Inf 2D Coursework 2

Planning in PDDL

Important Dates

- Deadline : 3pm Thursday 25th March 2021
- Live online Q&A: Fridays 11 am–1pm

Coursework Goals

- Formalize a reasonably sized planning problem
- Balance trade-offs in model design
- Actually implement and debug some PDDL

	L 		
	A		
	B	C	D 
F	E 		

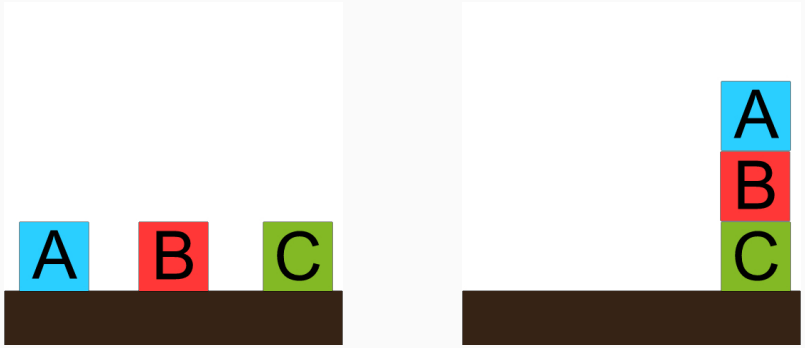
Assignment Outline

- Part 1a - Formalize problem in PDDL
- Part 1b - Backward State Space Search
- Parts 2 and 3 - Implement and extend model in PDDL for MetricFF planner

Defining Actions

$$
\begin{aligned}
&Action(Move(block, from, to)) : \\
&\quad PRECOND : On(b, from) \wedge Clear(block) \wedge Clear(to) \wedge \\
&\qquad Block(block) \wedge Block(to) \wedge \\
&\qquad (block \neq from) \wedge (block \neq to) \wedge (from \neq to) \\
&\quad EFFECT : On(block, to) \wedge \neg On(block, from) \wedge \\
&\qquad Clear(from) \wedge \neg Clear(to)
\end{aligned}
$$

Defining the Initial State



Initial

$On(A, Table) \wedge On(B, Table) \wedge On(C, Table) \wedge Block(A) \wedge Block(B) \wedge Block(C) \wedge Clear(A) \wedge Clear(B) \wedge Clear(C)$

Goal

$On(A, B) \wedge On(B, C)$

Implementing for FF planner

$$
\begin{aligned}
&Action(Move(b, x, y)) : \\
&\quad PRECOND : On(b, x) \wedge \\
&\qquad Clear(b) \wedge Clear(y) \wedge \\
&\qquad Block(b) \wedge Block(y) \wedge \\
&\qquad (b \neq x) \wedge (b \neq y) \wedge (x \neq y) \\
&\quad EFFECT : On(b, y) \wedge Clear(x) \wedge \\
&\qquad \neg On(b, x) \wedge \neg Clear(y)
\end{aligned}
$$

```
(:action MOVE
:parameters (
  ?b - block
  ?x - object
  ?y - block
)
:precondition (and
  (On ?b ?x)
  (Clear ?b)
  (Clear ?y)
  (not (= ?b ?x))
  (not (= ?b ?y))
  (not (= ?x ?y))
)
:effect (and
  (On ?b ?y)
  (Clear ?x)
  (not (On ?b ?x))
  (not (Clear ?y))
))
```

Backwards Search

- $g_1 = On(A, B) \wedge On(B, C)$

Backwards Search

- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$

Backwards Search

- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$
- Choose : $Move(A, x, B)$

Backwards Search

- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$
- Choose : $Move(A, x, B)$
- $g_2 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, C)$

Backwards Search

- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$
- Choose : $Move(A, x, B)$
- $g_2 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, C)$
- Available actions :
  $Move(B, x', C), Move(x', B, y), Move(x', A, y), Move(A, x', x)$

Backwards Search

- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$
- Choose : $Move(A, x, B)$
- $g_2 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, C)$
- Available actions :
  $Move(B, x', C), Move(x', B, y), Move(x', A, y), Move(A, x', x)$
- Choose : $Move(B, x', C)$

Backwards Search

- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$
- Choose : $Move(A, x, B)$
- $g_2 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, C)$
- Available actions :
  $Move(B, x', C), Move(x', B, y), Move(x', A, y), Move(A, x', x)$
- Choose : $Move(B, x', C)$
- $g_3 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, x') \wedge Clear(B) \wedge$
  $Clear(C) \wedge Block(C) \wedge B \neq x' \wedge B \neq C \wedge x' \neq C$

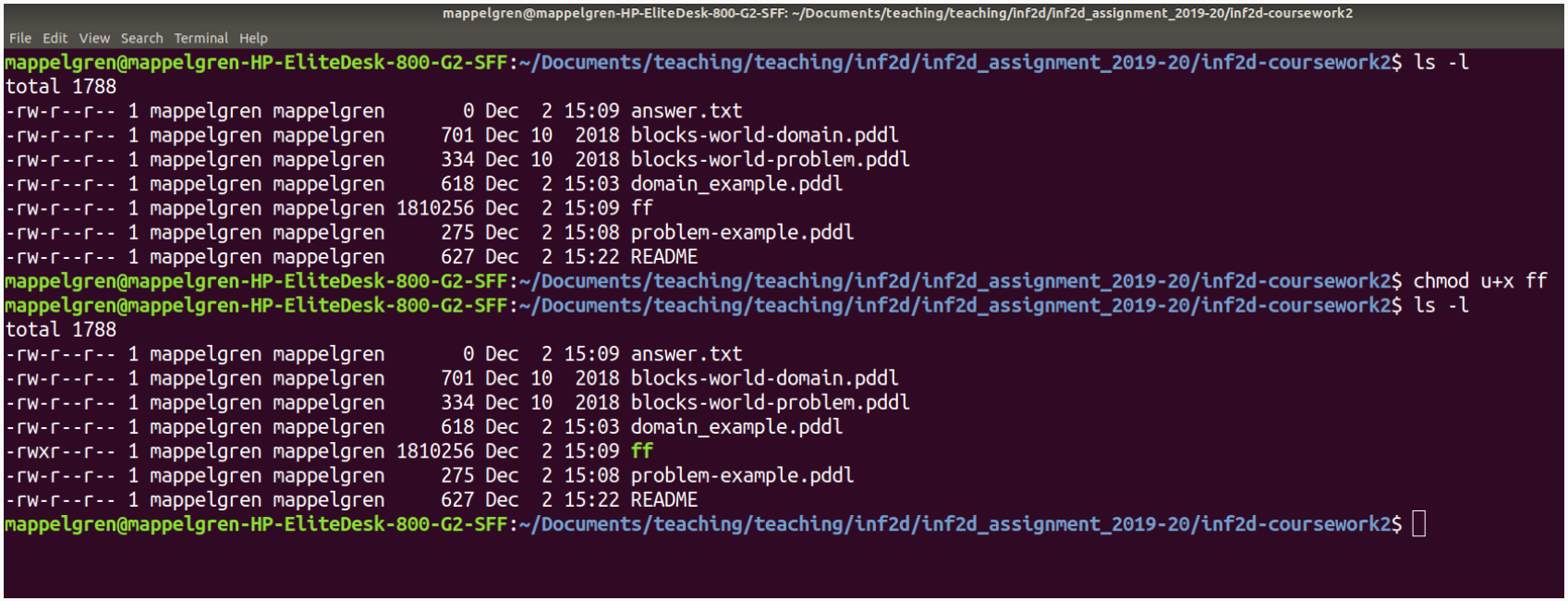
Backwards Search

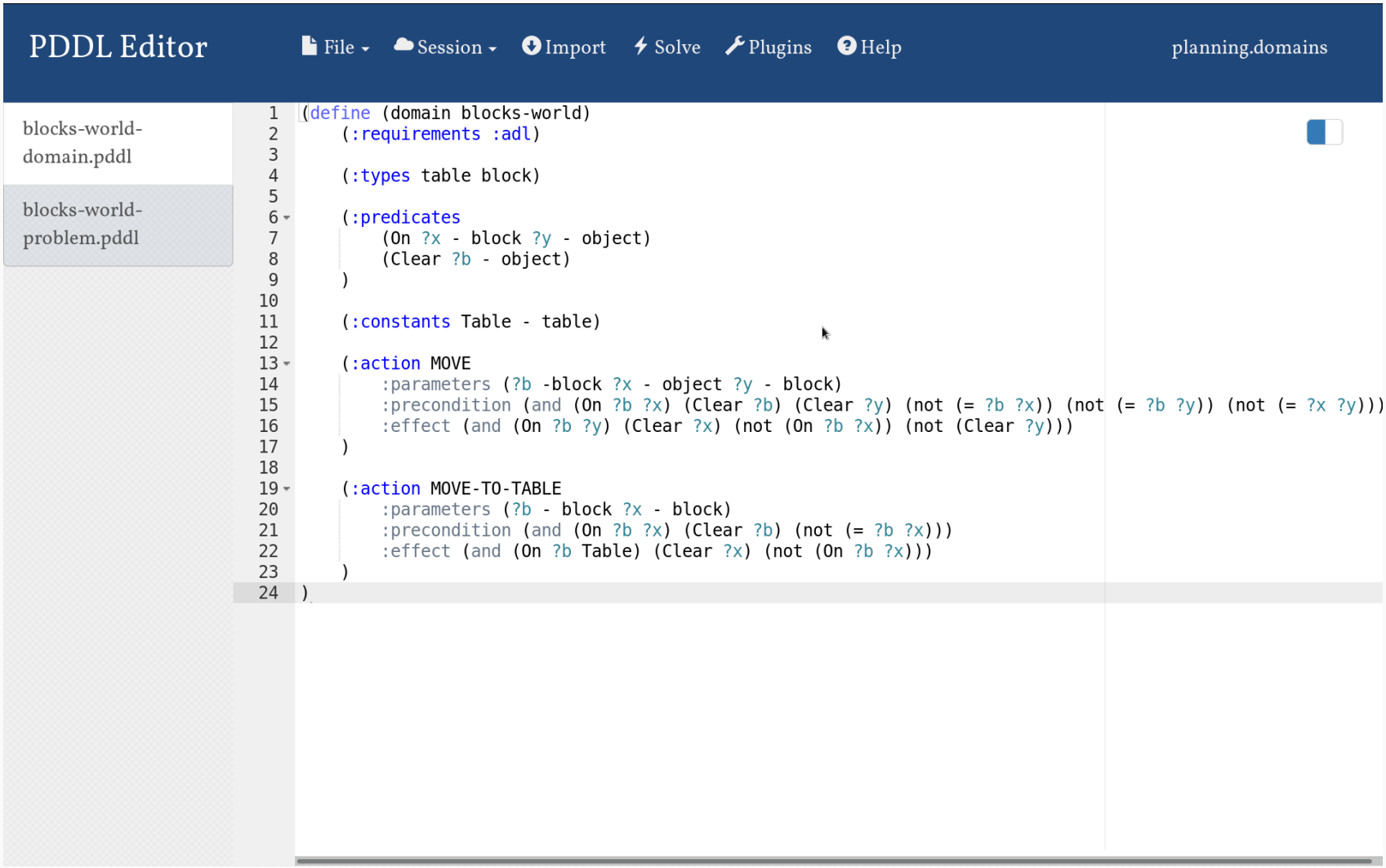
- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$
- Choose : $Move(A, x, B)$
- $g_2 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, C)$
- Available actions :
  $Move(B, x', C), Move(x', B, y), Move(x', A, y), Move(A, x', x)$
- Choose : $Move(B, x', C)$
- $g_3 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, x') \wedge Clear(B) \wedge$
  $Clear(C) \wedge Block(C) \wedge B \neq x' \wedge B \neq C \wedge x' \neq C$
- g_3 satisfies initial state by substituting
  $\{x = Table, x' = Table\}$

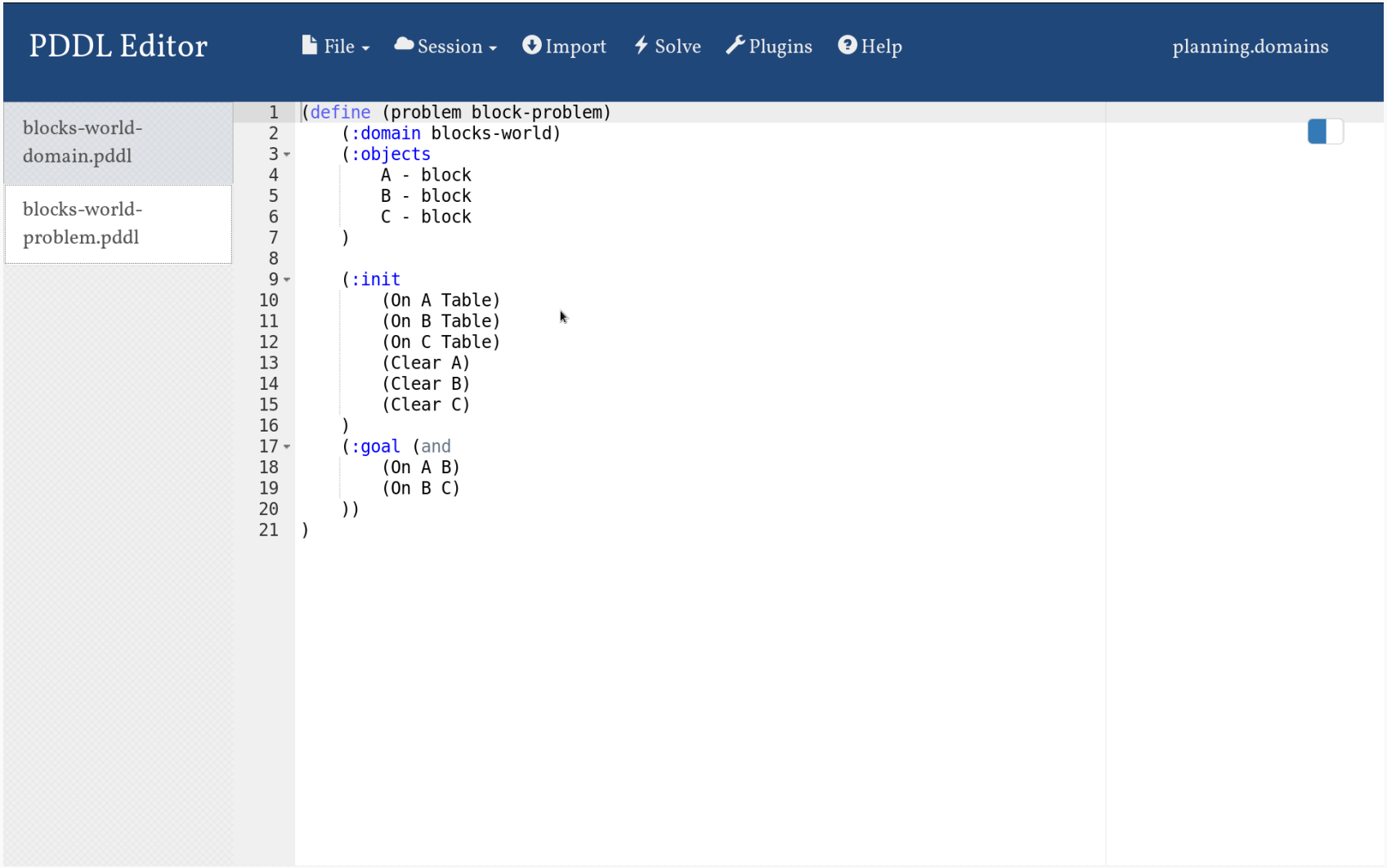
Backwards Search

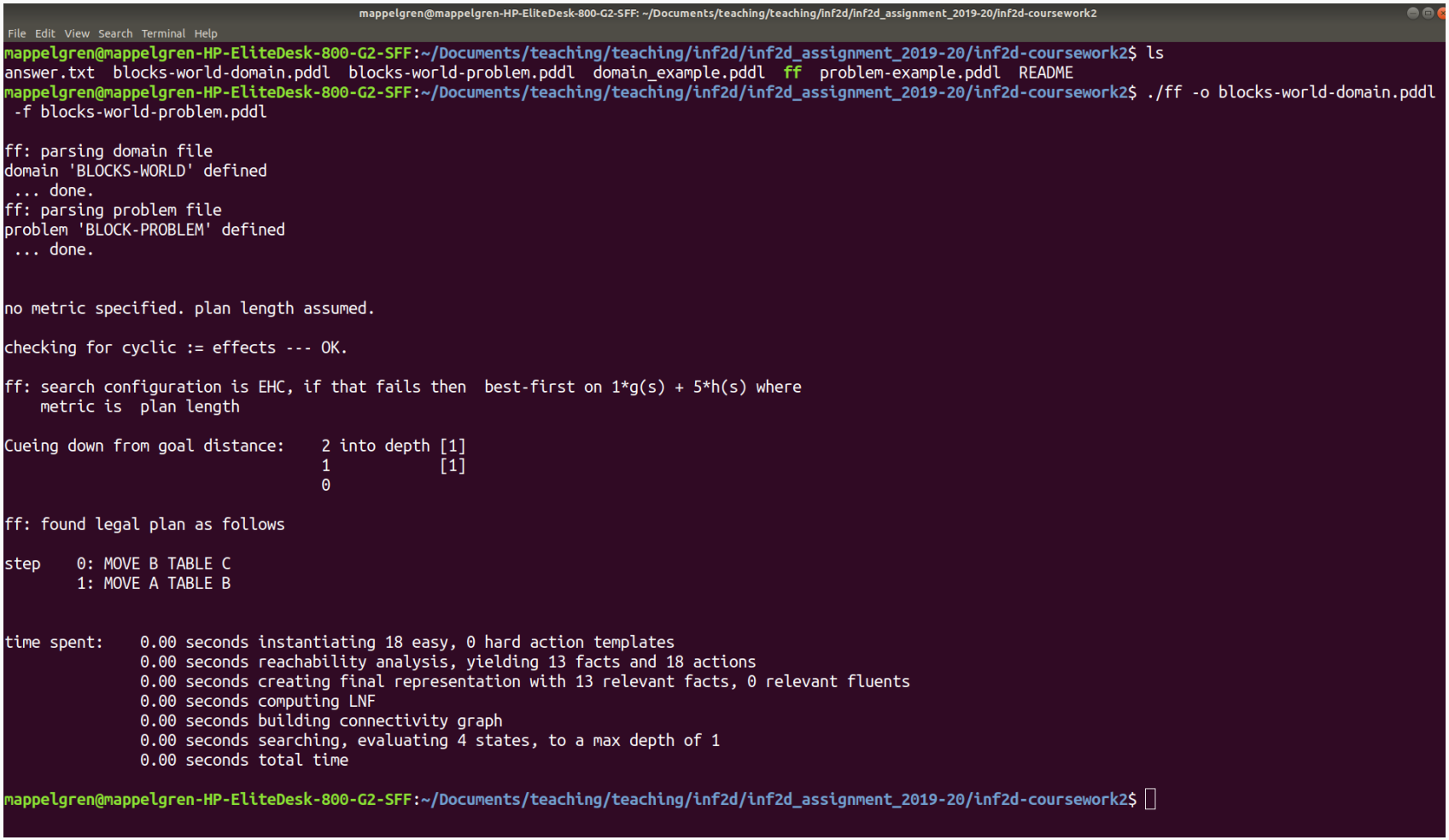
- $g_1 = On(A, B) \wedge On(B, C)$
- Available Actions: $Move(A, x, B), Move(B, x, C)$
- Choose : $Move(A, x, B)$
- $g_2 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, C)$
- Available actions :
  $Move(B, x', C), Move(x', B, y), Move(x', A, y), Move(A, x', x)$
- Choose : $Move(B, x', C)$
- $g_3 = On(A, x) \wedge Clear(A) \wedge Clear(B) \wedge Block(A) \wedge$
  $Block(B) \wedge A \neq x \wedge A \neq B \wedge x \neq B \wedge On(B, x') \wedge Clear(B) \wedge$
  $Clear(C) \wedge Block(C) \wedge B \neq x' \wedge B \neq C \wedge x' \neq C$
- g_3 satisfies initial state by substituting
  $\{x = Table, x' = Table\}$
- Done! Final Plan: $Move(B, Table, C), Move(A, Table, B)$

- MetricFF planner:
  https://fai.cs.uni-saarland.de/hoffmann/metric-ff.html
- PDDL wiki: https://planning.wiki/


mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF: ~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2
File Edit View Search Terminal Help
mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF:~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2$ ls -l
total 1788
-rw-r--r-- 1 mappelgren mappelgren 0 Dec 2 15:09 answer.txt
-rw-r--r-- 1 mappelgren mappelgren 701 Dec 10 2018 blocks-world-domain.pddl
-rw-r--r-- 1 mappelgren mappelgren 334 Dec 10 2018 blocks-world-problem.pddl
-rw-r--r-- 1 mappelgren mappelgren 618 Dec 2 15:03 domain_example.pddl
-rw-r--r-- 1 mappelgren mappelgren 1810256 Dec 2 15:09 ff
-rw-r--r-- 1 mappelgren mappelgren 275 Dec 2 15:08 problem-example.pddl
-rw-r--r-- 1 mappelgren mappelgren 627 Dec 2 15:22 README
mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF:~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2$ chmod u+x ff
mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF:~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2$ ls -l
total 1788
-rw-r--r-- 1 mappelgren mappelgren 0 Dec 2 15:09 answer.txt
-rw-r--r-- 1 mappelgren mappelgren 701 Dec 10 2018 blocks-world-domain.pddl
-rw-r--r-- 1 mappelgren mappelgren 334 Dec 10 2018 blocks-world-problem.pddl
-rw-r--r-- 1 mappelgren mappelgren 618 Dec 2 15:03 domain_example.pddl
-rwxr--r-- 1 mappelgren mappelgren 1810256 Dec 2 15:09 ff
-rw-r--r-- 1 mappelgren mappelgren 275 Dec 2 15:08 problem-example.pddl
-rw-r--r-- 1 mappelgren mappelgren 627 Dec 2 15:22 README
mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF:~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2$


PDDL Editor
File
Session
Import
Solve
Plugins
Help
planning.domains
blocks-world-domain.pddl
blocks-world-problem.pddl
(define (domain blocks-world)
(:requirements :adl)
(:types table block)
(:predicates
(On ?x - block ?y - object)
(Clear ?b - object)
)
(:constants Table - table)
(:action MOVE
:parameters (?b -block ?x - object ?y - block)
:precondition (and (On ?b ?x) (Clear ?b) (Clear ?y) (not (= ?b ?x)) (not (= ?b ?y)) (not (= ?x ?y)))
:effect (and (On ?b ?y) (Clear ?x) (not (On ?b ?x)) (not (Clear ?y)))
)
(:action MOVE-TO-TABLE
:parameters (?b - block ?x - block)
:precondition (and (On ?b ?x) (Clear ?b) (not (= ?b ?x)))
:effect (and (On ?b Table) (Clear ?x) (not (On ?b ?x)))
)
)


PDDL Editor
File
Session
Import
Solve
Plugins
Help
planning.domains
blocks-world-domain.pddl
blocks-world-problem.pddl
(define (problem block-problem)
(:domain blocks-world)
(:objects
A - block
B - block
C - block
)
(:init
(On A Table)
(On B Table)
(On C Table)
(Clear A)
(Clear B)
(Clear C)
)
(:goal (and
(On A B)
(On B C)
))
)


mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF: ~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2
File Edit View Search Terminal Help
mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF:~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2$ ls
answer.txt blocks-world-domain.pddl blocks-world-problem.pddl domain_example.pddl ff problem-example.pddl README
mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF:~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2$ ./ff -o blocks-world-domain.pddl
-f blocks-world-problem.pddl
ff: parsing domain file
domain 'BLOCKS-WORLD' defined
... done.
ff: parsing problem file
problem 'BLOCK-PROBLEM' defined
... done.
no metric specified. plan length assumed.
checking for cyclic := effects --- OK.
ff: search configuration is EHC, if that fails then best-first on 1*g(s) + 5*h(s) where
metric is plan length
Cueing down from goal distance: 2 into depth [1]
1 [1]
0
ff: found legal plan as follows
step 0: MOVE B TABLE C
1: MOVE A TABLE B
time spent: 0.00 seconds instantiating 18 easy, 0 hard action templates
0.00 seconds reachability analysis, yielding 13 facts and 18 actions
0.00 seconds creating final representation with 13 relevant facts, 0 relevant fluents
0.00 seconds computing LNF
0.00 seconds building connectivity graph
0.00 seconds searching, evaluating 4 states, to a max depth of 1
0.00 seconds total time
mappelgren@mappelgren-HP-EliteDesk-800-G2-SFF:~/Documents/teaching/teaching/inf2d/inf2d_assignment_2019-20/inf2d-coursework2$

Submission

```
Inf2d-ass2-s1202144/
  answers.txt
  domain-solution2.pddl
  problem-solution21.pddl
  problem-solution22.pddl
  problem-solution23.pddl
  domain-solution31.pddl
  problem-solution31.pddl
  ...
```

Compress

```
tar -cvzf Inf2d-ass2-s1202144.tar.gz Inf2d-ass2-s1202144
```

Check your archive file!

```
tar -tf Inf2d-ass2-s1202144.tar.gz
ls -l Inf2d-ass2-s1202144.tar.gz
```

Submit via LEARN

Questions?