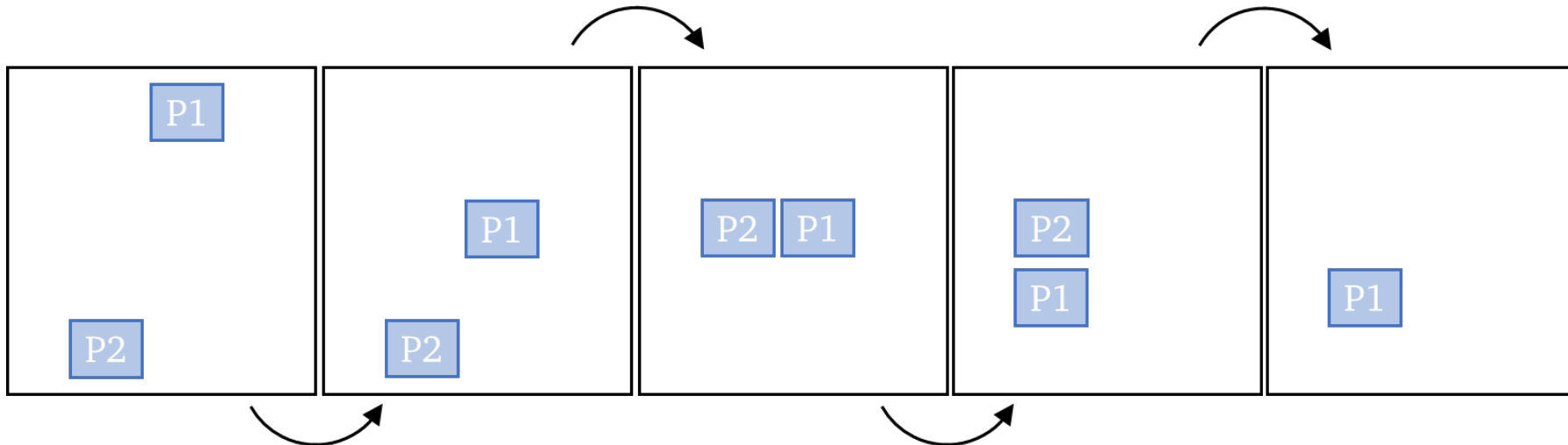
Tarea 7

Hacer dos columnas a partir de una columna.



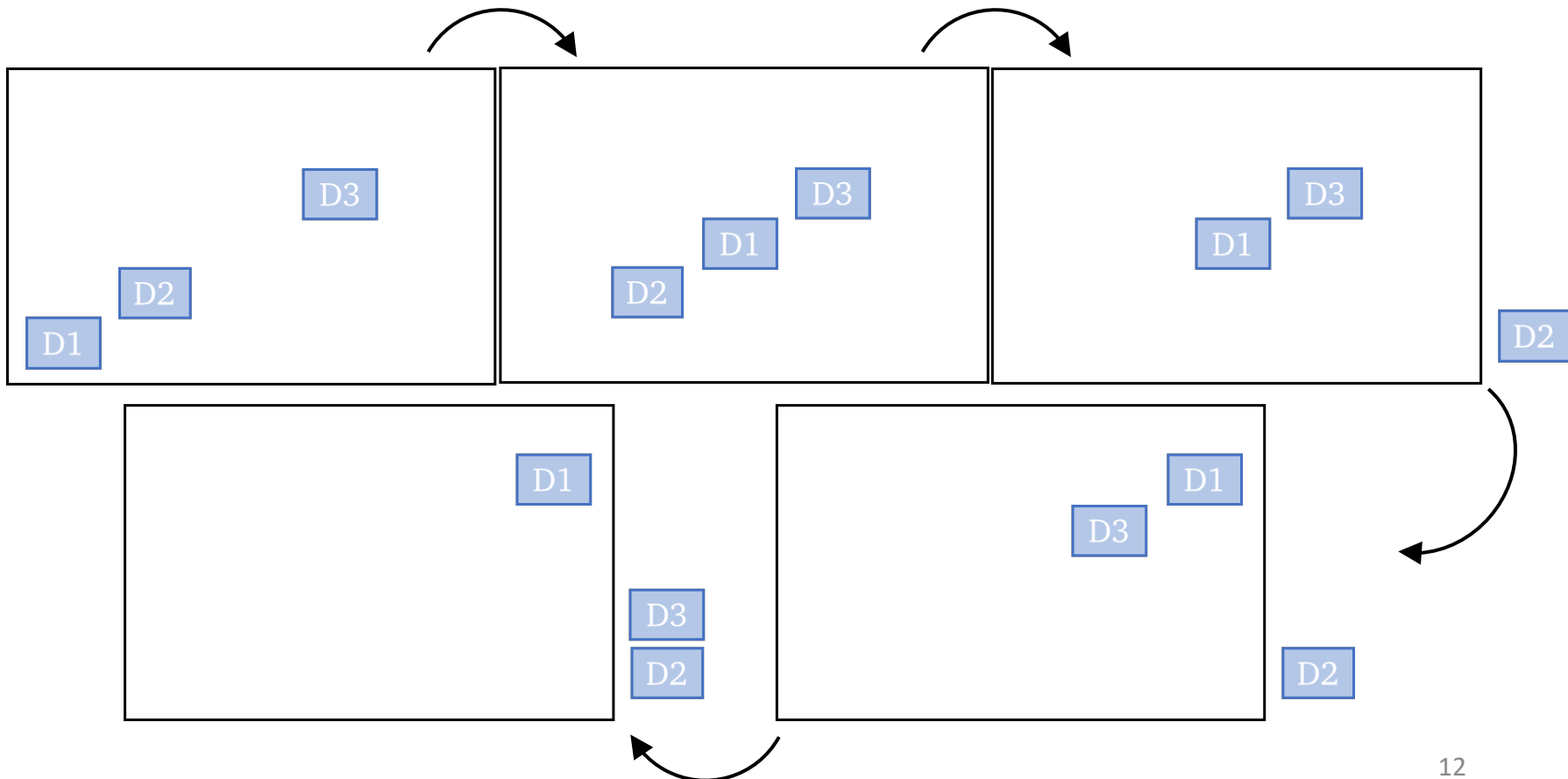
Tarea 8

Captura una pieza que simula un peón.



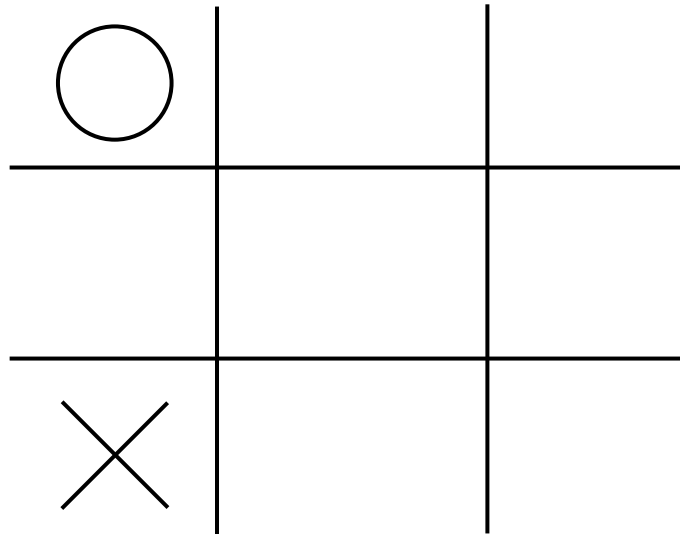
Tarea 9

Mover una pieza simulando una dama que se coma dos piezas diferentes que están en la misma diagonal.



Tarea 10

Mueven 'O'. Gana la partida con 'X' en 5 movimientos (si se quiere, se puede pedir la solución con todos los pasos intermedios).



Instrucciones extra

1. **Obstáculos y objetos a manipular** → En los ficheros de *The Kautham Project*, los obstáculos (lo que incluye a los objetos con los que hay que hacer pick-and-place) vienen descritos con una posición y orientación en el siguiente formato (TH, WZ, WY, WX, Z, Y, X). Este formato es, para la parte de orientación, un eje-ángulo, es decir, un eje de rotación, con vector director (WX, WY, WZ) y un ángulo de rotación con valor TH. Para la parte de posición, son las tres coordenadas espaciales X, Y, Z. Usad las formulas que podéis encontrar en la bibliografía de la asignatura (Siciliano et al. *Robotics modelling, planning and control*, capítulo 2) o en internet.
2. **Sim2Real** → En *The Kautham Project*, el robot tiene el tablero y objetos enfrente de él, pero en los UR3e, enfrente hay otra mesa y otro robot (o una pared). Por ello, es mejor cambiar los objetos a manipular y los objetos estáticos para que estén al lado izquierdo del robot y que este los manipule allí. Así será más fácil pasar de la simulación al experimento real.

Instrucciones extra

3. **Fichero `tamconfig`** (mas info en el guion de prácticas P5) → Cuando se resuelve la planificación de tareas, y pasamos a hacer la planificación de movimientos con *Kautham*, hay que modificar el fichero `tamconfig`. Allí encontrareis:
- a) Una primera parte, llamada `Problemfiles`, donde se llama al planificador de movimientos, los PDDLs, etc.
 - b) Una segunda parte, llamada `States`, donde se incluyen los distintos objetos que intervendrán en la manipulación. Allí se define también su posición y su posición como control (en escala cero-uno). Dicha posición se ha de cambiar también en el *problem file* de *Kautham*, `OMPL_RRT_...`
 - c) Una tercera parte, llamada `Actions`, donde se incluyen las dos acciones, *pick*, y *place*, que el robot puede ejecutar. Ahí encontrareis dos tipos de control, `Regioncontrols` y `Graspcontrols`. Los primeros se usan para definir configuraciones del robot, en particular, la inicial y final, mientras que los segundos se utilizan para definir las configuraciones en las que, cerrando o abriendo la pinza, se ejecutan las acciones *pick* y *place*. Como ambas son variables de tipo control, solo aceptan números entre cero y uno.

Instrucciones extra

¿Cómo encontrar estos números? Hay dos maneras:

1. Coger una configuración del UR3e real, pasarla a radianes, y normalizarla en el rango $[-\pi, \pi]$. Es decir, cada componente de la configuración se normaliza como:

$$q_i^* = \frac{(q_i + \pi)}{2\pi}$$

2. Abrir el *Kautham*, ir a las barras laterales que hay a la izquierda de la pantalla principal, encontrar la barra de controles, y con los botones encontrar una configuración inicial, una final, una de pick, y una de place. Se pueden copiar y pegar estos valores, ya en formato control (es decir, entre cero y uno) en la parte correspondiente del fichero `tampconfig`.