

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Laboratorio: Planificación de STRIPS

Objetivos de la actividad

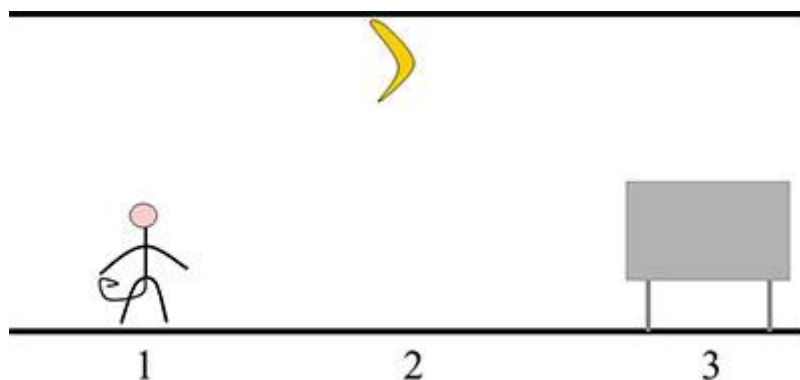
Con esta actividad vas a conseguir implementar un algoritmo de planificación basado en STRIPS.

Descripción de la actividad

Expresa el siguiente escenario en la representación tipo STRIPS:

En una habitación hay un mono, una caja y un plátano, tal como indica la figura (situación inicial). El objetivo del mono es tener el plátano. El mono puede:

- ▶ Ir de una posición a otra.
- ▶ Empujar la caja de una posición a otra si está en la misma posición que ella y no está sobre ella.
- ▶ Subirse a la caja si está en la misma posición que ella.
- ▶ Coger el plátano si está encima de la caja.



Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

El dominio del problema se modela con cuatro objetos, los cuartos o espacios en los que se van a desplazar la caja y el mono, la caja, el mono y las bananas. Con esta idea se modela el movimiento de la caja como que el mono la “cargase” para poder desplazarse de un cuarto a otro y no caer en el problema semántico de arrastrar o empujar el objeto.

Así mismo las bananas tienen la propiedad única de estar libres o no, esta propiedad (predicado) cambia cuando el mono las haya tomado. No se crea una situación en la que puedan ser liberadas de nuevo ya que en los problemas no se considera esta meta.

Dominio PDDL

```
;Autor : Daniel Proaño Guevara
;Asignatura : Razonamiento y Planificación

(define (domain monoBanana)

  (:requirements :strips :typing :negative-preconditions)

  (:types
    mono caja banana cuarto -object
  )

  (:predicates
    (enlace-cuarto ?c1 -cuarto ?c2 -cuarto)
    (loc-mono ?m -mono ?c1 -cuarto)
    (loc-caja ?c -caja ?c1 -cuarto)
    (loc-banana ?b -banana ?c1 -cuarto)
    (subir-mono ?mono -mono)
    (libre ?l)
  )

;; El mono puede cambiar de habitación
(:action mover-mono
  :parameters (?mono -mono ?cuartoOrigen -cuarto ?cuartoDestino -cuarto)
  :precondition (and
    (enlace-cuarto ?cuartoOrigen ?cuartoDestino)
```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

```

        (loc-mono ?mono ?cuartoOrigen)
        (not(subir-mono ?mono))
    )
    :effect (and
        (not (loc-mono ?mono ?cuartoOrigen))
        (loc-mono ?mono ?cuartoDestino)
    )
)

;;El mono toma la caja y se mueve con ella para poder cambiarla de habitación
(:action cargar-caja
    :parameters (?mono -mono ?caja -caja ?cuartoOrigen -cuarto ?cuartoDestino -cuarto)
    :precondition (and
        (enlace-cuarto ?cuartoOrigen ?cuartoDestino)
        (loc-mono ?mono ?cuartoOrigen)
        (loc-caja ?caja ?cuartoOrigen)
        (not(subir-mono ?mono))
    )
    :effect (and
        (not(loc-caja ?caja ?cuartoOrigen))
        (loc-caja ?caja ?cuartoDestino)
        (not (loc-mono ?mono ?cuartoOrigen))
        (loc-mono ?mono ?cuartoDestino)
    )
)

;;El mono puede subir a la caja siempre y cuando se encuentre en el suelo
(:action subir-caja
    :parameters (?mono -mono ?caja -caja ?cuarto -cuarto)
    :precondition (and
        (loc-caja ?caja ?cuarto)
        (loc-mono ?mono ?cuarto)
        (not(subir-mono ?mono))
    )
    :effect (and
        (subir-mono ?mono)
    )
)

;; El mono puede regresar a su estado original en el suelo

```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

```
(:action bajar-caja
  :parameters (?mono -mono ?caja -caja ?cuarto -cuarto)
  :precondition (and
    (loc-caja ?caja ?cuarto)
    (loc-mono ?mono ?cuarto)
    (subir-mono ?mono)
  )
  :effect (and
    (not (subir-mono ?mono))
  )
)

;;Las bananas pueden ser tomadas
(:action cojer-banana
  :parameters (?mono -mono ?banana -banana ?cuarto -cuarto)
  :precondition (and
    (loc-mono ?mono ?cuarto)
    (loc-banana ?banana ?cuarto)
    (subir-mono ?mono)
    (libre ?banana)
  )
  :effect (and
    (not (libre ?banana))
  )
)
)
```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Problema 1: Solución al problema planteado

```

;Autor : Daniel Proaño Guevara
;Asignatura : Razonamiento y Planificación
(define (problem problema1) (:domain monoBanana)
  (:objects
    mono1 -mono
    cuarto1 cuarto2 cuarto3 -cuarto
    caja1 -caja
    banana1 -banana
  )
  (:init
    ;; Disposición del espacio de movilidad para el mono
    (enlace-cuarto cuarto1 cuarto2)
    (enlace-cuarto cuarto2 cuarto1)

    (enlace-cuarto cuarto2 cuarto3)
    (enlace-cuarto cuarto3 cuarto2)

    ;; Ubicación original del mono
    (loc-mono mono1 cuarto1)
    (not(subir-mono mono1))

    ;; Ubicación original de la caja
    (loc-caja caja1 cuarto3)

    ;; Ubicación original de la banana
    (loc-banana banana1 cuarto2)
    (libre banana1)
  )
  (:goal (and
    (not(libre banana1))
  ))
)

```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Salida1

La salida se obtuvo con el planificador <http://solver.planning.domains/solve>

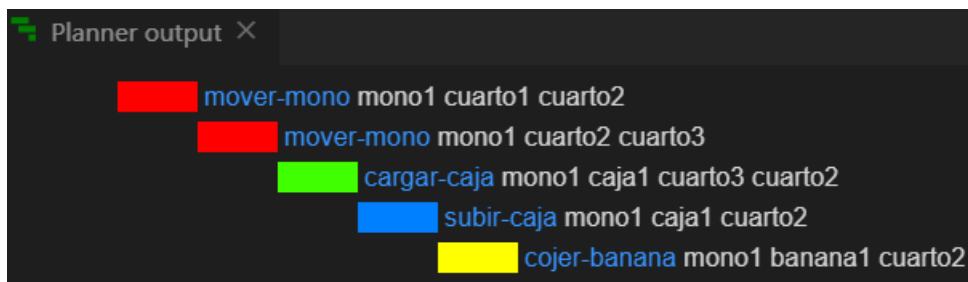
```

Planning service: http://solver.planning.domains/solve
Domain: monoBanana, Problem: problema1
--- OK.
Match tree built with 15 nodes.

PDDL problem description loaded:
  Domain: MONOBANANA
  Problem: PROBLEMA1
  #Actions: 15
  #Fluents: 10
Landmarks found: 1
Starting search with IW (time budget is 60 secs)...
rel_plan size: 4
#RP_fluents 4
Caption
{#goals, #UNnachieved, #Achieved} -> IW(max_w)

{1/1/0}:IW(1) -> [2][3][4][5][6]rel_plan size: 0
#RP_fluents 0Plan found with cost: 5
Total time: -3.7998e-10
Nodes generated during search: 12
Nodes expanded during search: 10
IW search completed
0.00100: (mover-mono mono1 cuarto1 cuarto2)
0.00200: (mover-mono mono1 cuarto2 cuarto3)
0.00300: (cargar-caja mono1 caja1 cuarto3 cuarto2)
0.00400: (subir-caja mono1 caja1 cuarto2)
0.00500: (cojer-banana mono1 banana1 cuarto2)
Planner found 1 plan(s) in 0.574secs.

```



```

Planner output X
  mover-mono mono1 cuarto1 cuarto2
  mover-mono mono1 cuarto2 cuarto3
  cargar-caja mono1 caja1 cuarto3 cuarto2
  subir-caja mono1 caja1 cuarto2
  cojer-banana mono1 banana1 cuarto2

```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Salida con planificador PDDL4J

```
C:\Gradle\pddl4j-master>gradlew run -PArgs=-o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem1.pddl
```

> Configure project :

```
arguments          list          :          -o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem1.pddl
```

> Task :run

parsing domain file "domain.pddl" done successfully

parsing problem file "problem1.pddl" done successfully

encoding problem done successfully (15 ops, 8 facts)

* starting A*

* A* succeeded

found plan as follows:

0: (mover-mono mono1 cuarto1 cuarto2) [1]

1: (mover-mono mono1 cuarto2 cuarto3) [1]

2: (cargar-caja mono1 caja1 cuarto3 cuarto2) [1]

3: (subir-caja mono1 caja1 cuarto2) [1]

4: (cojer-banana mono1 banana1 cuarto2) [1]

plan total cost: 5.00

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

time spent: 0.08 seconds parsing

0.03 seconds encoding

0.01 seconds searching

0.11 seconds total time

memory used: 0.01 MBytes for problem representation

0.00 MBytes for searching

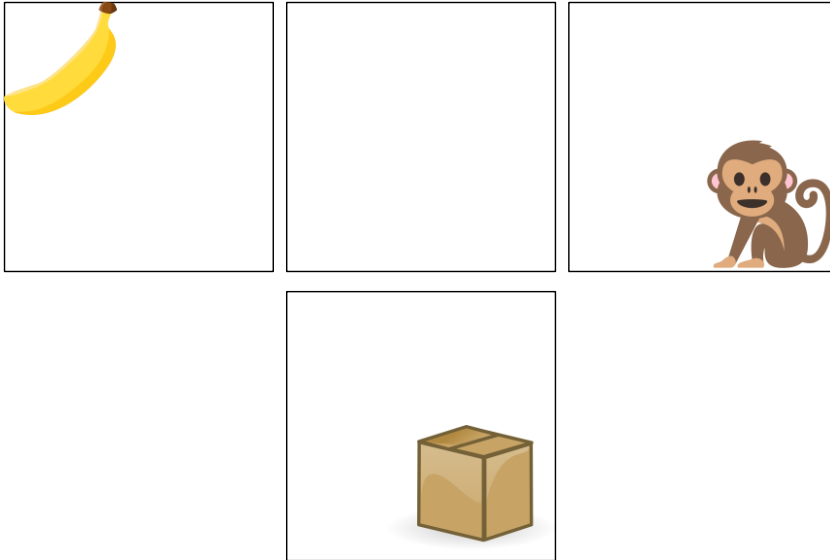
0.01 MBytes total

```
C:\Gradle\pddl4j-master>gradlew run -PArgs=-o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem1.pddl
> Configure project :
arguments list : -o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem1.pddl
> Task :run
parsing domain file "domain.pddl" done successfully
parsing problem file "problem1.pddl" done successfully
encoding problem done successfully (15 ops, 8 facts)
* starting A*
* A* succeeded
found plan as follows:
0: ( mover-mono mono1 cuarto1 cuarto2) [1]
1: ( mover-mono mono1 cuarto2 cuarto3) [1]
2: ( cargar-caja mono1 cajal cuarto3 cuarto2) [1]
3: ( subir-caja mono1 cajal cuarto2) [1]
4: ( cojer-banana mono1 bananal cuarto2) [1]
plan total cost: 5.00
time spent:      0.08 seconds parsing
                0.03 seconds encoding
                0.01 seconds searching
                0.11 seconds total time
memory used:    0.01 MBytes for problem representation
                0.00 MBytes for searching
                0.01 MBytes total
```

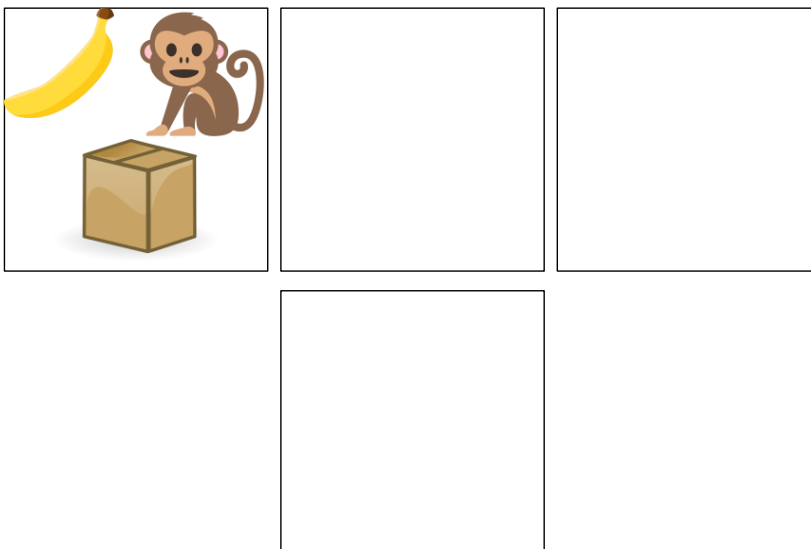

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Problema 2

Inicio:



Meta:



```
;Autor : Daniel Proaño Guevara
;Asignatura : Razonamiento y Planificación
(define (problem problema2) (:domain monoBanana)
(:objects
  mono1 -mono
  cuarto1 cuarto2 cuarto3 cuarto4 -cuarto
  caja1 -caja
```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

```

    banana1 -banana
)

(:init
  ;; Disposición del espacio de movilidad para el mono
  (enlace-cuarto cuarto1 cuarto2)
  (enlace-cuarto cuarto2 cuarto1)

  (enlace-cuarto cuarto2 cuarto3)
  (enlace-cuarto cuarto3 cuarto2)

  (enlace-cuarto cuarto2 cuarto4)
  (enlace-cuarto cuarto4 cuarto2)

  ;; Ubicación original del mono
  (loc-mono mono1 cuarto1)
  (not(subir-mono mono1))

  ;; Ubicación original de la caja
  (loc-caja caja1 cuarto4)

  ;; Ubicación original de la banana
  (loc-banana banana1 cuarto3)
  (libre banana1)
)

(:goal (and
  (not(libre banana1))
))

)

```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Salida 2

La salida se obtuvo con el planificador <http://solver.planning.domains/solve>

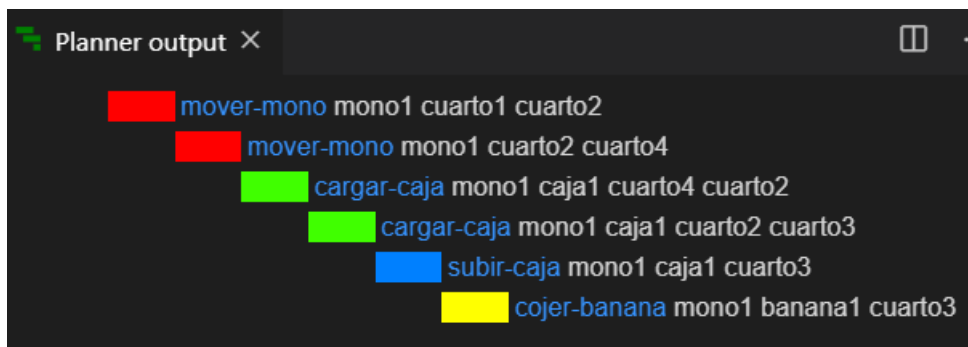
```

Planning service: http://solver.planning.domains/solve
Domain: monoBanana, Problem: problema2
--- OK.
Match tree built with 21 nodes.

PDDL problem description loaded:
  Domain: MONOBANANA
  Problem: PROBLEMA2
  #Actions: 21
  #Fluents: 12
Landmarks found: 1
Starting search with IW (time budget is 60 secs)...
rel_plan size: 5
#RP_fluents 5
Caption
{#goals, #UNnachieved, #Achieved} -> IW(max_w)

{1/1/0}:IW(1) -> [2][3][4][5][6][7]rel_plan size: 0
#RP_fluents 0Plan found with cost: 6
Total time: 4.47035e-10
Nodes generated during search: 19
Nodes expanded during search: 17
IW search completed
0.00100: (mover-mono mono1 cuarto1 cuarto2)
0.00200: (mover-mono mono1 cuarto2 cuarto4)
0.00300: (cargar-caja mono1 caja1 cuarto4 cuarto2)
0.00400: (cargar-caja mono1 caja1 cuarto2 cuarto3)
0.00500: (subir-caja mono1 caja1 cuarto3)
0.00600: (cojer-banana mono1 banana1 cuarto3)
Planner found 1 plan(s) in 0.837secs.

```



```

Planner output X
  mover-mono mono1 cuarto1 cuarto2
  mover-mono mono1 cuarto2 cuarto4
    cargar-caja mono1 caja1 cuarto4 cuarto2
    cargar-caja mono1 caja1 cuarto2 cuarto3
      subir-caja mono1 caja1 cuarto3
      cojer-banana mono1 banana1 cuarto3

```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Salida con planificador PDDL4J

```
C:\Gradle\pddl4j-master>gradlew run -PArgs=-o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem2.pddl
```

> Configure project :

```
arguments          list          :          -o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem2.pddl
```

> Task :run

parsing domain file "domain.pddl" done successfully

parsing problem file "problem2.pddl" done successfully

encoding problem done successfully (21 ops, 10 facts)

* starting A*

* A* succeeded

found plan as follows:

0: (mover-mono mono1 cuarto1 cuarto2) [1]

1: (mover-mono mono1 cuarto2 cuarto4) [1]

2: (cargar-caja mono1 caja1 cuarto4 cuarto2) [1]

3: (cargar-caja mono1 caja1 cuarto2 cuarto3) [1]

4: (subir-caja mono1 caja1 cuarto3) [1]

5: (cojer-banana mono1 banana1 cuarto3) [1]

plan total cost: 6.00

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

time spent: 0.07 seconds parsing

0.03 seconds encoding

0.01 seconds searching

0.11 seconds total time

memory used: 0.01 MBytes for problem representation

0.00 MBytes for searching

0.02 MBytes total

```
C:\Gradle\pddl4j-master>gradlew run -PArgs=-o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem2.pddl
> Configure project :
arguments list : -o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem2.pddl
> Task :run
parsing domain file "domain.pddl" done successfully
parsing problem file "problem2.pddl" done successfully
encoding problem done successfully (21 ops, 10 facts)
* starting A*
* A* succeeded
found plan as follows:
0: ( mover-mono mono1 cuarto1 cuarto2) [1]
1: ( mover-mono mono1 cuarto2 cuarto4) [1]
2: ( cargar-caja mono1 cajal cuarto4 cuarto2) [1]
3: ( cargar-caja mono1 cajal cuarto2 cuarto3) [1]
4: ( subir-caja mono1 cajal cuarto3) [1]
5: ( cojer-banana mono1 bananal cuarto3) [1]
plan total cost: 6.00

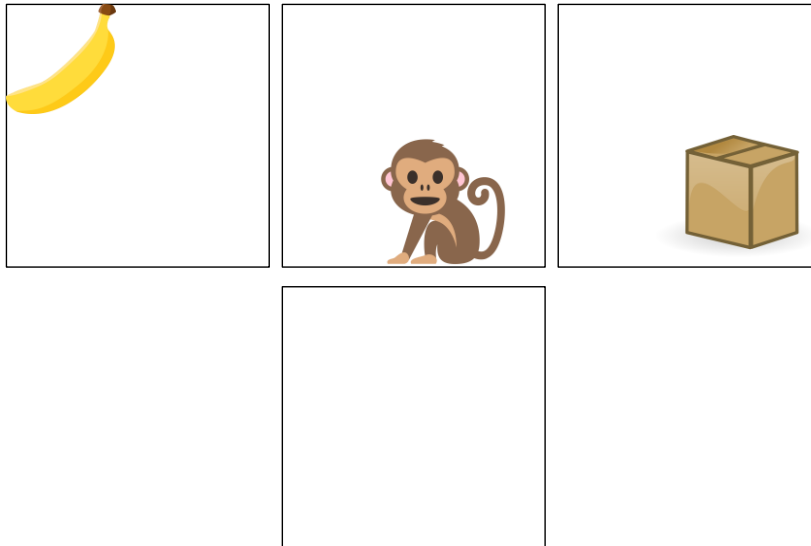
time spent:      0.07 seconds parsing
                0.03 seconds encoding
                0.01 seconds searching
                0.11 seconds total time

memory used:     0.01 MBytes for problem representation
                0.00 MBytes for searching
                0.02 MBytes total
```

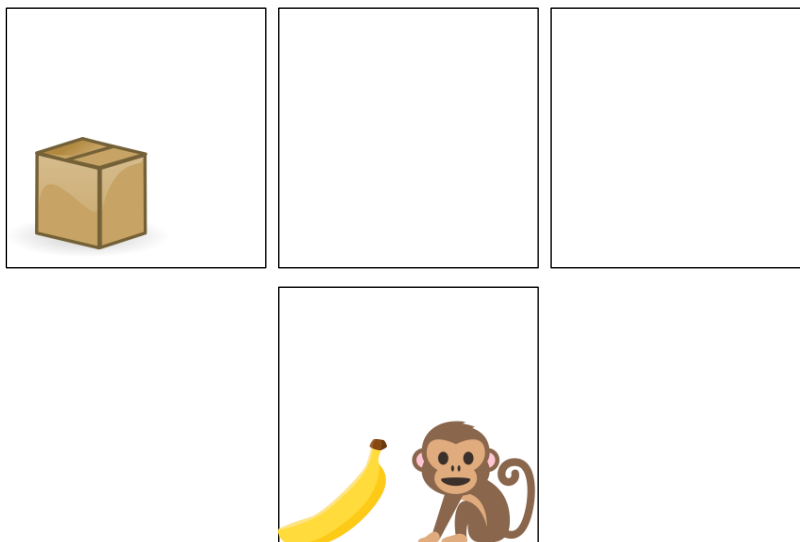
Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Problema 3

Inicio:



Meta:



```
;Autor : Daniel Proaño Guevara
;Asignatura : Razonamiento y Planificación
(define (problem problema3) (:domain monoBanana)
  (:objects
    mono1 -mono
    cuarto1 cuarto2 cuarto3 cuarto4 -cuarto
    caja1 -caja
    banana1 -banana
  )
)
```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

```
(:init
  ;; Disposición del espacio de movilidad para el mono
  (enlace-cuarto cuarto1 cuarto2)
  (enlace-cuarto cuarto2 cuarto1)

  (enlace-cuarto cuarto2 cuarto3)
  (enlace-cuarto cuarto3 cuarto2)

  (enlace-cuarto cuarto2 cuarto4)
  (enlace-cuarto cuarto4 cuarto2)

  ;; Ubicación original del mono
  (loc-mono mono1 cuarto2)
  (not(subir-mono mono1))

  ;; Ubicación original de la caja
  (loc-caja caja1 cuarto1)

  ;; Ubicación original de la banana
  (loc-banana banana1 cuarto3)
  (libre banana1)
)

(:goal (and
  (not(libre banana1))
  (loc-mono mono1 cuarto4)
))

)
```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Salida 3

La salida se obtuvo con el planificador <http://solver.planning.domains/solve>

```

Planning service: http://solver.planning.domains/solve
Domain: monoBanana, Problem: problema3
--- OK.
Match tree built with 21 nodes.

PDDL problem description loaded:
  Domain: MONOBANANA
  Problem: PROBLEMA3
  #Actions: 21
  #Fluents: 12
Landmarks found: 2
Starting search with IW (time budget is 60 secs)...
rel_plan size: 5
#RP_fluents 5
Caption
{#goals, #UNnachieved, #Achieved} -> IW(max_w)

{2/2/0}:IW(1) -> [2][3][4][5][6]rel_plan size: 3
#RP_fluents 4
{2/1/1}:IW(1) -> [2][3][4]rel_plan size: 0
#RP_fluents 0Plan found with cost: 8
Total time: -1.04308e-10
Nodes generated during search: 24
Nodes expanded during search: 17
IW search completed
0.00100: (mover-mono mono1 cuarto2 cuarto1)
0.00200: (cargar-caja mono1 caja1 cuarto1 cuarto2)
0.00300: (cargar-caja mono1 caja1 cuarto2 cuarto3)
0.00400: (subir-caja mono1 caja1 cuarto3)
0.00500: (cojer-banana mono1 banana1 cuarto3)
0.00600: (bajar-caja mono1 caja1 cuarto3)
0.00700: (mover-mono mono1 cuarto3 cuarto2)
0.00800: (mover-mono mono1 cuarto2 cuarto4)
Planner found 1 plan(s) in 1.531secs.

```


Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

```

Planner output X
  mover-mono mono1 cuarto2 cuarto1
    cargar-caja mono1 caja1 cuarto1 cuarto2
      cargar-caja mono1 caja1 cuarto2 cuarto3
        subir-caja mono1 caja1 cuarto3
          cojer-banana mono1 banana1 cuarto3
            bajar-caja mono1 caja1 cuarto3
              mover-mono mono1 cuarto3 cuarto2
                mover-mono mono1 cuarto2 cuarto4

```

Salida con planificador PDDL4J

```
C:\Gradle\pddl4j-master>gradlew run -PArgs=-o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem3.pddl
```

> Configure project :

```
arguments list : -o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem3.pddl
```

> Task :run

parsing domain file "domain.pddl" done successfully

parsing problem file "problem3.pddl" done successfully

encoding problem done successfully (21 ops, 10 facts)

* starting A*

* A* succeeded

found plan as follows:

0: (mover-mono mono1 cuarto2 cuarto1) [1]

1: (cargar-caja mono1 caja1 cuarto1 cuarto2) [1]

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

2: (cargar-caja mono1 caja1 cuarto2 cuarto3) [1]

3: (subir-caja mono1 caja1 cuarto3) [1]

4: (cojer-banana mono1 banana1 cuarto3) [1]

5: (bajar-caja mono1 caja1 cuarto3) [1]

6: (mover-mono mono1 cuarto3 cuarto2) [1]

7: (mover-mono mono1 cuarto2 cuarto4) [1]

plan total cost: 8.00

time spent: 0.08 seconds parsing

0.03 seconds encoding

0.01 seconds searching

0.11 seconds total time

memory used: 0.01 MBytes for problem representation

0.00 MBytes for searching

0.02 MBytes total

```
C:\Gradle\pddl4j-master>gradlew run -PArgs=-o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem3.pddl
> Configure project :
arguments list : -o,pddl/Laboratorio/domain.pddl,-f,pddl/Laboratorio/problem3.pddl
> Task :run
parsing domain file "domain.pddl" done successfully
parsing problem file "problem3.pddl" done successfully
encoding problem done successfully (21 ops, 10 facts)
* starting A*
* A* succeeded
found plan as follows:
0: ( mover-mono mono1 cuarto2 cuarto1) [1]
1: ( cargar-caja mono1 cajal cuarto1 cuarto2) [1]
2: ( cargar-caja mono1 cajal cuarto2 cuarto3) [1]
3: ( subir-caja mono1 cajal cuarto3) [1]
4: ( cojer-banana mono1 bananal cuarto3) [1]
5: ( bajar-caja mono1 cajal cuarto3) [1]
6: ( mover-mono mono1 cuarto3 cuarto2) [1]
7: ( mover-mono mono1 cuarto2 cuarto4) [1]
plan total cost: 8.00
time spent:      0.08 seconds parsing
                0.03 seconds encoding
                0.01 seconds searching
                0.11 seconds total time
memory used:    0.01 MBytes for problem representation
                0.00 MBytes for searching
                0.02 MBytes total
BUILD SUCCESSFUL in 2s
10 actionable tasks: 1 executed, 9 up-to-date
C:\Gradle\pddl4j-master>
```

Asignatura	Datos del alumno	Fecha
Razonamiento y Planificación Automática	Apellidos: Proaño Guevara	02/27/2020
	Nombre: Daniel	

Dificultades

La principal dificultad que me encontré fue la imposibilidad de ejecutar el planeador LPG-TD, tanto en raspberry como en una terminal de Debian, así mismo como el planificador en internet <https://stripsfiddle.herokuapp.com/>

Según la documentación realizada era necesario cambiar los objetos del dominio por constantes, pero ya que fue posible ejecutar el planificador PDDL4J ya no se consideró necesario modificar todo el código de planificación para que corra en este servicio.

Referencias

Dolejsi, J. (n.d.). Configuring the PDDL planner. Retrieved February 27, 2020, from <https://github.com/jan-dolejsi/vscode-pddl/wiki/Configuring-the-PDDL-planner>

Fernando Berzal, F., Richard Fikes, R., Edwin Pednault, E., & McDermott, D. (n.d.). *PDDL*. Granada. Retrieved from <http://ipc.icapshttp://ipc.icaps--conference.org/conference.org/>

Helmert, M. (n.d.). *What is PDDL?* Retrieved from <https://www.cs.toronto.edu/~sheila/2542/s14/A1/introtopddl2.pdf>

Pellier, D., & Fiorino, H. (2018). PDDL4J: a planning domain description library for java. *Journal of Experimental and Theoretical Artificial Intelligence*, 30(1), 143–176. <https://doi.org/10.1080/0952813X.2017.1409278>