Reasoning Systems
Day 2

Course Manager: GU Zhan (Sam)
zhan.gu@nus.edu.sg

Reasoning Systems

Day 2

Day 2 Agenda

2.1 Reasoning Using Informed Search

- 2.1.1 Machine Learning ≈ Search
- 2.1.2 Informed Search Techniques
- 2.1.3 Exercise

2.2 Search Based Reasoning Applications

- 2.2.1 Reasoning System Case Studies
- 2.2.2 Reasoning System Demos
- 2.2.3 Exercise

2.3 Search Reasoning [Workshop]

- 2.3.1 Informed Search
- 2.3.2 Reasoning System Exploration
- 2.3.3 Workshop Submission

2.1 Reasoning Using Informed Search

2.1.1 Machine Learning ≈ Search

2.1.2 Informed Search Techniques

2.1.3 Exercise

2.1 Reasoning Using Informed Search

2.1.1 Machine Learning ≈ Search

2.1.2 Informed Search Techniques

2.1.3 Exercise

Machine Learning ≈ Search

A *model* could be considered just as a tangible text file stored in computer/server, e.g. model.txt

model.txt - Notepad

File Edit Format View Help

```
Output  : Salary
Input 1 : Year-of-Education
Input 2 : Year-of-Working

Formula : Output = a x Input 1 + b x Input 2 + c
           (Salary)    (Year-of-Education) (Year-of-Working)

Formula parameters/coefficients       : a = 520
Formula parameters/coefficients       : b = 798
Formula parameters/coefficients       : c = -5007
```

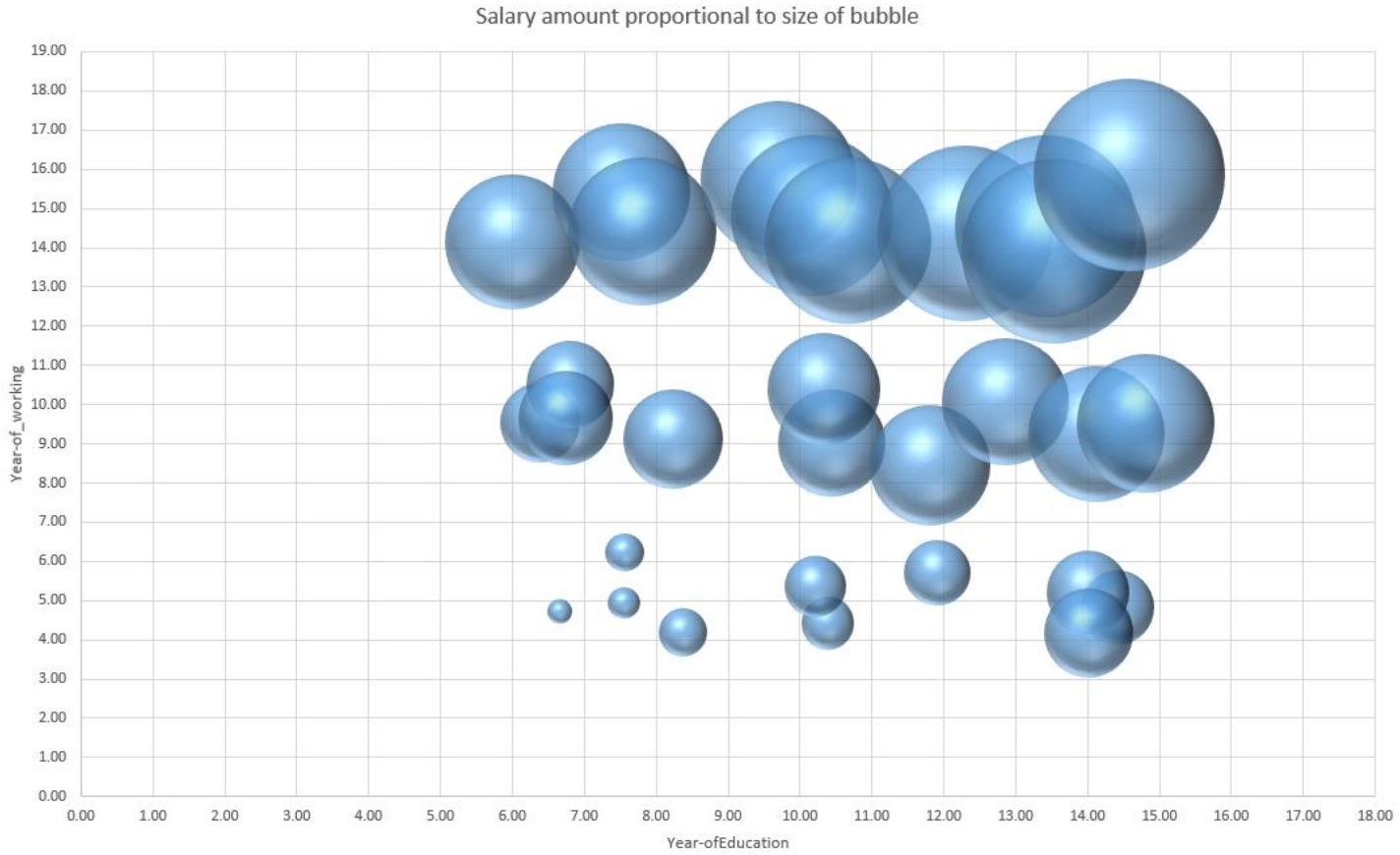
Salary.xlsx

Salary v02.xlsx - Excel

File Home Insert Page Layout Formulas Data Review View Team Enterprise Connect Tell me what you want to do... Sign in Share

AB85

sn	Year-of-Education	Year-of-Working	Salary
1	6.65	4.72	$1,817
2	7.55	4.94	$2,305
3	7.56	6.22	$2,832
4	8.37	4.19	$3,502
5	10.39	4.42	$3,890
6	10.22	5.35	$4,568
7	11.91	5.70	$4,884
8	14.41	4.82	$5,478
9	14.00	5.20	$6,181
10	14.01	4.18	$6,597
11	6.38	9.54	$5,860
12	6.81	10.52	$6,519
13	6.74	9.66	$7,018
14	8.23	9.12	$7,405
15	10.44	9.01	$7,945
16	10.33	10.40	$8,384
17	11.80	8.45	$8,954
18	12.85	10.06	$9,482
19	14.12	9.25	$10,102
20	14.81	9.52	$10,339
21	6.00	14.16	$10,067
22	7.51	15.43	$10,301
23	7.80	14.42	$11,056
24	9.69	15.76	$11,652
25	10.17	14.81	$12,078
26	10.66	14.17	$12,463
27	12.30	14.38	$13,144
28	13.42	14.55	$13,561
29	13.53	13.93	$13,851
30	14.57	15.86	$14,344



data regression

Ready 100%

AG83

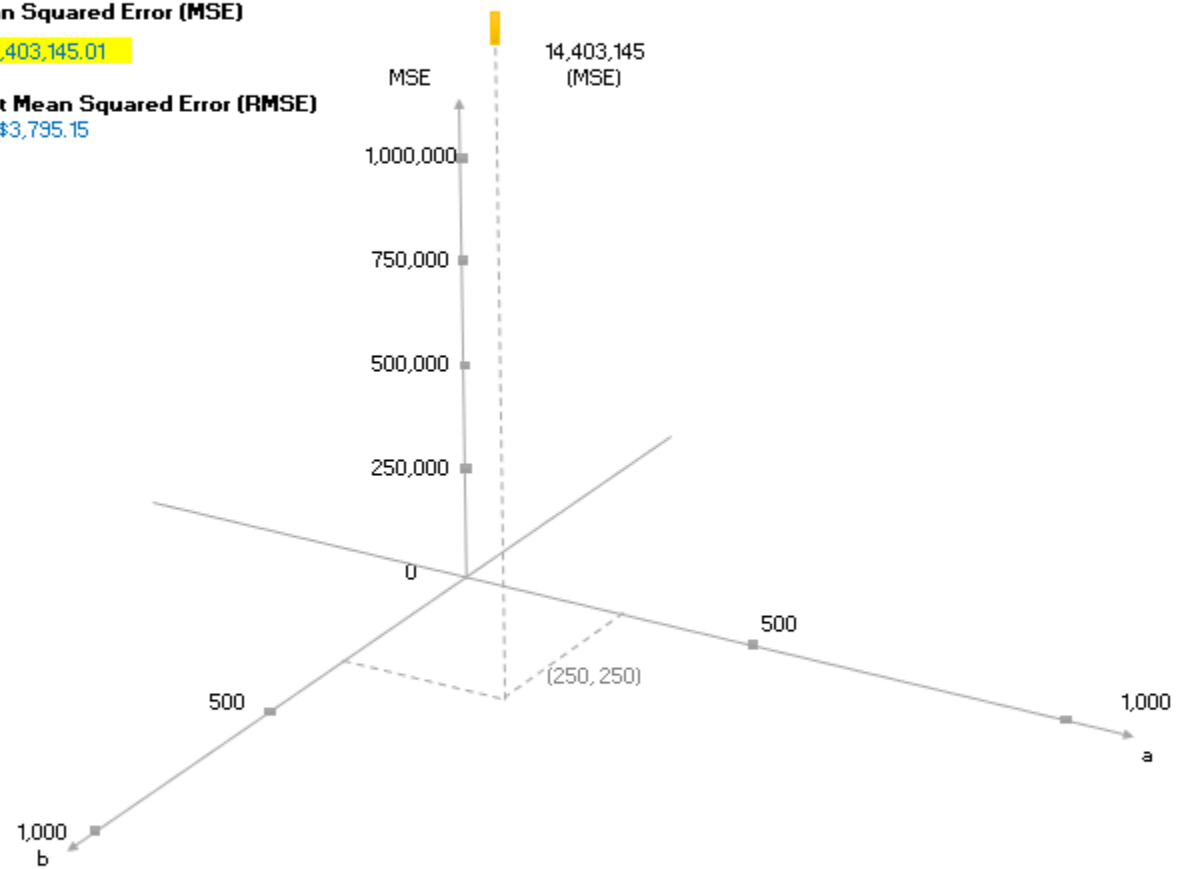
	Year-of-Education	Year-of-Working	Actual Salary	Predicted Salary	Error	Squared Error
1	6.65	4.72	$1,817	$3,094	$1,277	1631350.93
2	7.55	4.94	$2,305	$3,372	$1,067	1137955.29
3	7.56	6.22	$2,832	$3,696	$864	746840.55
4	8.37	4.19	$3,502	$3,390	-$112	12550.90
5	10.39	4.42	$3,890	$3,953	$63	4021.72
6	10.22	5.35	$4,568	$4,142	-$426	181762.96
7	11.91	5.70	$4,884	$4,653	-$231	53529.31
8	14.41	4.82	$5,478	$5,059	-$419	175312.16
9	14.00	5.20	$6,181	$5,052	-$1,129	1274054.04
10	14.01	4.18	$6,597	$4,798	-$1,799	3235867.79
11	6.38	9.54	$5,860	$4,231	-$1,629	2653418.26
12	6.81	10.52	$6,519	$4,581	-$1,938	3753984.50
13	6.74	9.66	$7,018	$4,348	-$2,670	7129131.61
14	8.23	9.12	$7,405	$4,587	-$2,818	7940908.24
15	10.44	9.01	$7,945	$5,113	-$2,832	8018831.95
16	10.33	10.40	$8,384	$5,432	-$2,952	8713505.11
17	11.80	8.45	$8,954	$5,312	-$3,642	13265681.99
18	12.85	10.06	$9,482	$5,978	-$3,504	12278000.66
19	14.12	9.25	$10,102	$6,093	-$4,009	16070103.38
20	14.81	9.52	$10,339	$6,331	-$4,008	16060688.83
21	6.00	14.16	$10,067	$5,289	-$4,778	22827331.14
22	7.51	15.43	$10,301	$5,985	-$4,316	18624972.22
23	7.80	14.42	$11,056	$5,805	-$5,251	27575400.32
24	9.69	15.76	$11,652	$6,613	-$5,039	25396319.86
25	10.17	14.81	$12,078	$6,495	-$5,583	31173713.56
26	10.66	14.17	$12,463	$6,459	-$6,004	36048232.00
27	12.30	14.38	$13,144	$6,921	-$6,223	38724723.03
28	13.42	14.55	$13,561	$7,244	-$6,317	39906045.47
29	13.53	13.93	$13,851	$7,113	-$6,738	45402118.75
30	14.57	15.86	$14,344	$7,857	-$6,487	42077993.84

Mean Squared Error (MSE)

14,403,145.01

Root Mean Squared Error (RMSE)

$3,795.15



Salary = a x Eduaction + b x Working + c

a =	250
b =	250
c =	250

data | regression | learning 1 | learning 2

AG83

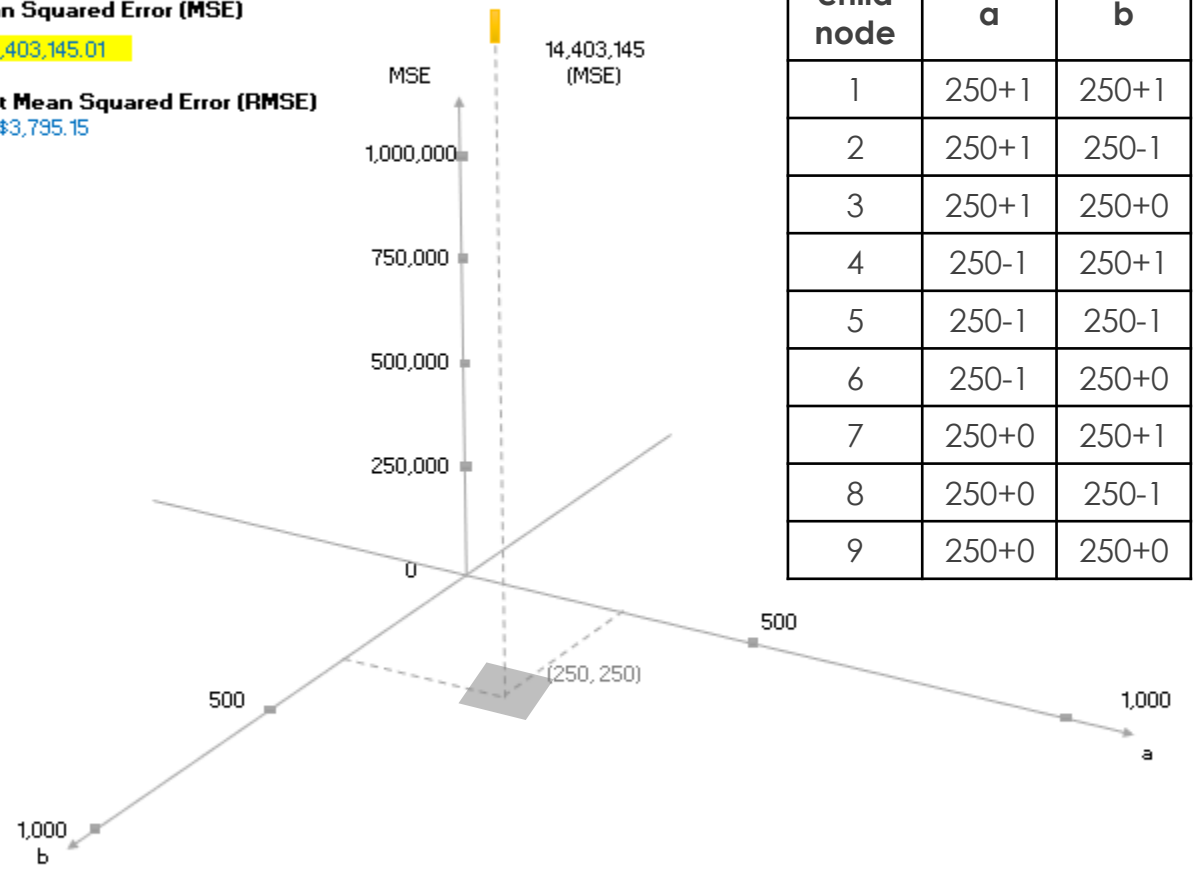
	Year-of-Education	Year-of-Working	Actual Salary	Predicted Salary	Error	Squared Error
1	6.65	4.72	$1,817	$3,094	$1,277	1631350.93
2	7.55	4.94	$2,305	$3,372	$1,067	1137955.29
3	7.56	6.22	$2,832	$3,696	$864	746840.55
4	8.37	4.19	$3,502	$3,390	-$112	12550.90
5	10.39	4.42	$3,890	$3,953	$63	4021.72
6	10.22	5.35	$4,568	$4,142	-$426	181762.96
7	11.91	5.70	$4,884	$4,653	-$231	53529.31
8	14.41	4.82	$5,478	$5,059	-$419	175312.16
9	14.00	5.20	$6,181	$5,052	-$1,129	1274054.04
10	14.01	4.18	$6,597	$4,798	-$1,799	3235867.79
11	6.38	9.54	$5,860	$4,231	-$1,629	2653418.26
12	6.81	10.52	$6,519	$4,581	-$1,938	3753984.50
13	6.74	9.66	$7,018	$4,348	-$2,670	7129131.61
14	8.23	9.12	$7,405	$4,587	-$2,818	7940908.24
15	10.44	9.01	$7,945	$5,113	-$2,832	8018831.95
16	10.33	10.40	$8,384	$5,432	-$2,952	8713505.11
17	11.80	8.45	$8,954	$5,312	-$3,642	13265681.99
18	12.85	10.06	$9,482	$5,978	-$3,504	12278000.66
19	14.12	9.25	$10,102	$6,093	-$4,009	16070103.38
20	14.81	9.52	$10,339	$6,331	-$4,008	16060688.83
21	6.00	14.16	$10,067	$5,289	-$4,778	22827331.14
22	7.51	15.43	$10,301	$5,985	-$4,316	18624972.22
23	7.80	14.42	$11,056	$5,805	-$5,251	27575400.32
24	9.69	15.76	$11,652	$6,613	-$5,039	25396319.86
25	10.17	14.81	$12,078	$6,495	-$5,583	31173713.56
26	10.66	14.17	$12,463	$6,459	-$6,004	36048232.00
27	12.30	14.38	$13,144	$6,921	-$6,223	38724723.03
28	13.42	14.55	$13,561	$7,244	-$6,317	39906045.47
29	13.53	13.93	$13,851	$7,113	-$6,738	45402118.75
30	14.57	15.86	$14,344	$7,857	-$6,487	42077993.84

Mean Squared Error (MSE)

14,403,145.01

Root Mean Squared Error (RMSE)

$3,795.15



child node	a	b
1	250+1	250+1
2	250+1	250-1
3	250+1	250+0
4	250-1	250+1
5	250-1	250-1
6	250-1	250+0
7	250+0	250+1
8	250+0	250-1
9	250+0	250+0

Salary = a x Eduaction + b x Working + c

a =	250
b =	250
c =	250

data | regression | learning 1 | learning 2

AG83

	Year-of-Education	Year-of-Working	Actual Salary	Predicted Salary	Error	Squared Error
1	6.65	4.72	$1,817	$3,094	$1,277	1631350.93
2	7.55	4.94	$2,305	$3,372	$1,067	1137955.29
3	7.56	6.22	$2,832	$3,696	$864	746840.55
4	8.37	4.19	$3,502	$3,390	-$112	12550.90
5	10.39	4.42	$3,890	$3,953	$63	4021.72
6	10.22	5.35	$4,568	$4,142	-$426	181762.96
7	11.91	5.70	$4,884	$4,653	-$231	53529.31
8	14.41	4.82	$5,478	$5,059	-$419	175312.16
9	14.00	5.20	$6,181	$5,052	-$1,129	1274054.04
10	14.01	4.18	$6,597	$4,798	-$1,799	3235867.79
11	6.38	9.54	$5,860	$4,231	-$1,629	2653418.26
12	6.81	10.52	$6,519	$4,581	-$1,938	3753984.50
13	6.74	9.66	$7,018	$4,348	-$2,670	7129131.61
14	8.23	9.12	$7,405	$4,587	-$2,818	7940908.24
15	10.44	9.01	$7,945	$5,113	-$2,832	8018831.95
16	10.33	10.40	$8,384	$5,432	-$2,952	8713505.11
17	11.80	8.45	$8,954	$5,312	-$3,642	13265681.99
18	12.85	10.06	$9,482	$5,978	-$3,504	12278000.66
19	14.12	9.25	$10,102	$6,093	-$4,009	16070103.38
20	14.81	9.52	$10,339	$6,331	-$4,008	16060688.83
21	6.00	14.16	$10,067	$5,289	-$4,778	22827331.14
22	7.51	15.43	$10,301	$5,985	-$4,316	18624972.22
23	7.80	14.42	$11,056	$5,805	-$5,251	27575400.32
24	9.69	15.76	$11,652	$6,613	-$5,039	25396319.86
25	10.17	14.81	$12,078	$6,495	-$5,583	31173713.56
26	10.66	14.17	$12,463	$6,459	-$6,004	36048232.00
27	12.30	14.38	$13,144	$6,921	-$6,223	38724723.03
28	13.42	14.55	$13,561	$7,244	-$6,317	39906045.47
29	13.53	13.93	$13,851	$7,113	-$6,738	45402118.75
30	14.57	15.86	$14,344	$7,857	-$6,487	42077993.84

Mean Squared Error (MSE)

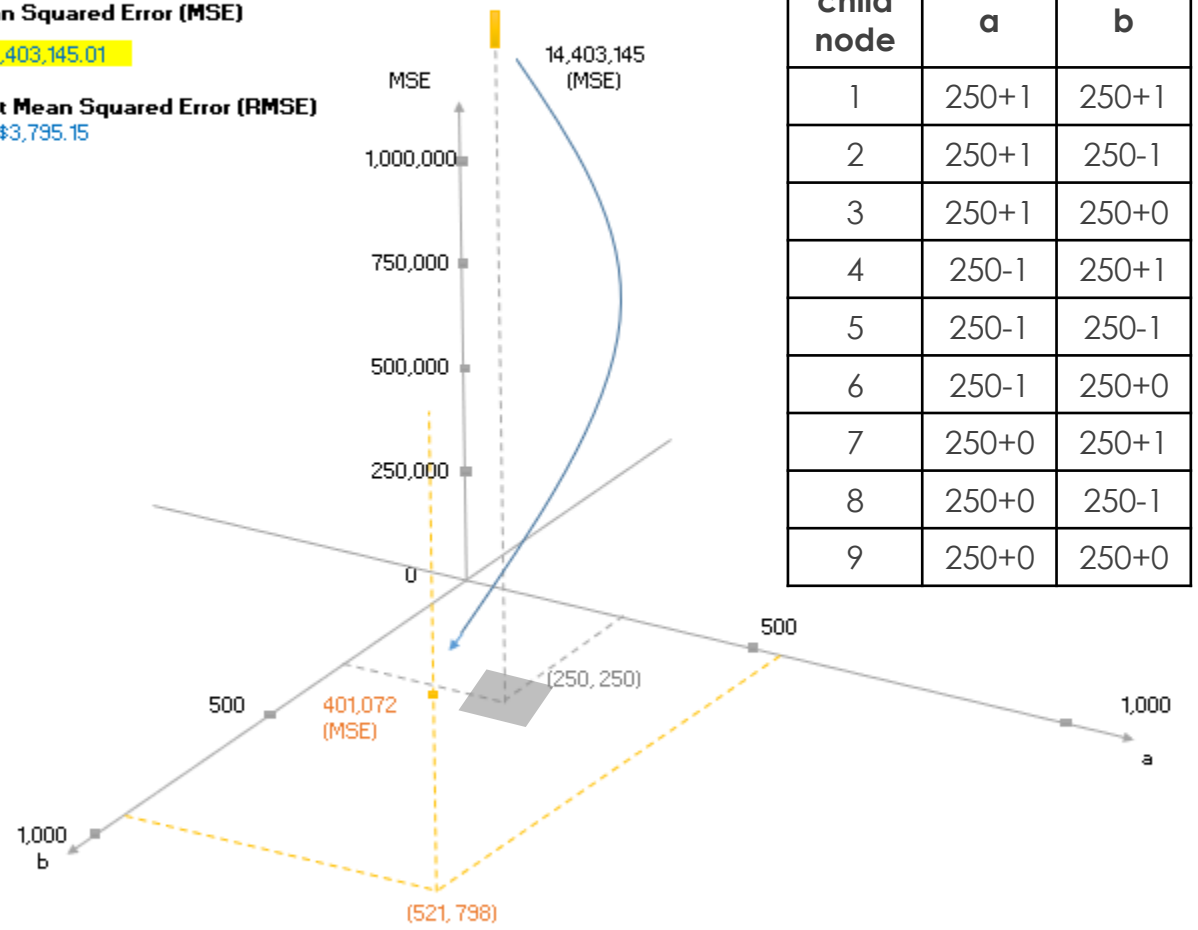
14,403,145.01

Root Mean Squared Error (RMSE)

$3,795.15

Salary = a x Eduaction + b x Working + c

a =	250
b =	250
c =	250



child node	a	b
1	250+1	250+1
2	250+1	250-1
3	250+1	250+0
4	250-1	250+1
5	250-1	250-1
6	250-1	250+0
7	250+0	250+1
8	250+0	250-1
9	250+0	250+0

Salary v02.xlsx - Excel

File Home Insert Page Layout Formulas Data Review View Team Enterprise Connect Tell me what you want to do... Sign in Share

AZ92

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.984375561
R Square	0.968995246
Adjusted R Square	0.966698597
Standard Error	667.5580743
Observations	30

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	376041022.2	188020511.1	421.9170953	4.30806E-21
Residual	27	12032112.13	445633.7825		
Total	29	388073134.3			

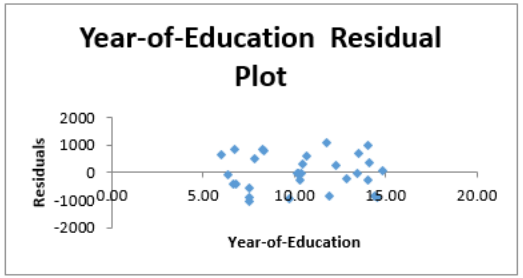
	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	-5007.776515	551.1804093	-9.085548816	1.06587E-09	-6138.705299	-3876.847731	-6138.705299	-3876.847731
Year-of-Education	520.7382643	42.89431991	12.14002846	1.90221E-12	432.7263898	608.7501389	432.7263898	608.7501389
Year-of-Working	798.286873	30.15824157	26.46994093	7.53819E-21	736.4072726	860.1664733	736.4072726	860.1664733

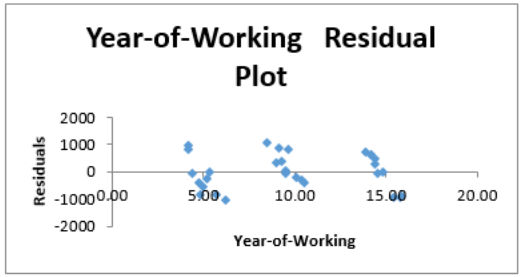
RESIDUAL OUTPUT

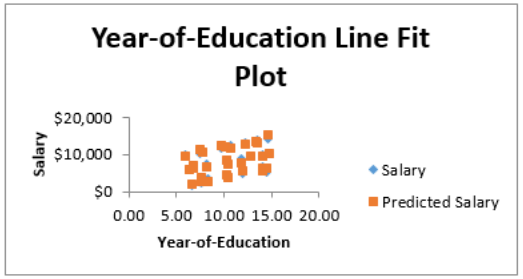
Observation	*Predicted Salary*	*Residuals*	*Standard Residuals*
1	2227.378049	-410.3780489	-0.63710672
2	2865.1592	-560.1591999	-0.869640059
3	3897.843043	-1065.843043	-1.654707816
4	2696.024781	805.9752188	1.251266313
5	3934.190757	-44.19075715	-0.06860559
6	4582.710338	-14.71033805	-0.022837613
7	5745.008025	-861.0080251	-1.336704047
8	6348.358938	-870.3589379	-1.351221221
9	6439.491245	-258.4912447	-0.401304382
10	5626.422479	970.5775209	1.506809301
11	5932.578944	-72.57894434	-0.112677891
12	6933.034306	-414.0343056	-0.642783012
13	6208.28567	809.7143298	1.257071236
14	6555.942708	849.057292	1.318150686
15	7624.171392	320.8286079	0.498082348

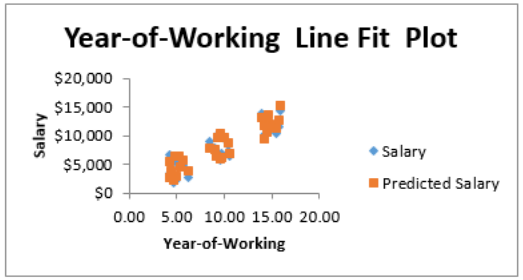
PROBABILITY OUTPUT

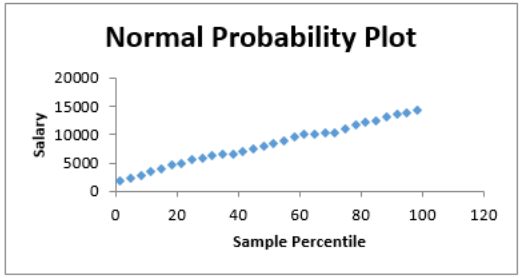
Percentile	*Salary*
1.666666667	1817
5	2305
8.333333333	2832
11.66666667	3502
15	3890
18.33333333	4568
21.66666667	4884
25	5478
28.33333333	5860
31.66666667	6181
35	6519
38.33333333	6597
41.66666667	7018
45	7405
48.33333333	7945











data regression

Ready 100%

Machine Learning ≈ Search

Machine learning procedure:

- Learn the best knowledge (***a function + a set of coefficients***), which can ***accurately (minimum error)*** predict **Salary** based on **Education** & **Working**.

Search / Optimization procedure :

- Use ‘generate and test’ method to obtain the knowledge/solution (***a function + the best set of coefficients among all possible coefficients space***), which can minimize the outcome of an ***evaluation function***, e.g. the difference(***error***) between predicted value and target value, so as to more ***accurately*** predict **Salary** based on **Education** & **Working**.

Machine Learning ≈ Search

Same *evaluation function*; different names:

- **Supervised Machine Learning:**

  Loss function, Accuracy, F1, Precision, Recall, MSE, RMSE, MAE, etc.

- **Unsupervised Machine Learning:**

  Silhouette coefficient, Dunn's index, etc.

- **Reinforcement learning:**

  Reward function, Q function, etc.

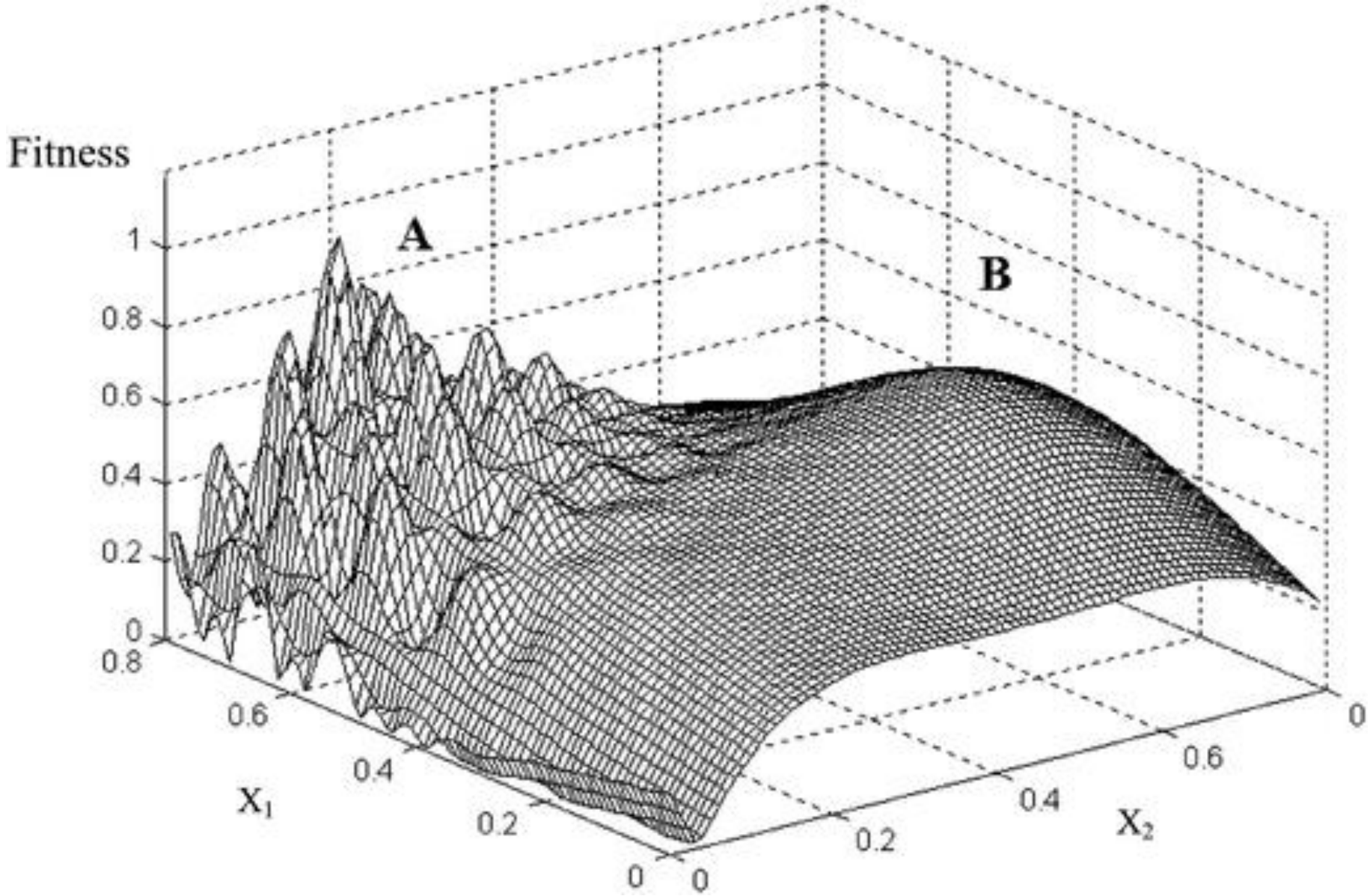
- **Search:**

  Evaluation function, Heuristic function, Goal test function, etc.

- **Optimization:**

  Objective function, Utility function, Expectation maximization, etc.

Machine Learning ≈ Search



Area A:
Non-convex search/parameter space

Area B:
Convex search/parameter space

Source: https://pubs.rsc.org/en/content/articlelanding/2005/an/b307211h#!divAbstract

2.1 Reasoning Using Informed Search

2.1.1 Machine Learning ≈ Search

2.1.2 Informed Search Techniques

2.1.3 Exercise

Informed Search Techniques

- **Use Heuristics (domain specific knowledge)**
- **Hill Climbing Search (HC)**
- **A Star Search (A*)**

Informed Search Techniques

Use Heuristics

- **Basic Idea**
  - It works by firstly sorting the list of nodes, then explore them orderly, according to their **optimality** (best score) determined by an evaluation function ***f*(*n*)**
- **Typical Best-first Strategies**
  - Use heuristics only : Hill Climbing, Tabu search
  - Use heuristics and past cost : A*, Late Acceptance
  - Use heuristics and randomness : Genetic algorithms

Informed Search Techniques

Use Heuristics

- **A key component of evaluation function *f(n)* is a heuristic function: *h(n)* = estimated cost of the cheapest path from node n to a goal node. Or estimated degree of difference between the current states/solutions and ultimate goal state.**
- **By convention, the lower the heuristic value the more promising the node: better to check first.** h(n) = 0 when n is goal
- ☺ **When there is no other information available but only the heuristics, *f(n)* = *h(n)***

Informed Search Techniques

Use Heuristics

- **Heuristic Function (Knowledge)**
  - ***h(n)*** = Correct or Incorrect positions

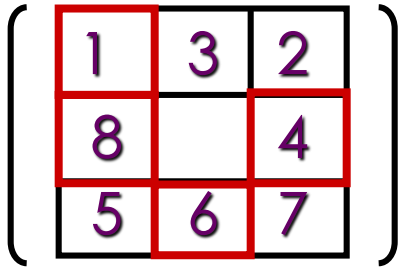
Initial=

1	3	2
8		4
5	6	7

Goal=

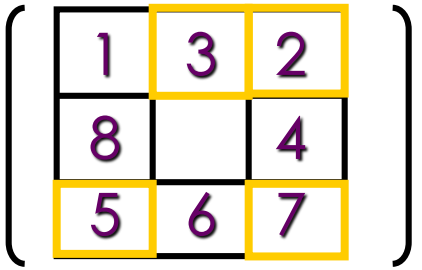
1	2	3
8		4
7	6	5

  - $h_1(n)$ = number of tiles in their **correct** goal state positions



= 4

  - $h_2(n)$ = number of tiles in their **incorrect** goal state positions

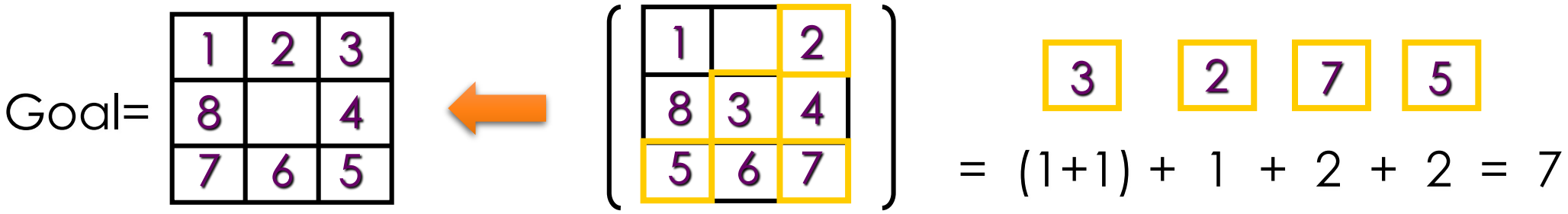


= 4

Informed Search Techniques

Use Heuristics

- **Heuristic Function (Knowledge)**
  - ***h(n)*** = Manhattan distance (sum of the horizontal & vertical distance each tile is away from its goal state position)

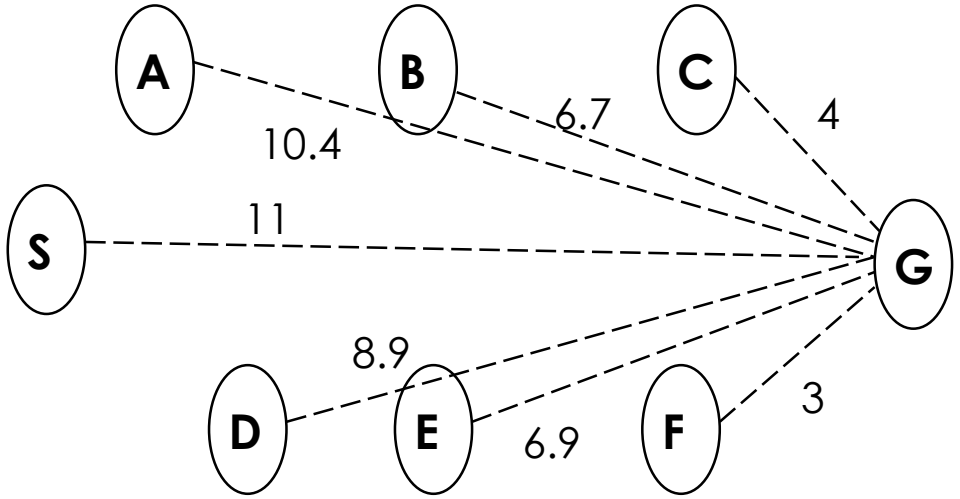


☺ **Manhattan distance gives a better estimate of the distance to the Goal state**

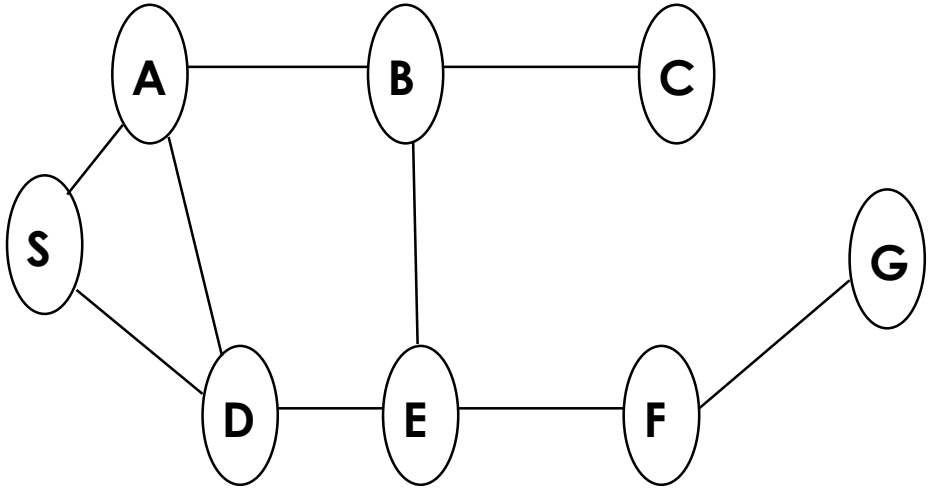
Informed Search Techniques

Use Heuristics

- **Heuristic Function (Knowledge)**
  - ***h(n)*** = "straight-line" distance between each city & the goal (This is useful estimation or heuristic.)



Heuristic Distances



Actual Roads

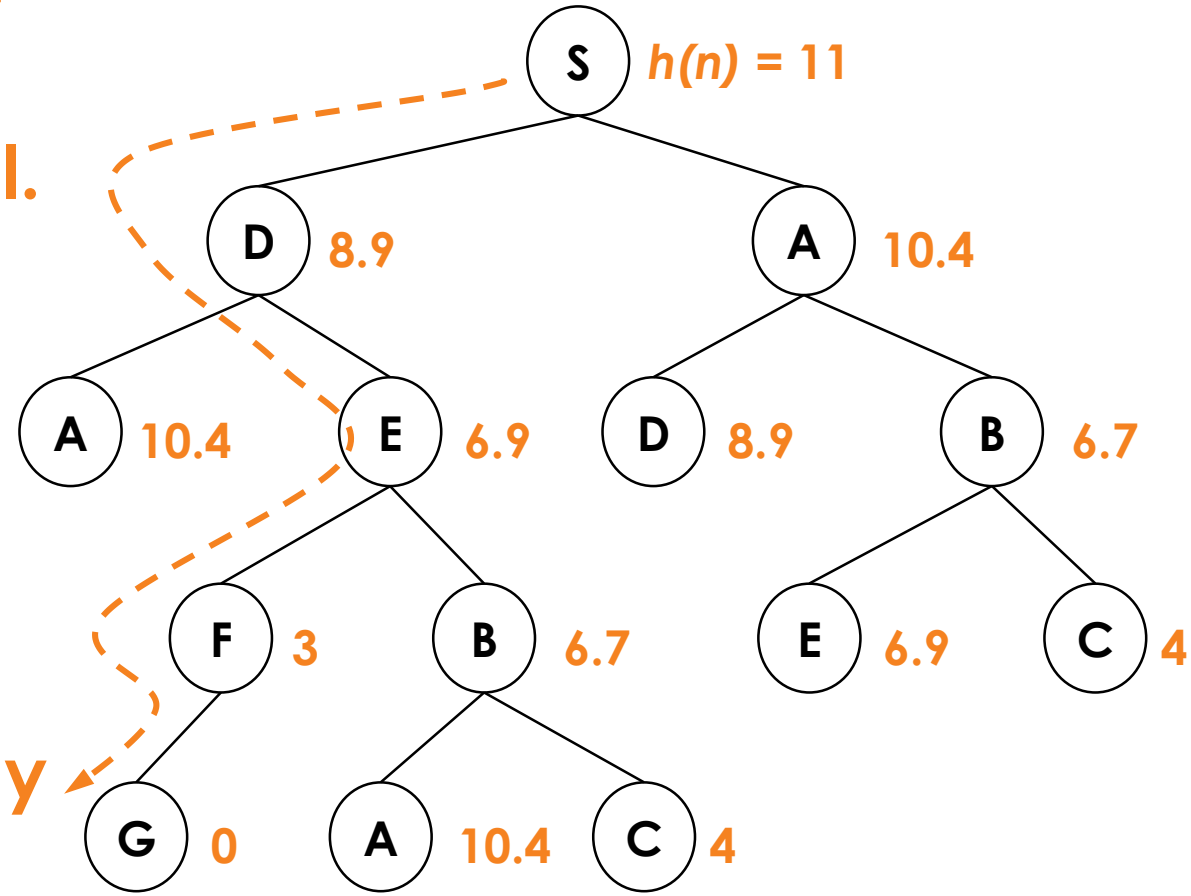
Informed Search Techniques

Hill Climbing (Greedy Best First Search)

- Minimize the cost to reach the goal state by expanding the node that is closest to the goal.
- Using only the heuristic values for evaluation function:

$$f(n) = h(n)$$

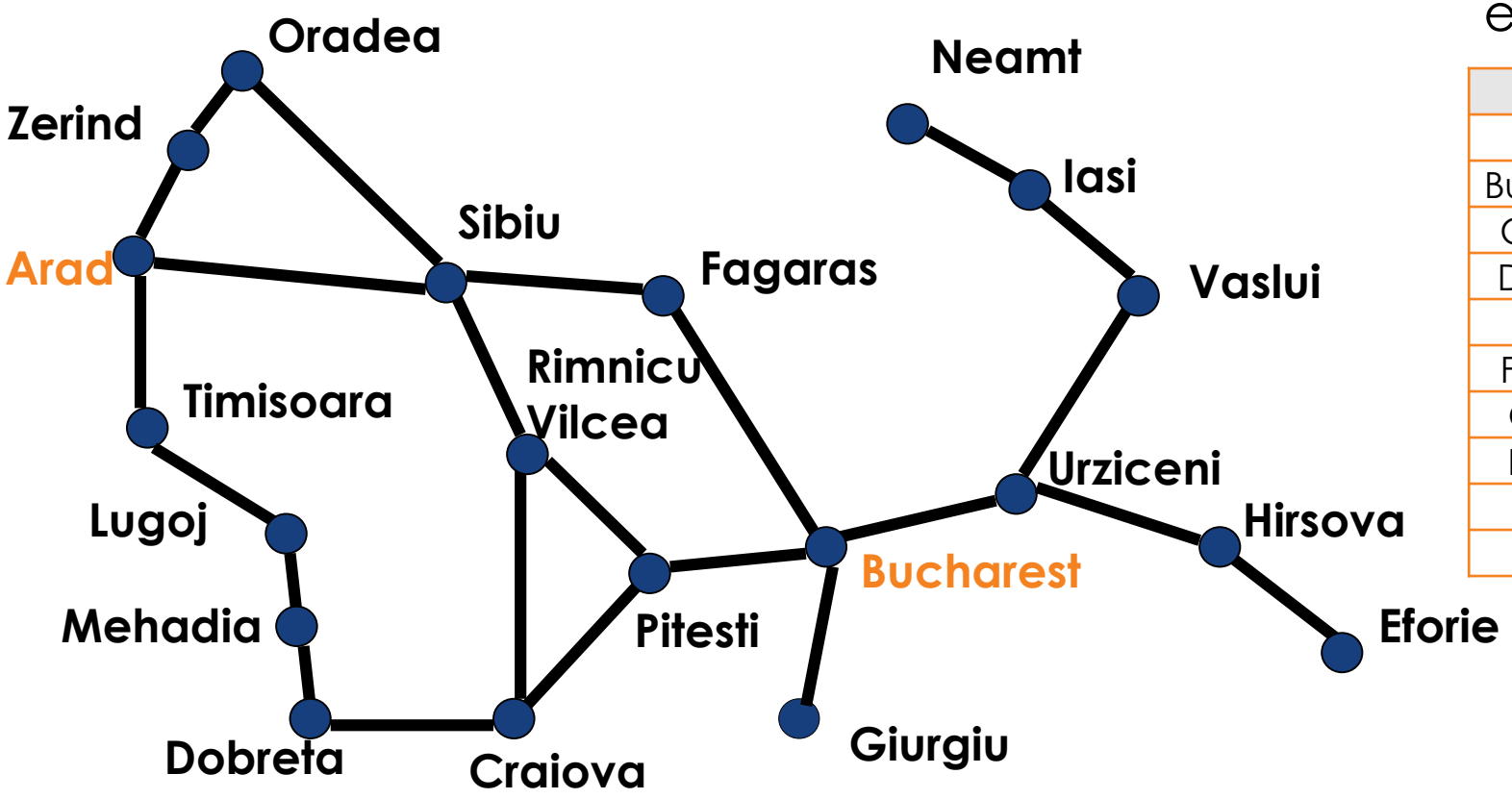
- Select search node with min(*f(n)*) at each step.
- Follow a single path all the way to a goal, but can back track when it hits a dead end.



Informed Search Techniques

Hill Climbing (Greedy Best First Search)

- **Arad → Bucharest**

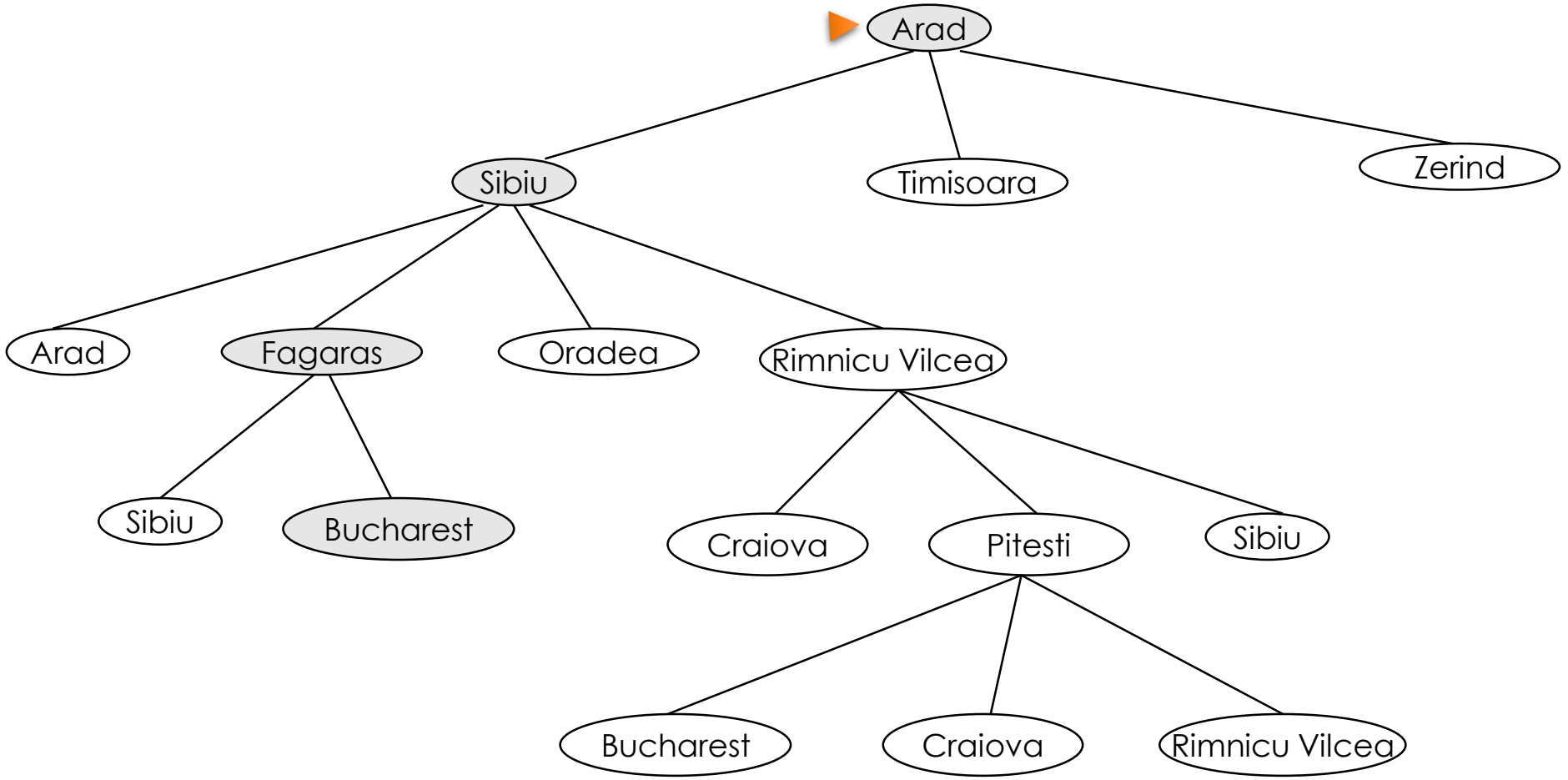


h(n) =Straight-line distance from each city to **Bucharest**

City	Distance	City	Distance
Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Dobreta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

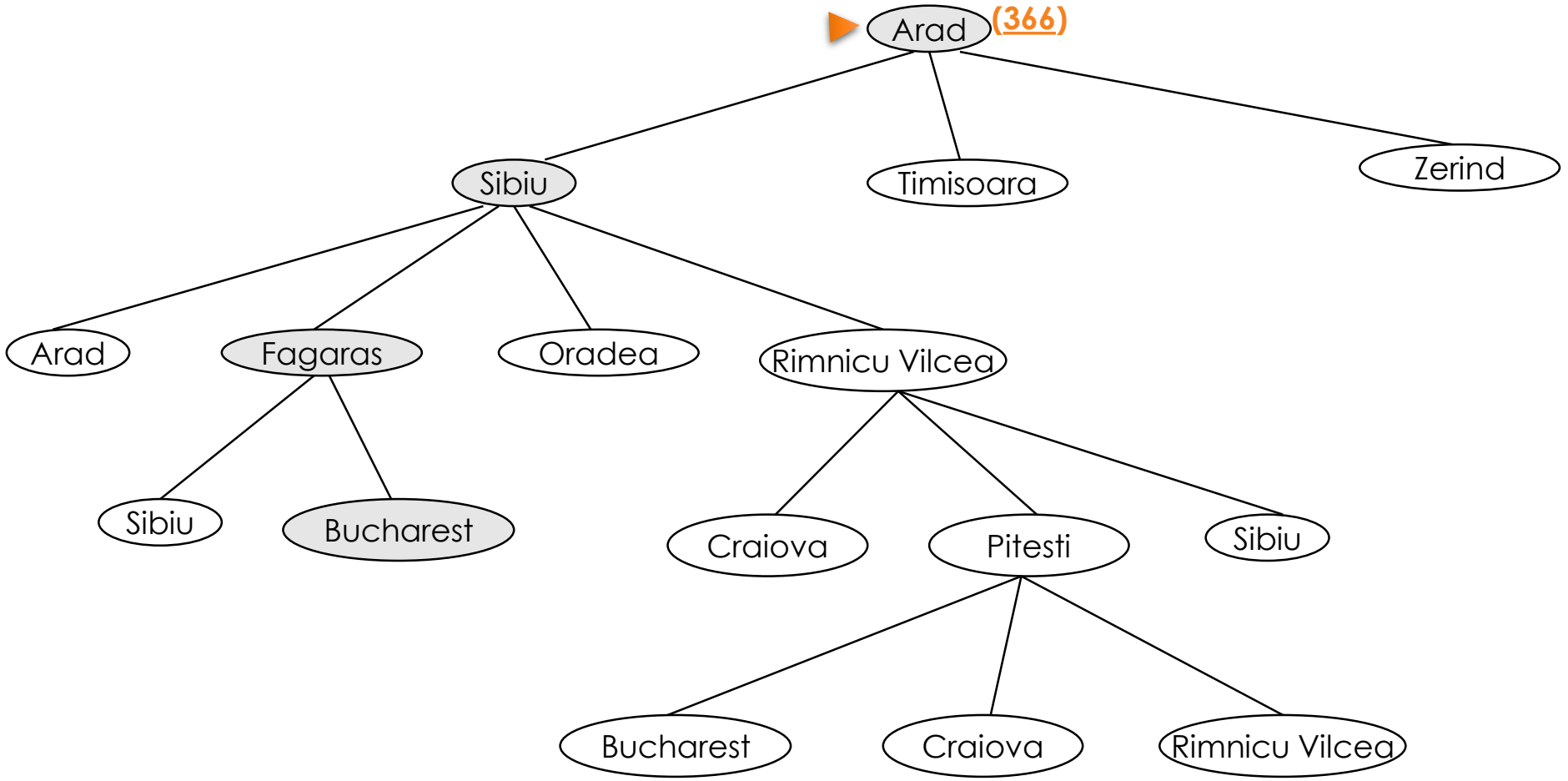
Informed Search Techniques

Hill Climbing (Greedy Best First Search)



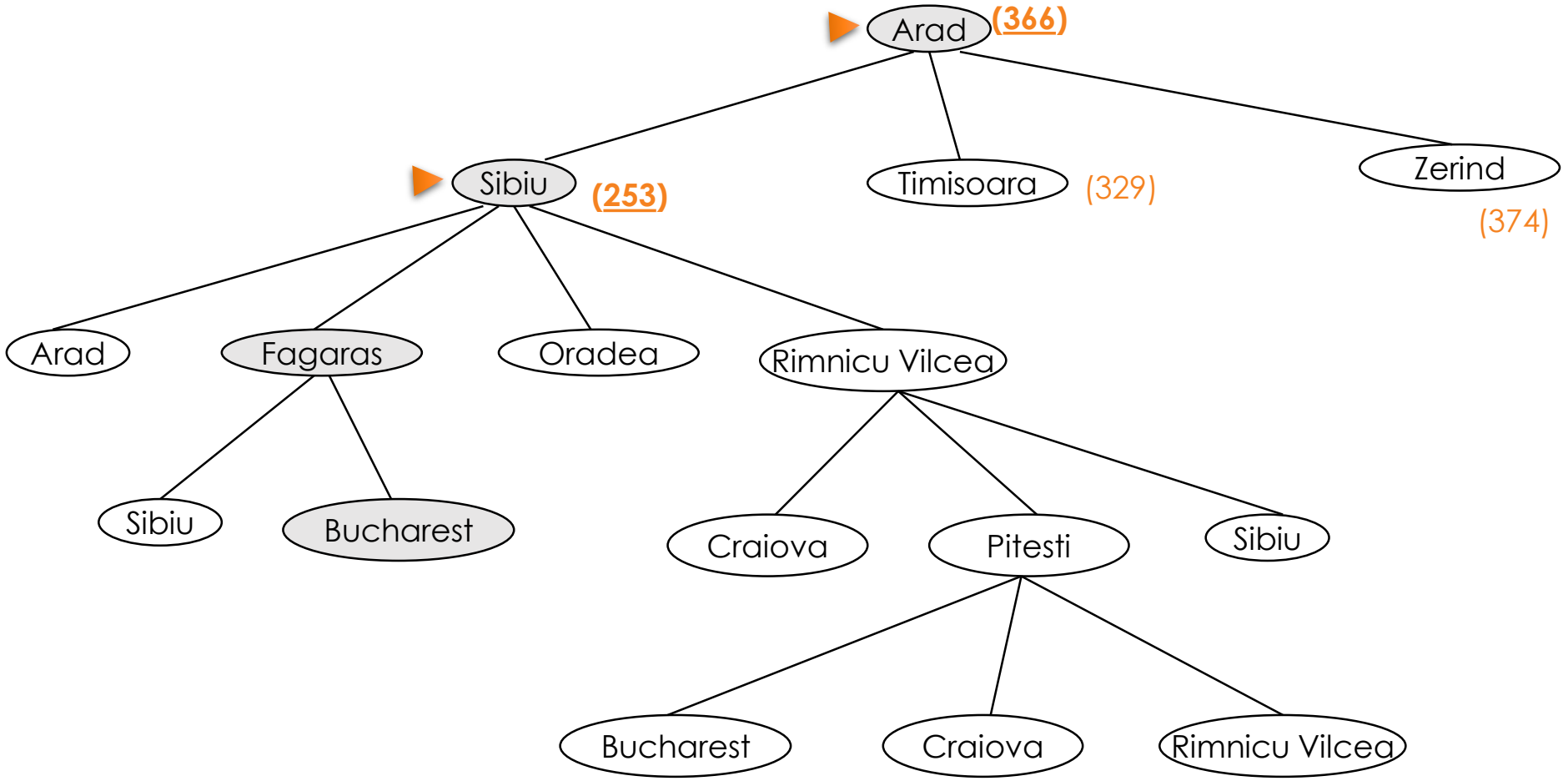
Informed Search Techniques

Hill Climbing (Greedy Best First Search)



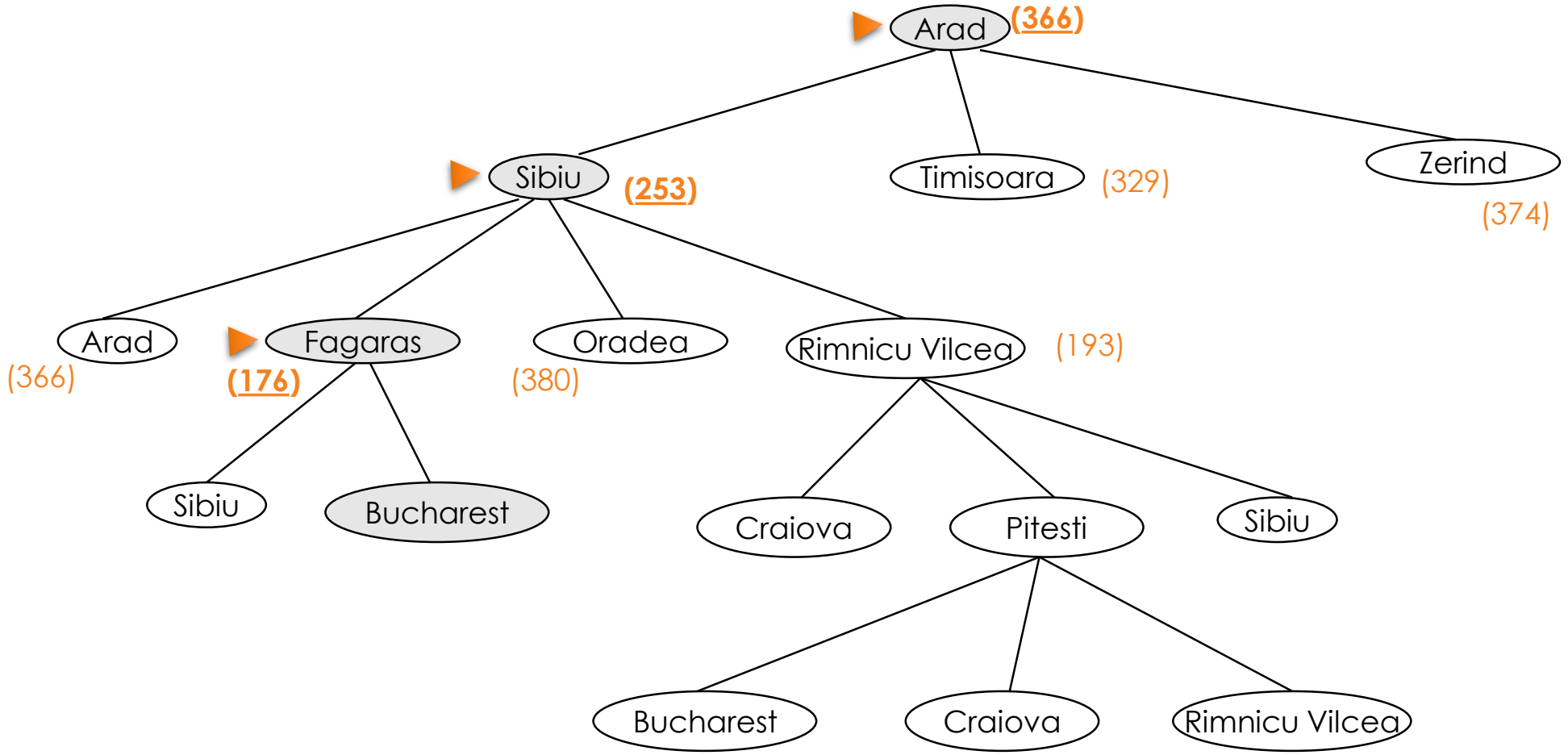
Informed Search Techniques

Hill Climbing (Greedy Best First Search)



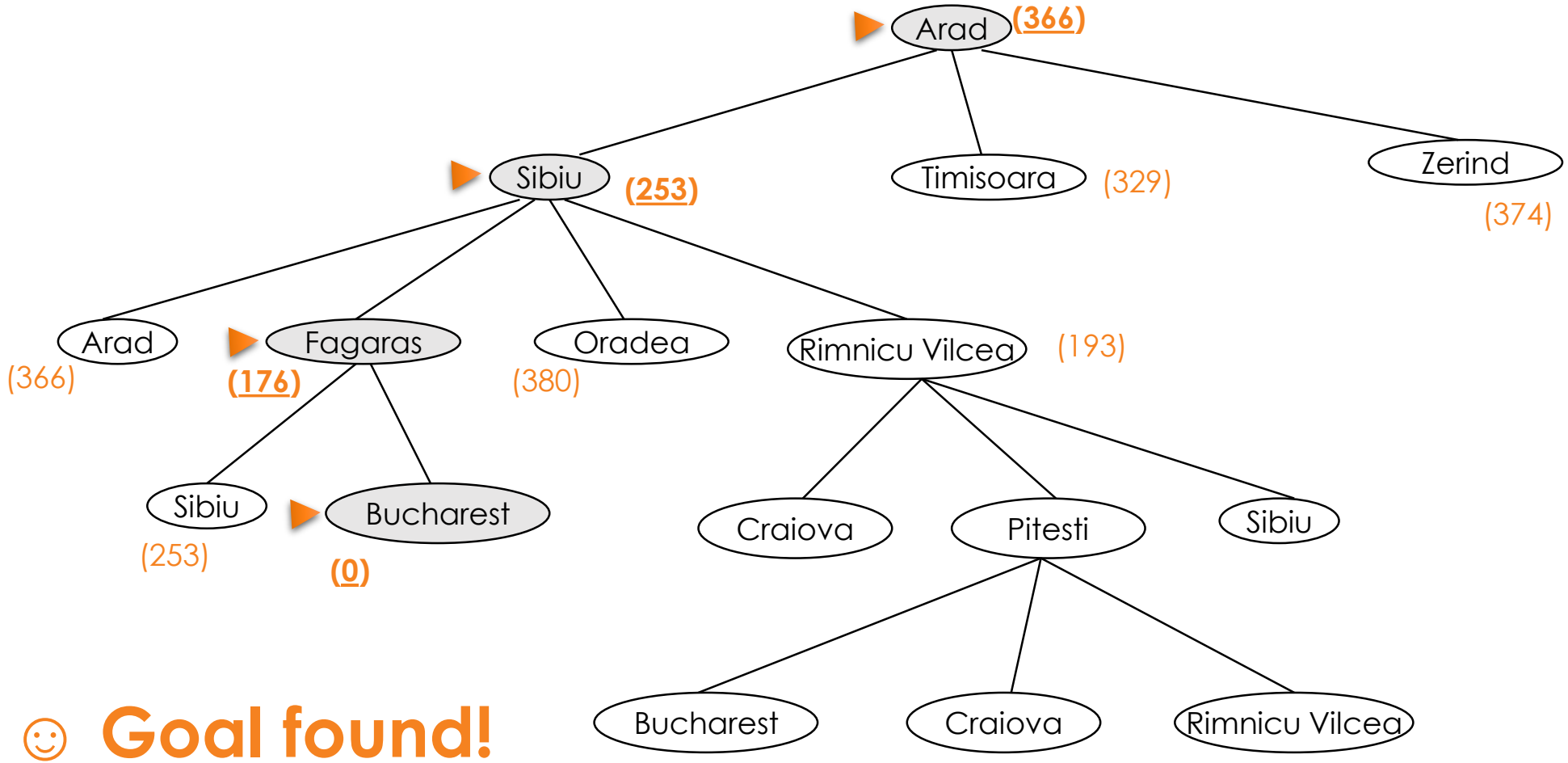
Informed Search Techniques

Hill Climbing (Greedy Best First Search)



Informed Search Techniques

Hill Climbing (Greedy Best First Search)

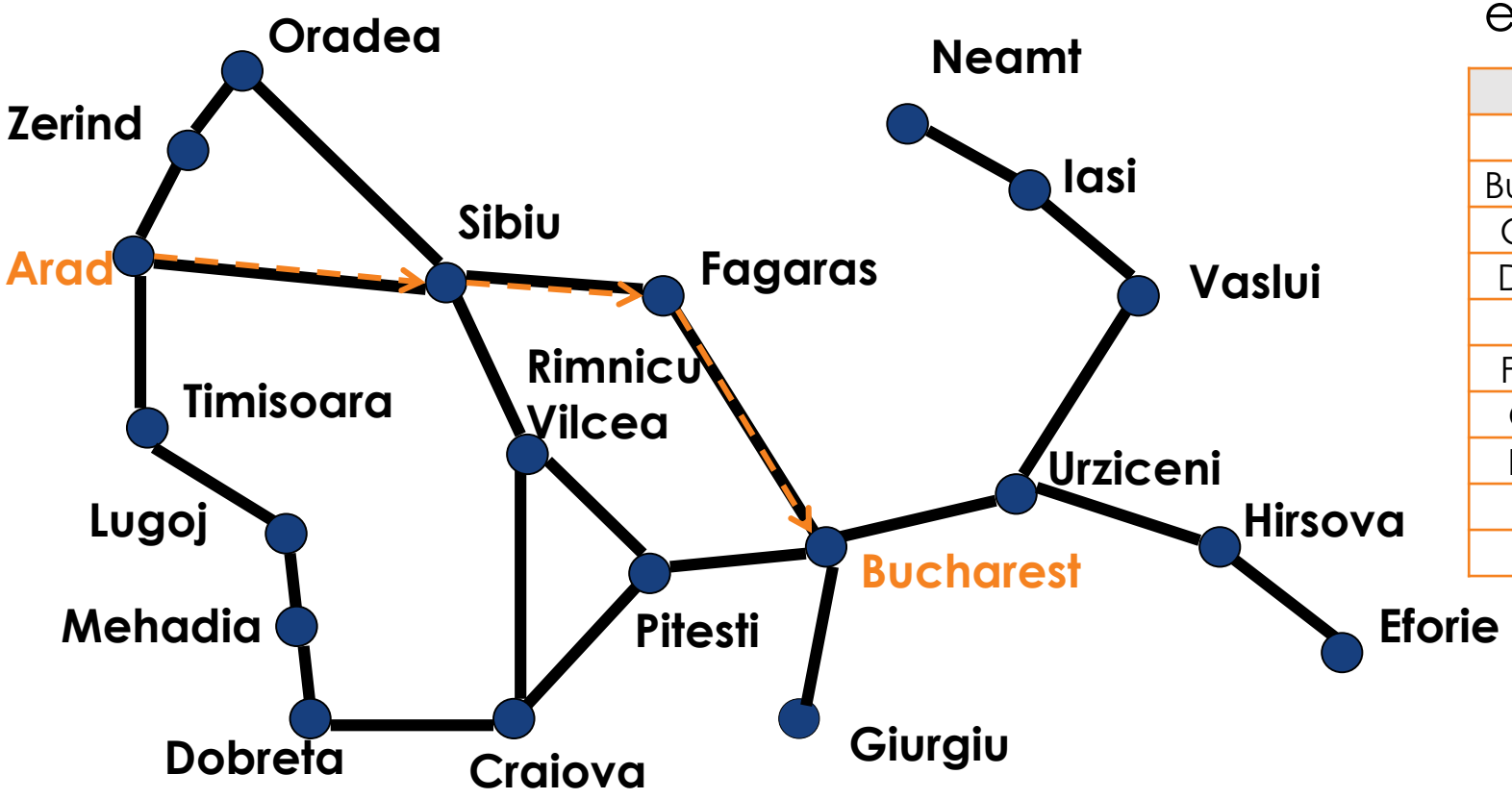


☺ **Goal found!**

Informed Search Techniques

Hill Climbing (Greedy Best First Search)

- **Arad → Bucharest**



h(n) =Straight-line distance from each city to **Bucharest**

City	Distance	City	Distance
Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Dobreta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

Informed Search Techniques

A* Search

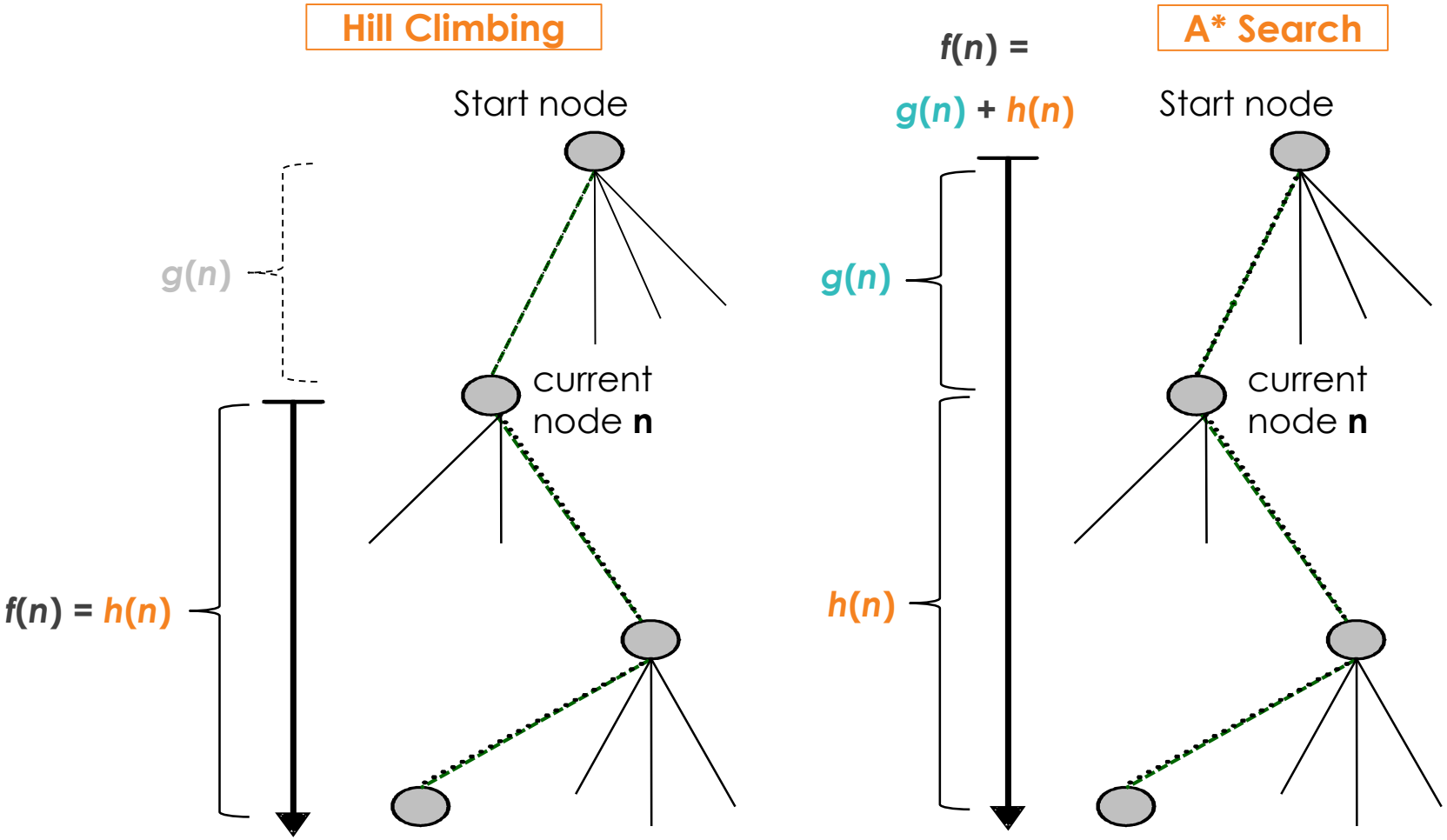
- **A* search is the most widely-known form of best-first search**
  - This strategy evaluates each search node by combining ***g(n)***, the past (path) cost from the start node to current node ***n***, and ***h(n)***, the estimated future (path) cost: the cheapest path/cost from current node ***n*** to a goal node
  - Estimated total cost of the cheapest solution through ***n***

$$f(n) = g(n) + h(n)$$

☺ ***g(n)*** **is exactly known, but** ***h(n)*** **is only an estimation with possible error.**

Informed Search Techniques

A* Search vs. Hill Climbing



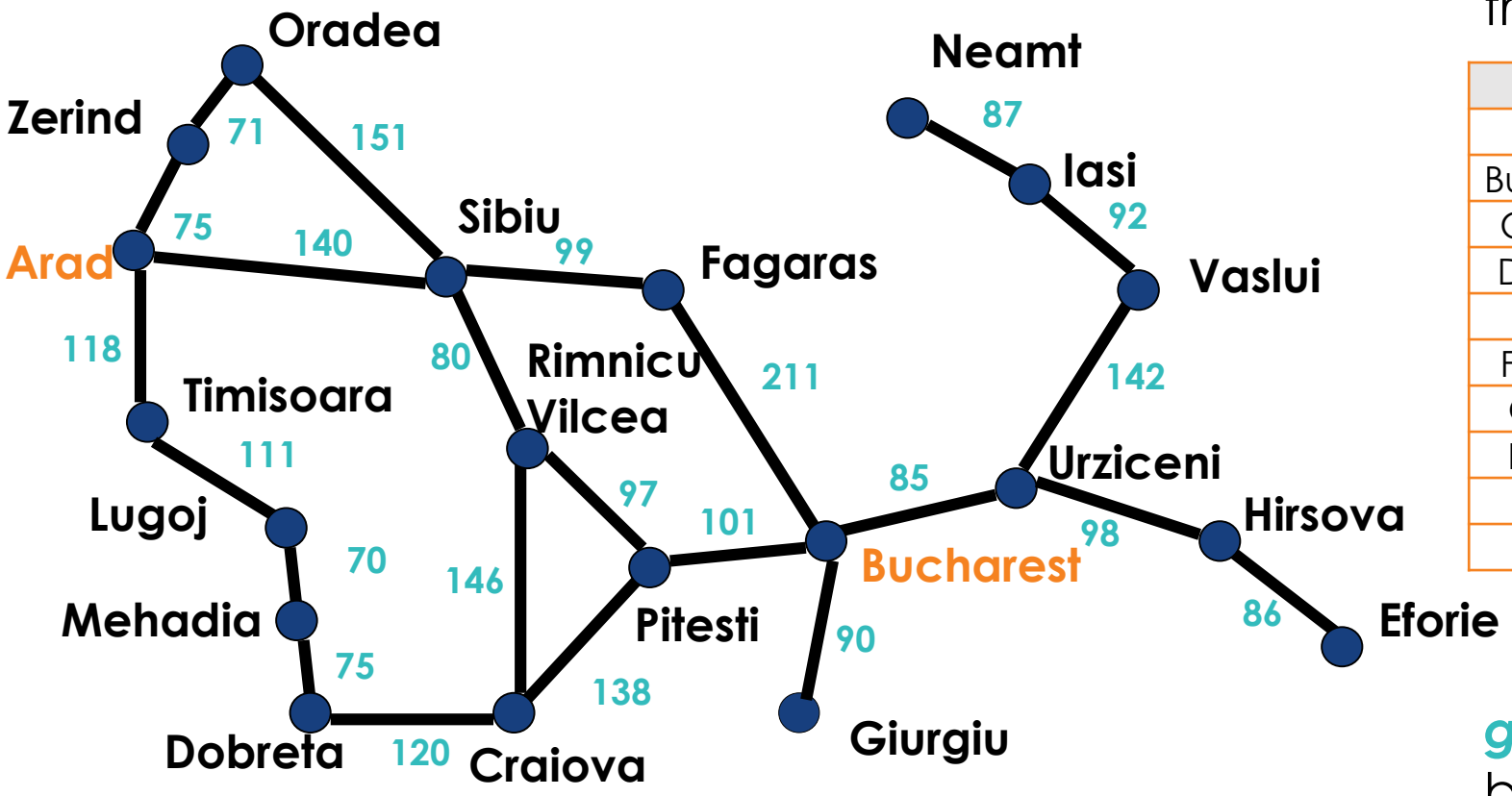
Select then expand "best-path-**from-n**-to-goal" child-node at each layer

Select then expand "best-path-**from-start**-to-goal" child-node at each layer

Informed Search Techniques

A* Search

- **Arad → Bucharest**



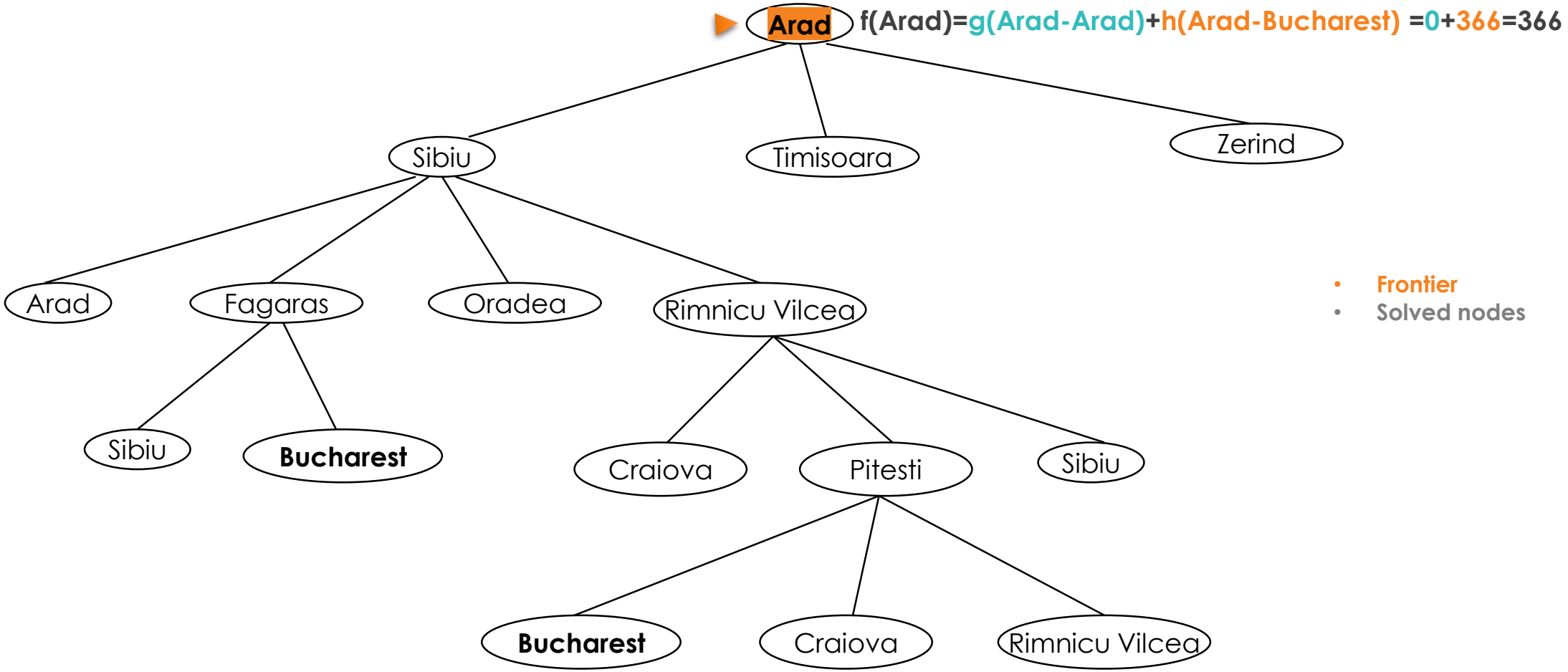
h(n) = Straight-line distance from each city to **Bucharest**

City	Distance	City	Distance
Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Dobreta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

g(n) = Actual path distance between adjacent cities

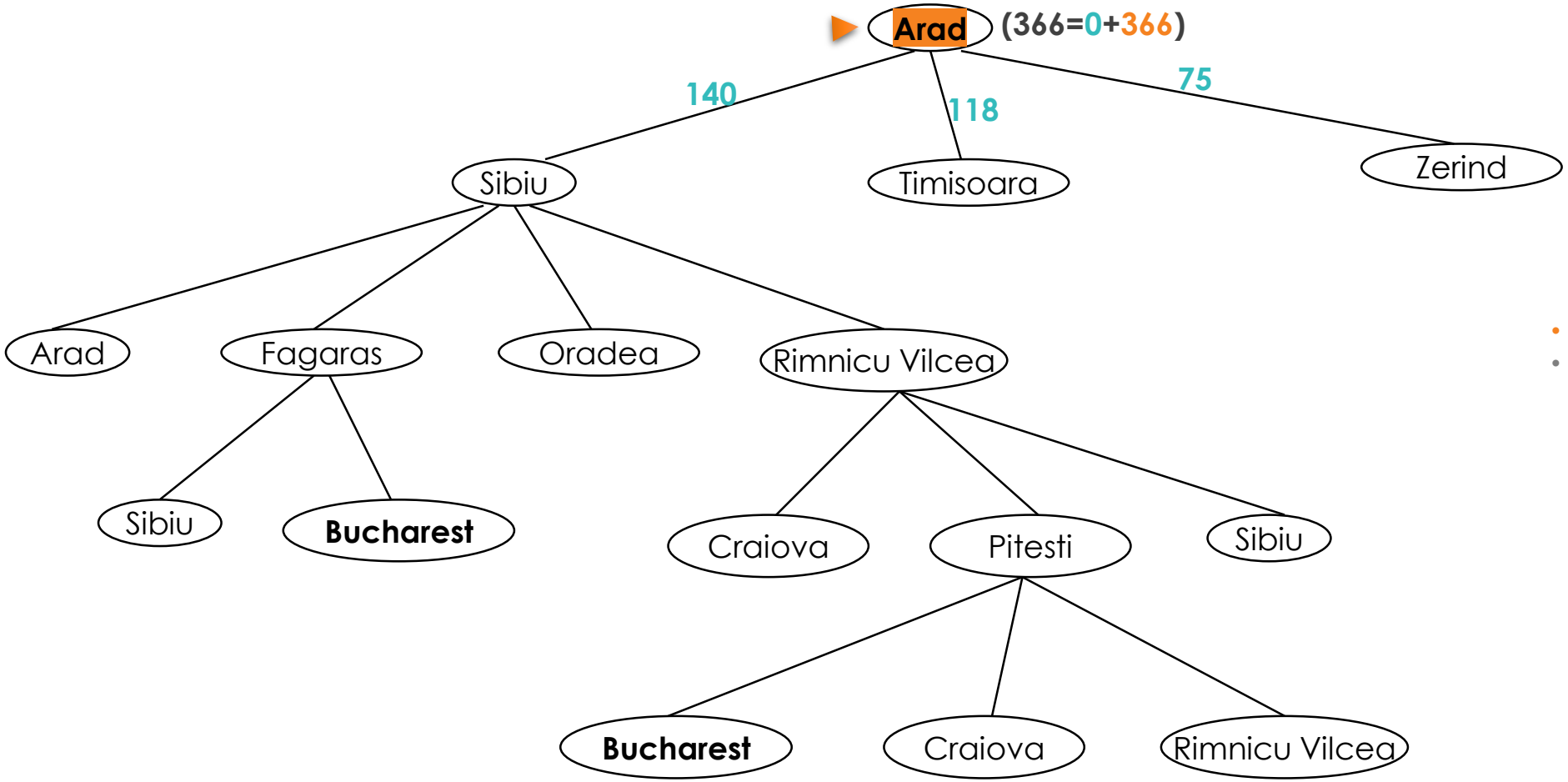
Informed Search Techniques

A* Search



Informed Search Techniques

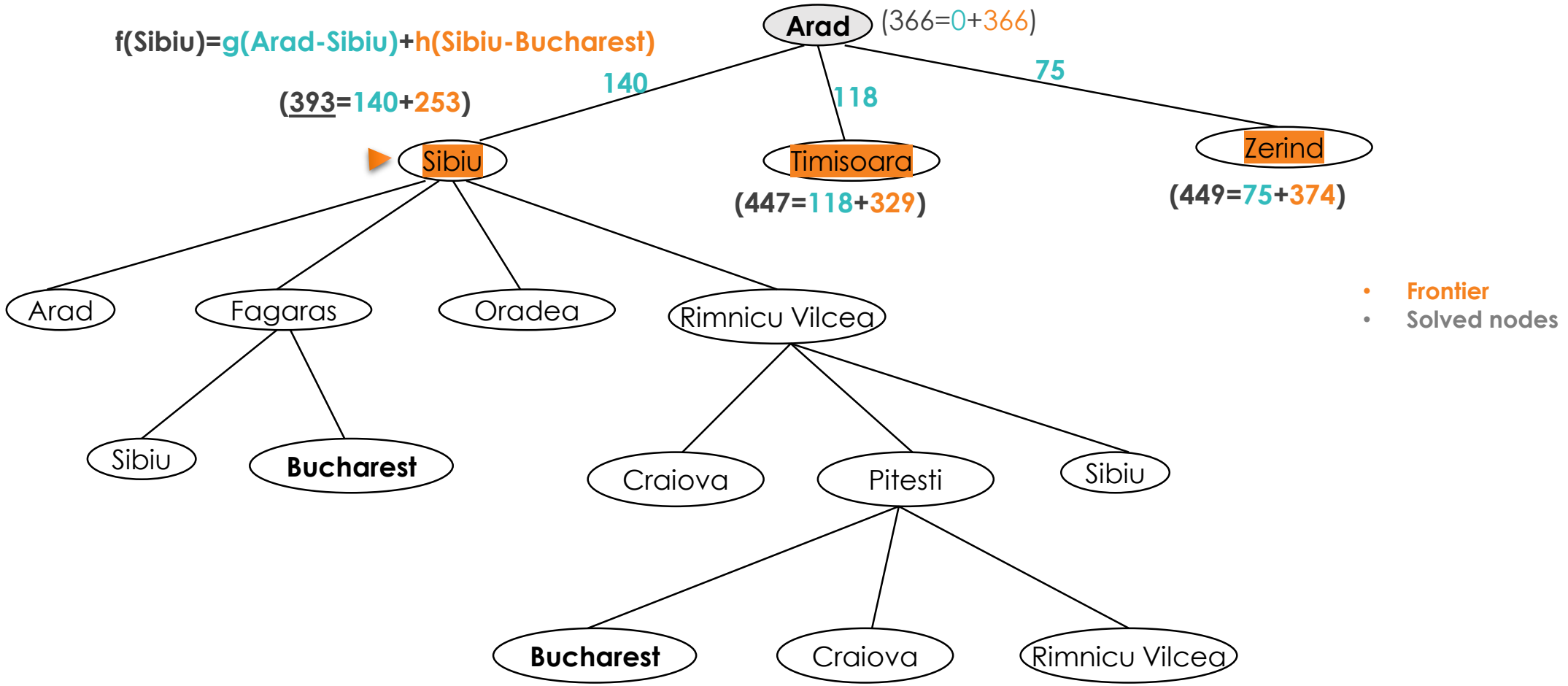
A* Search



- Frontier
- Solved nodes

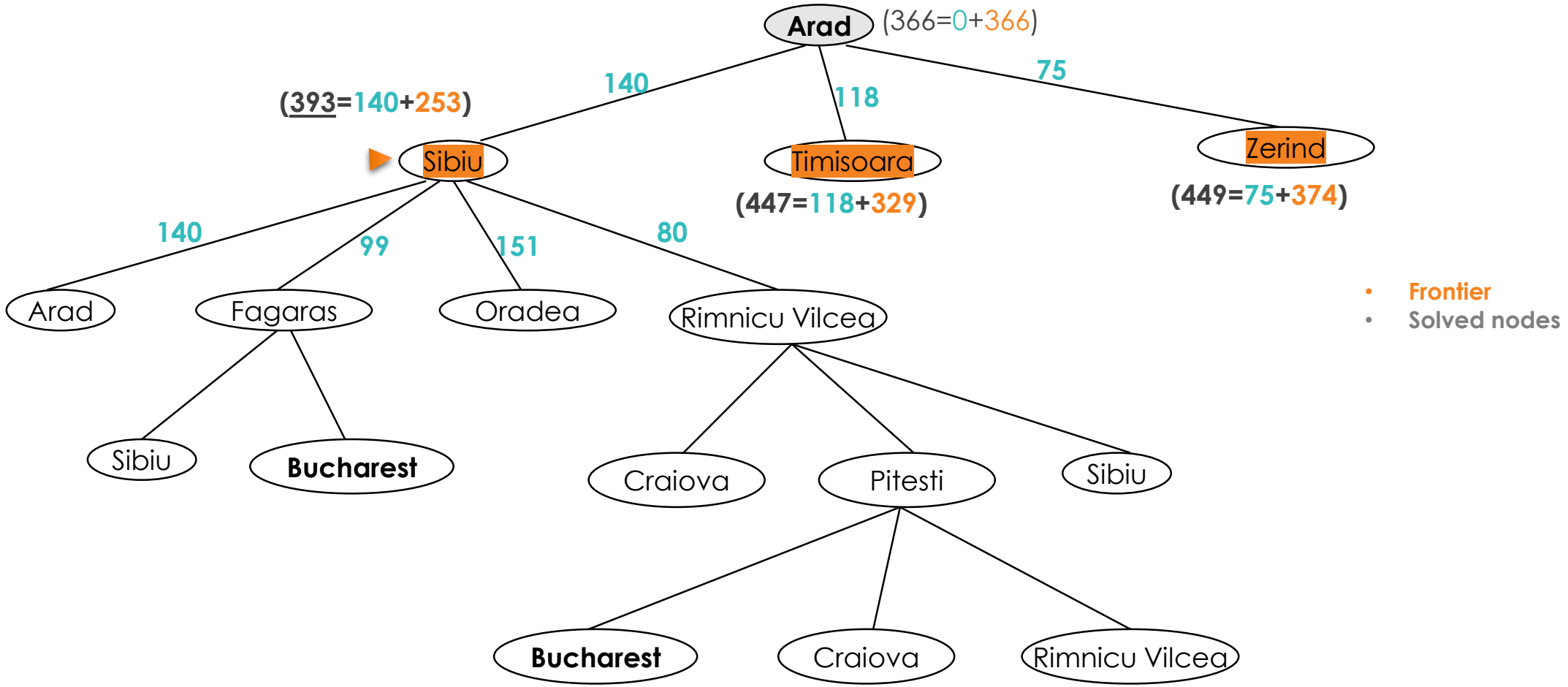
Informed Search Techniques

A* Search



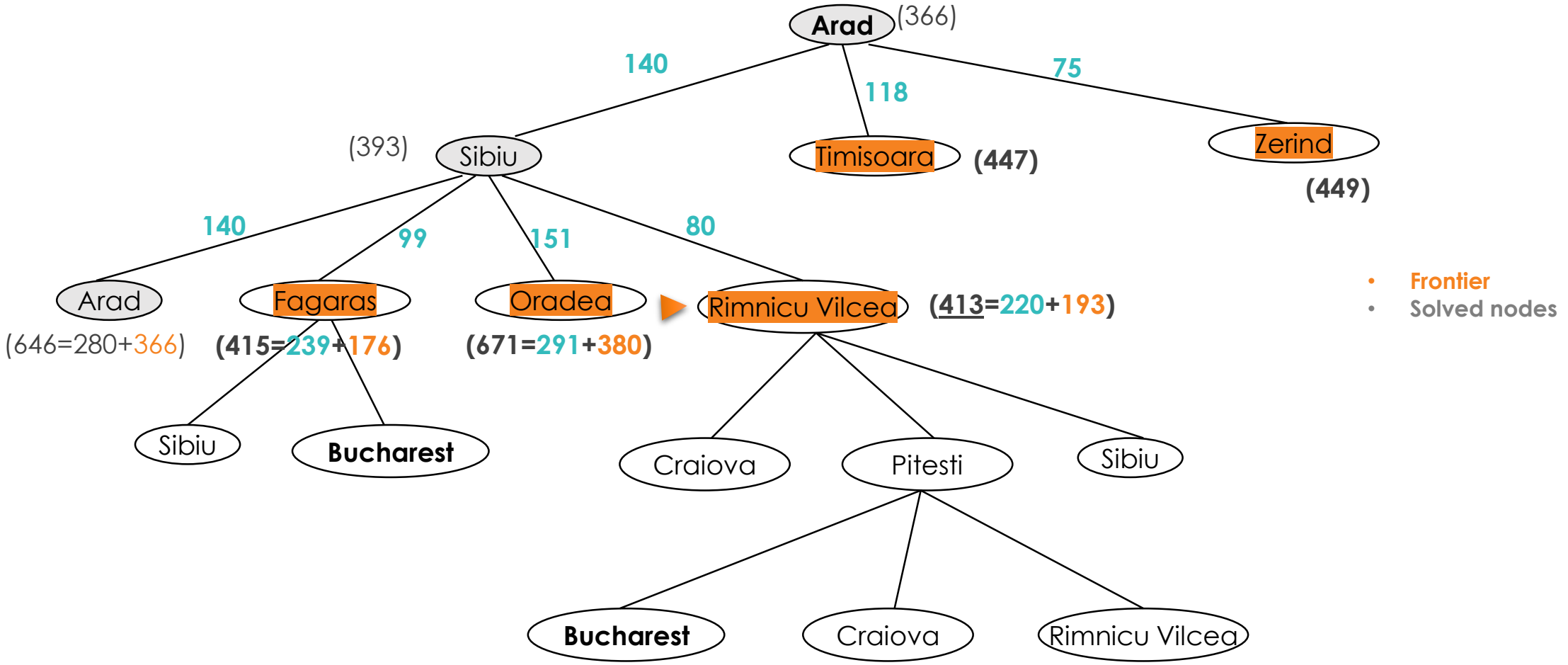
Informed Search Techniques

A* Search



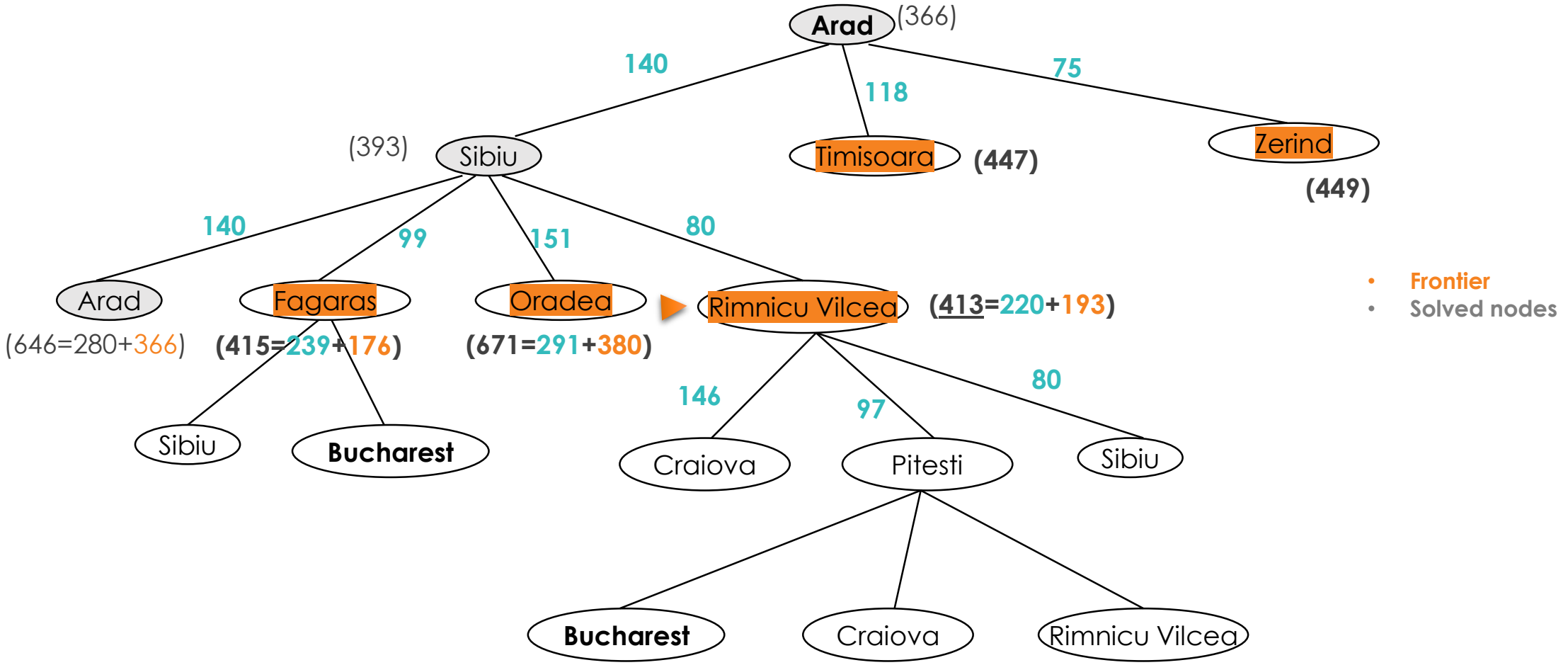
Informed Search Techniques

A* Search



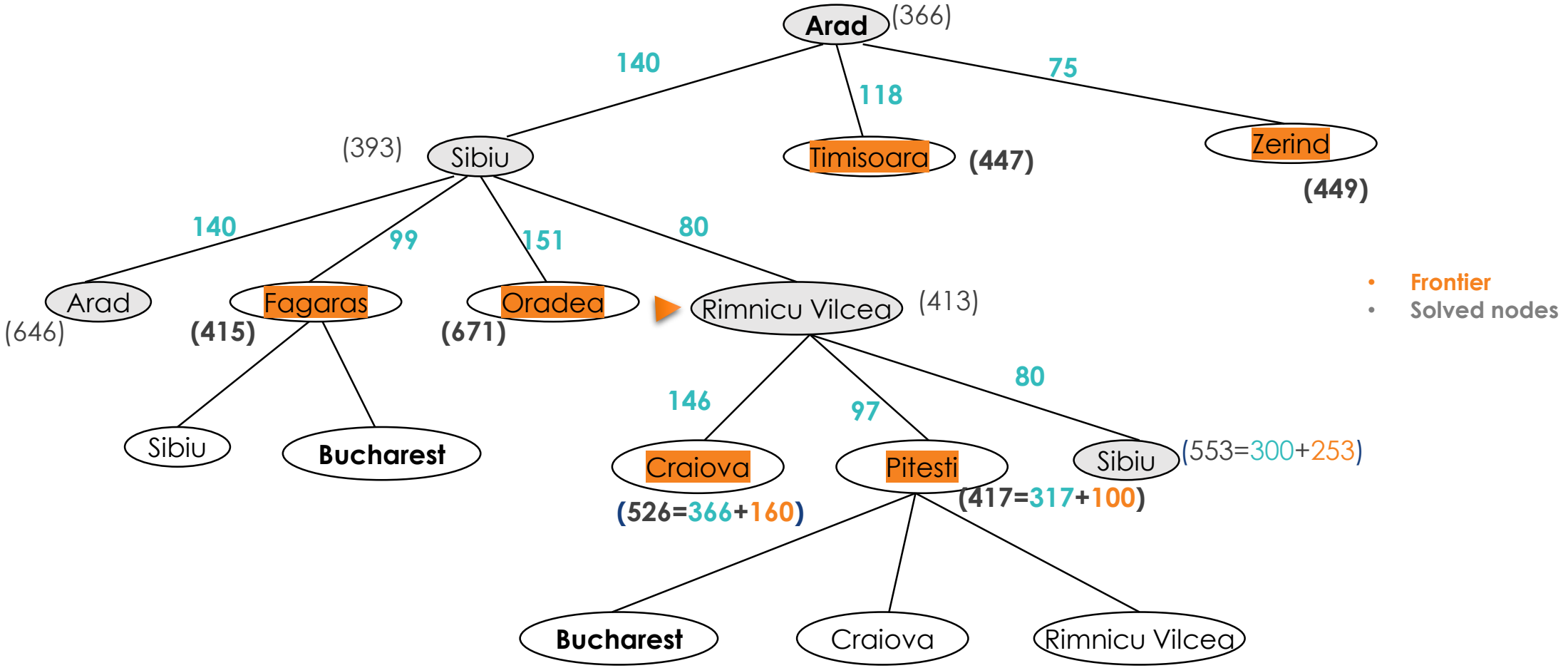
Informed Search Techniques

A* Search



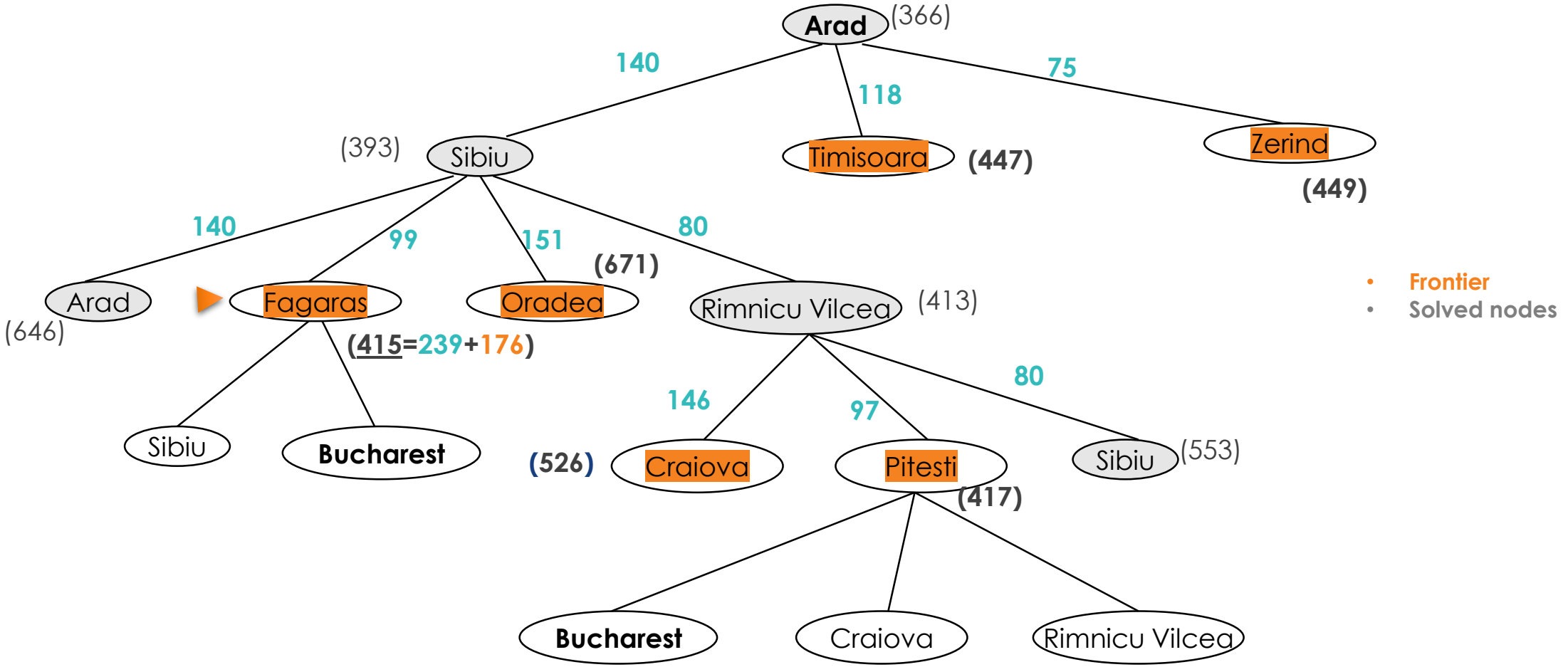
Informed Search Techniques

A* Search



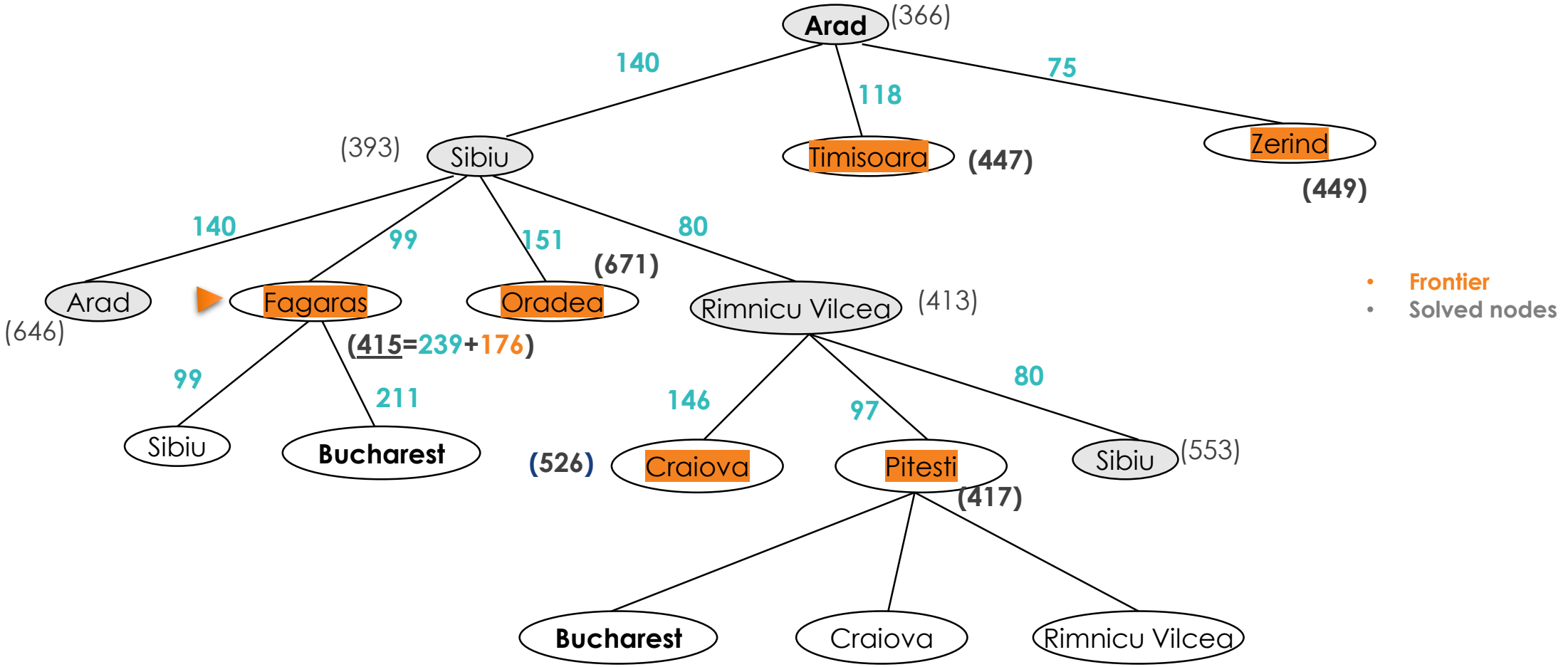
Informed Search Techniques

A* Search



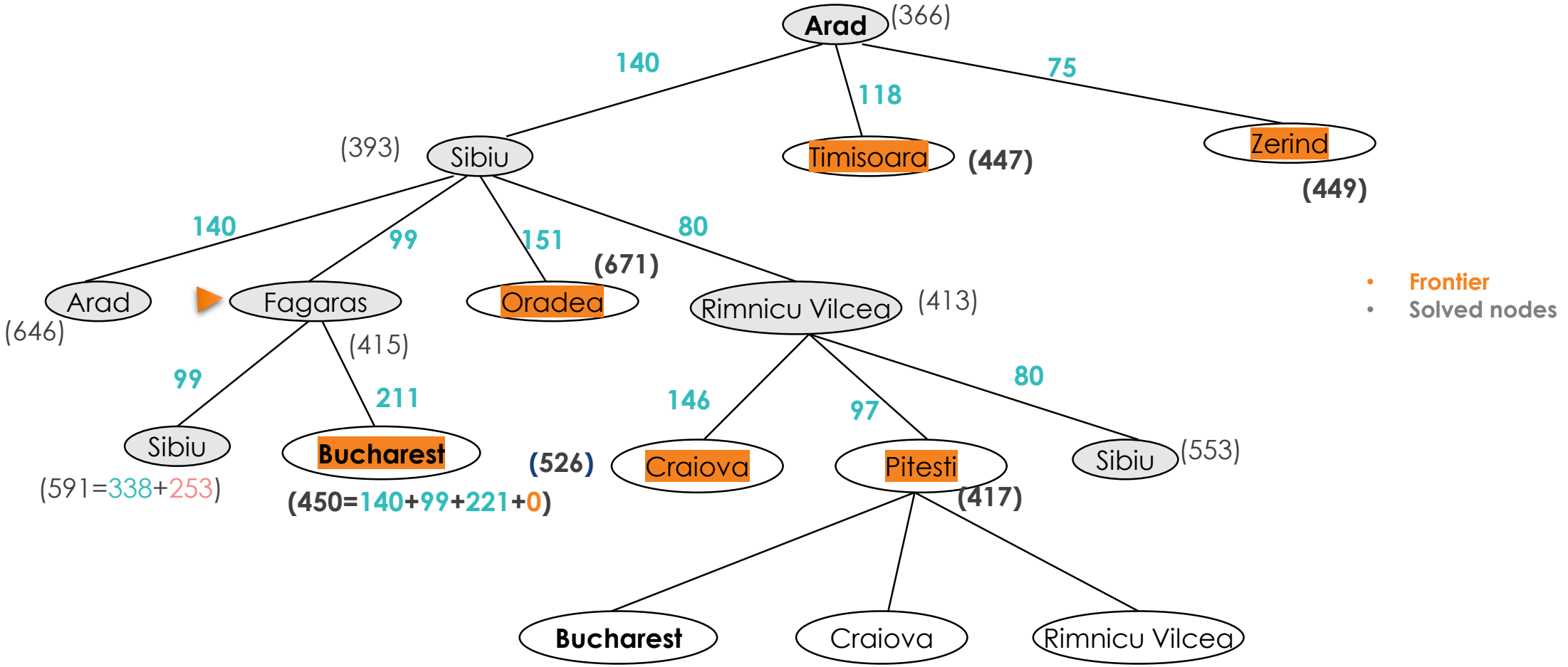
Informed Search Techniques

A* Search



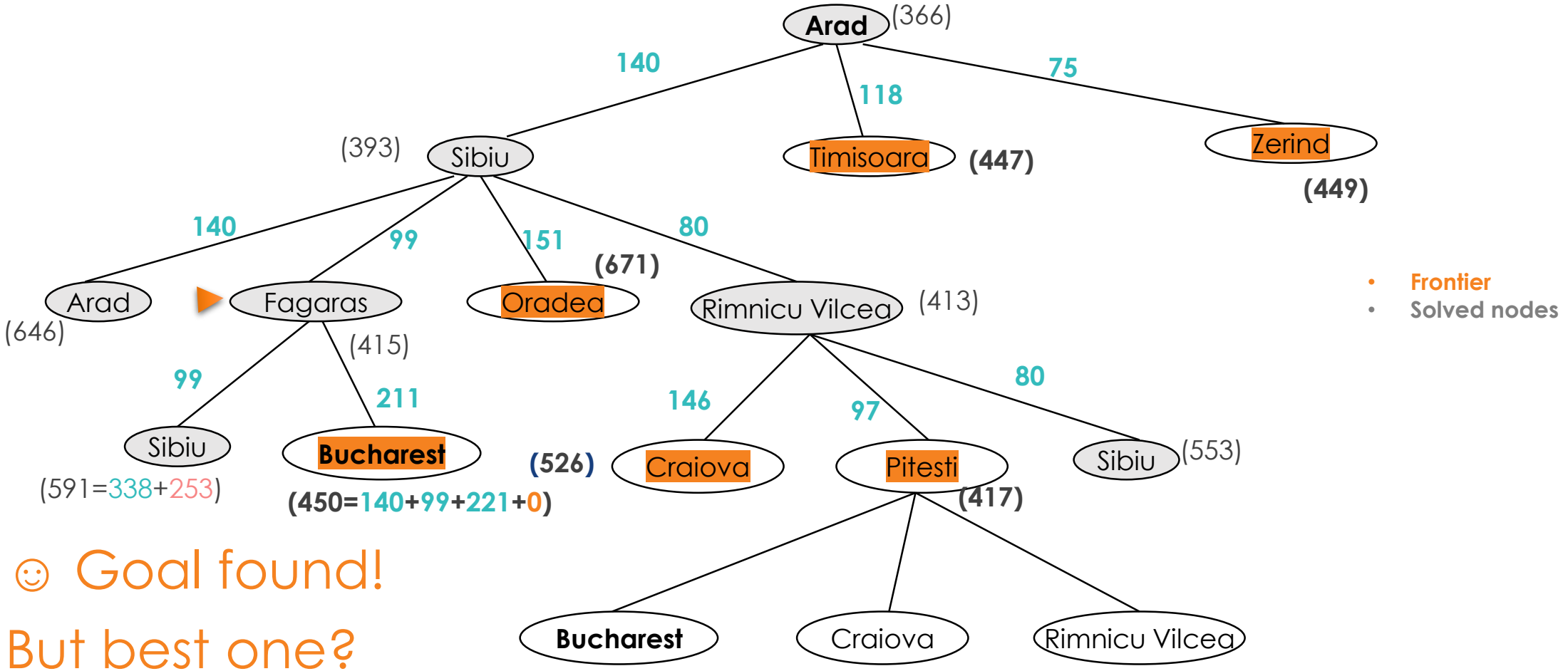
Informed Search Techniques

A* Search



Informed Search Techniques

A* Search



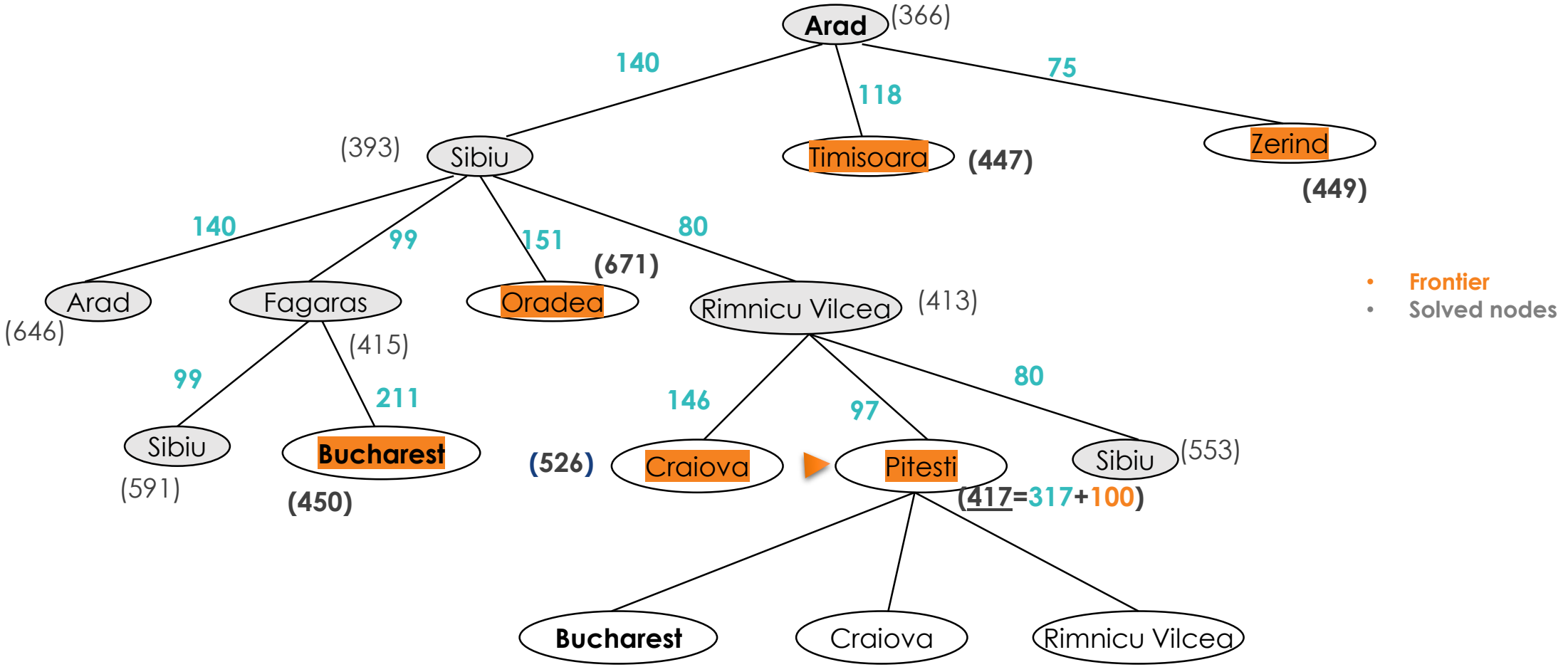
☺ Goal found!

But best one?

(Path distance/cost **450** is not in front of frontier node list)

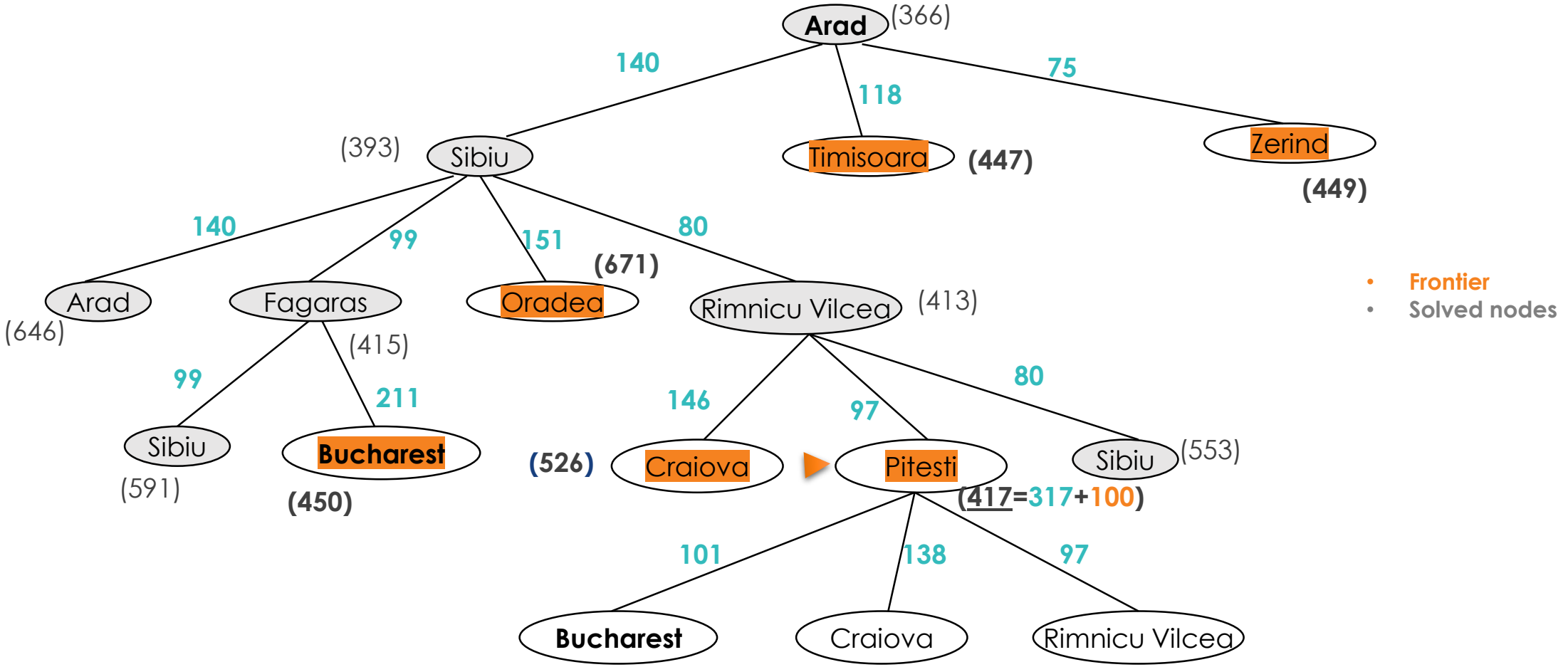
Informed Search Techniques

A* Search



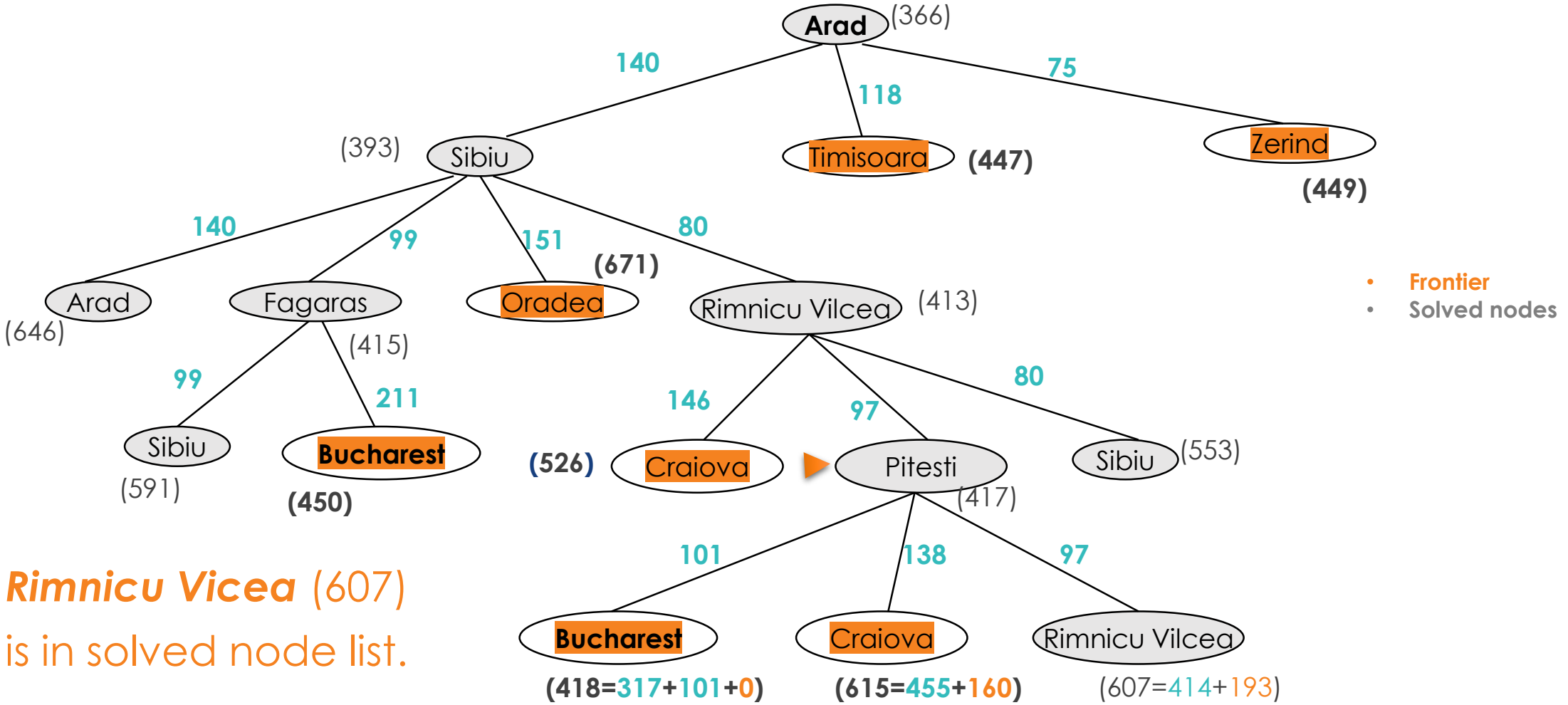
Informed Search Techniques

A* Search



Informed Search Techniques

A* Search

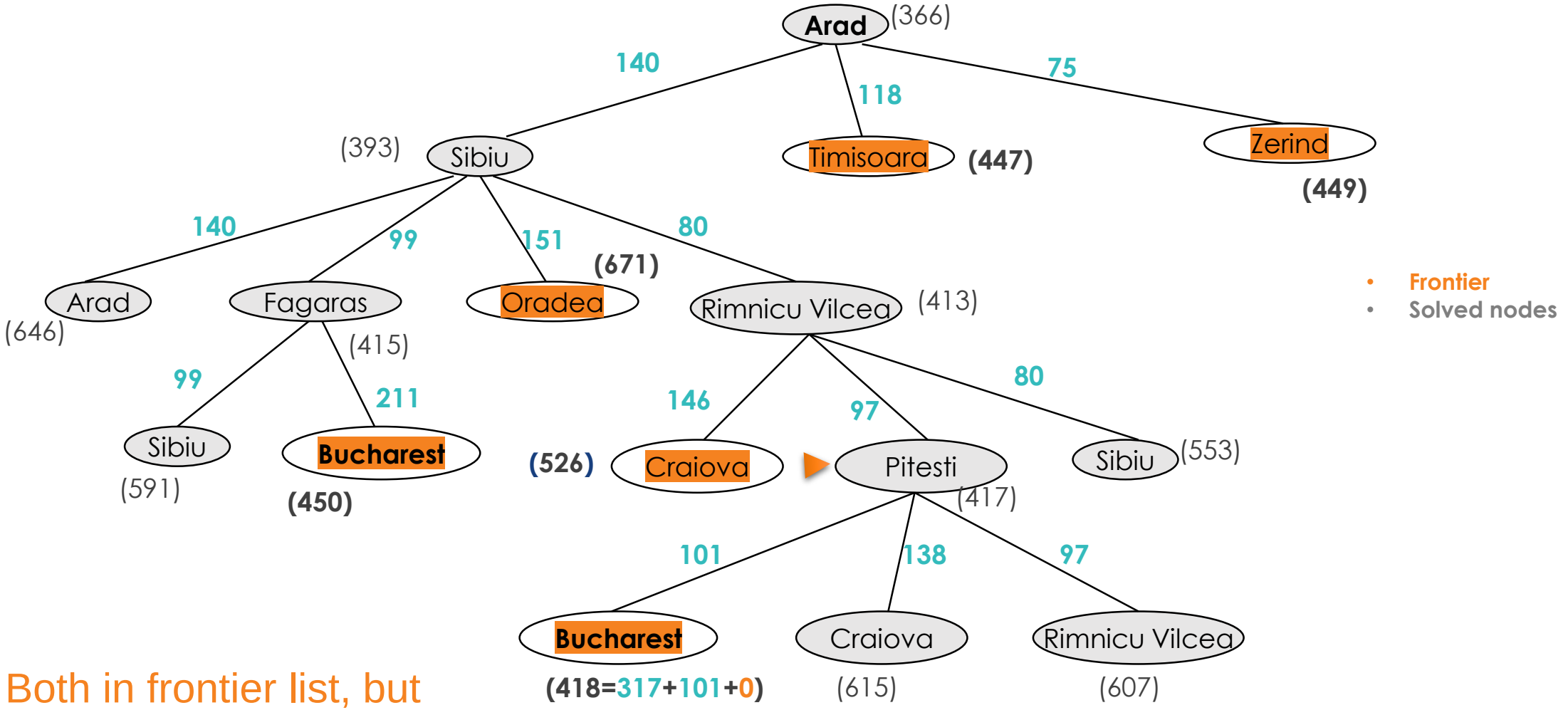


Rimnicu Vicea (607)
is in solved node list.

What about ***Craiova*** (615)?

Informed Search Techniques

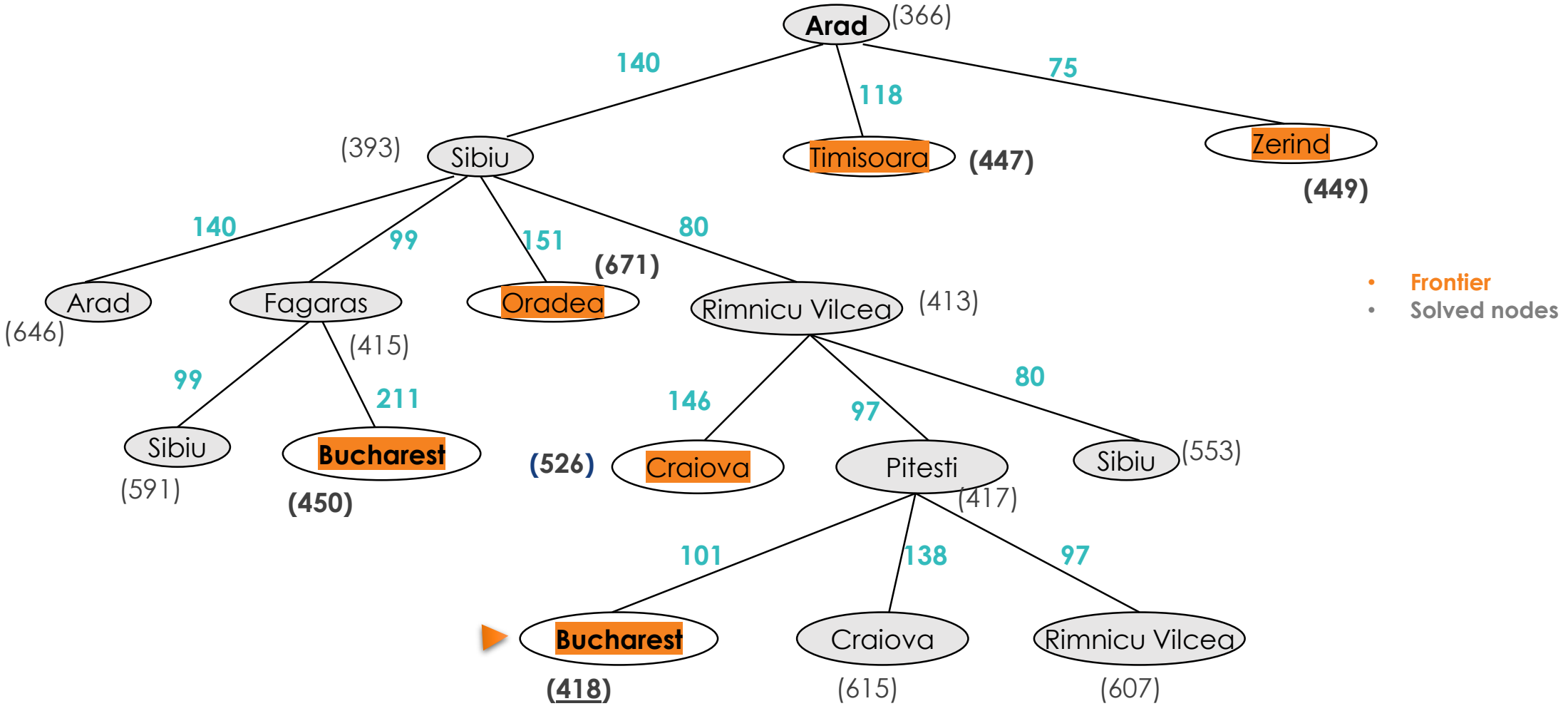
A* Search



Both in frontier list, but ***Craiova*** (615) is worse than ***Craiova*** (526).

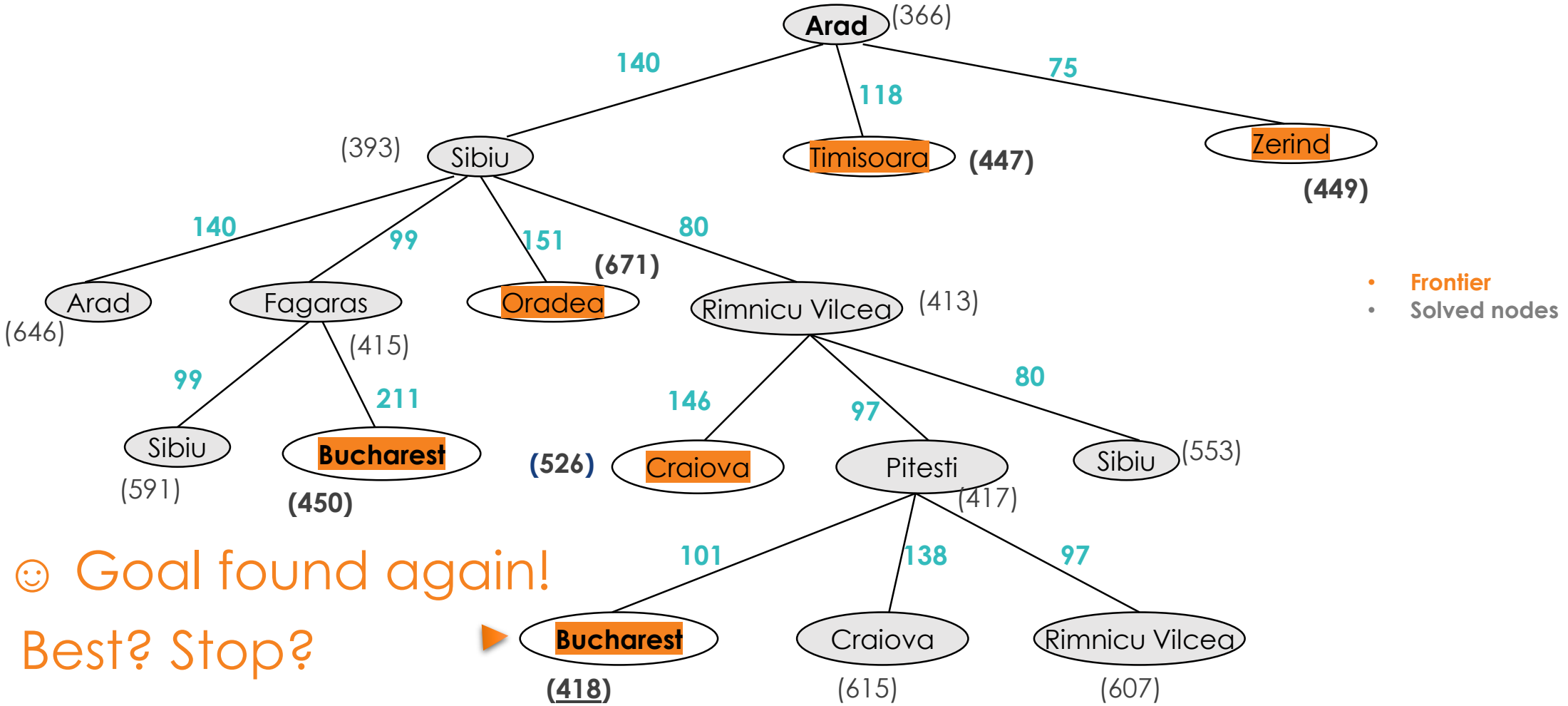
Informed Search Techniques

A* Search



Informed Search Techniques

A* Search

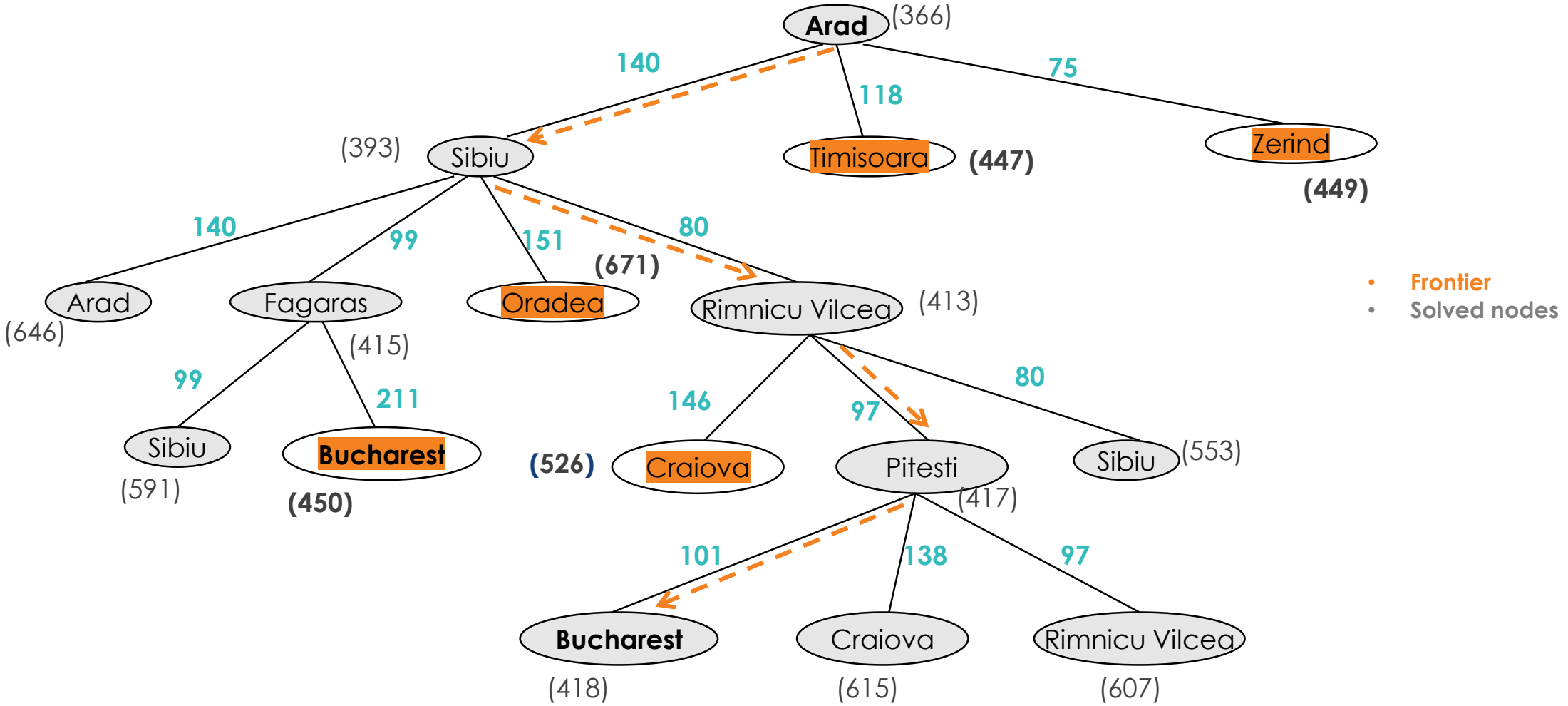


☺ Goal found again!

Best? Stop?

Informed Search Techniques

A* Search



2.1 Reasoning Using Informed Search

2.1.1 Machine Learning ≈ Search

2.1.2 Informed Search Techniques

2.1.3 Exercise

[Exercise] Reasoning Using Informed Search

- **Compare the results among A* Search & Hill Climbing & Uninformed search (Dijkstra's algorithm).**

- **Are they the same, in terms of iterations (e.g. number of updates to *Frontier List*) to reach the goal city?**

- **Which algorithm can find the *optimal* goal? And *most rapidly/efficiently*?**

2.2 Search Based Reasoning Applications

2.2.1 Reasoning System Case Studies

2.2.2 Reasoning System Demos

2.2.3 Exercise

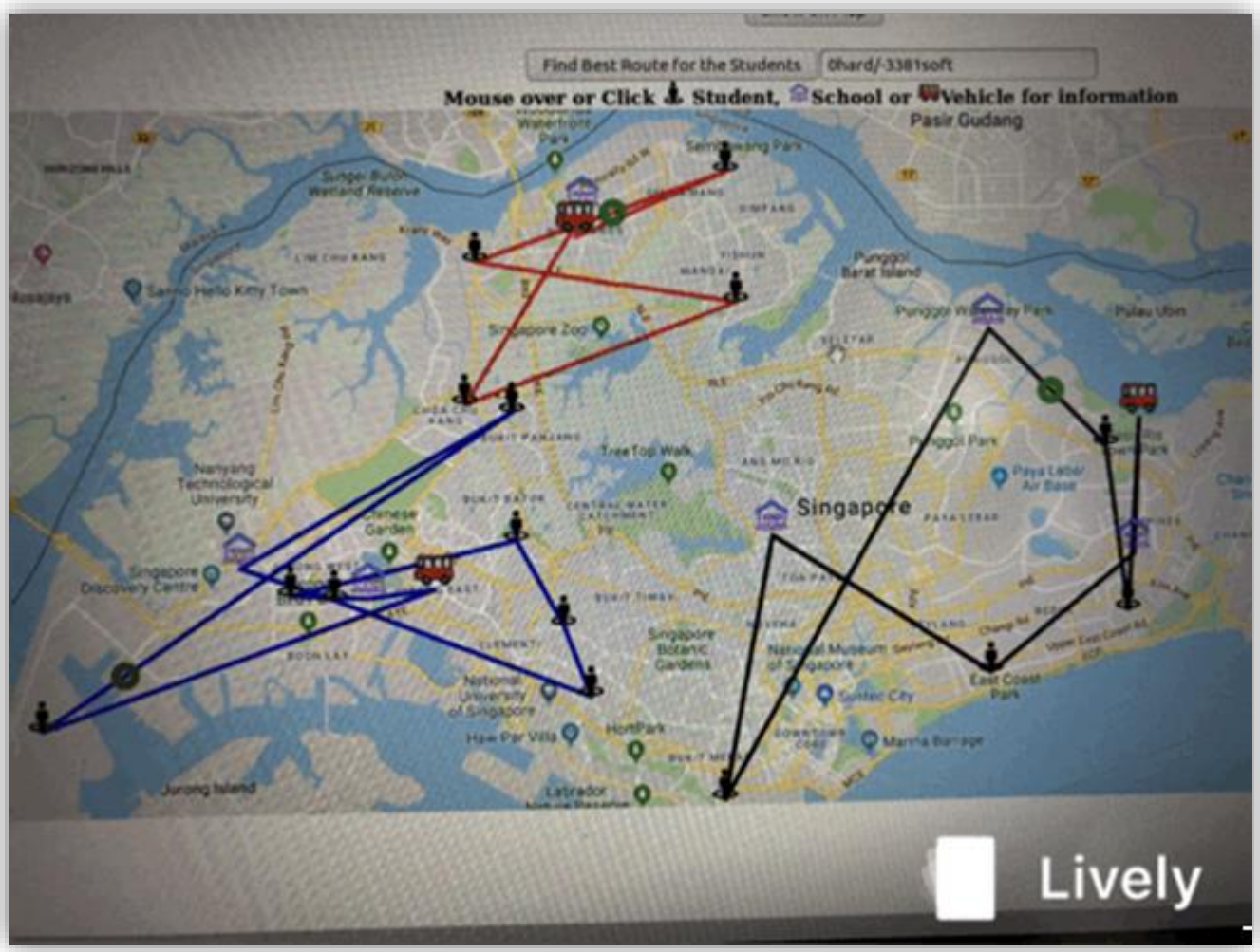
2.2 Search Based Reasoning Applications

2.2.1 Reasoning System Case Studies

2.2.2 Reasoning System Demos

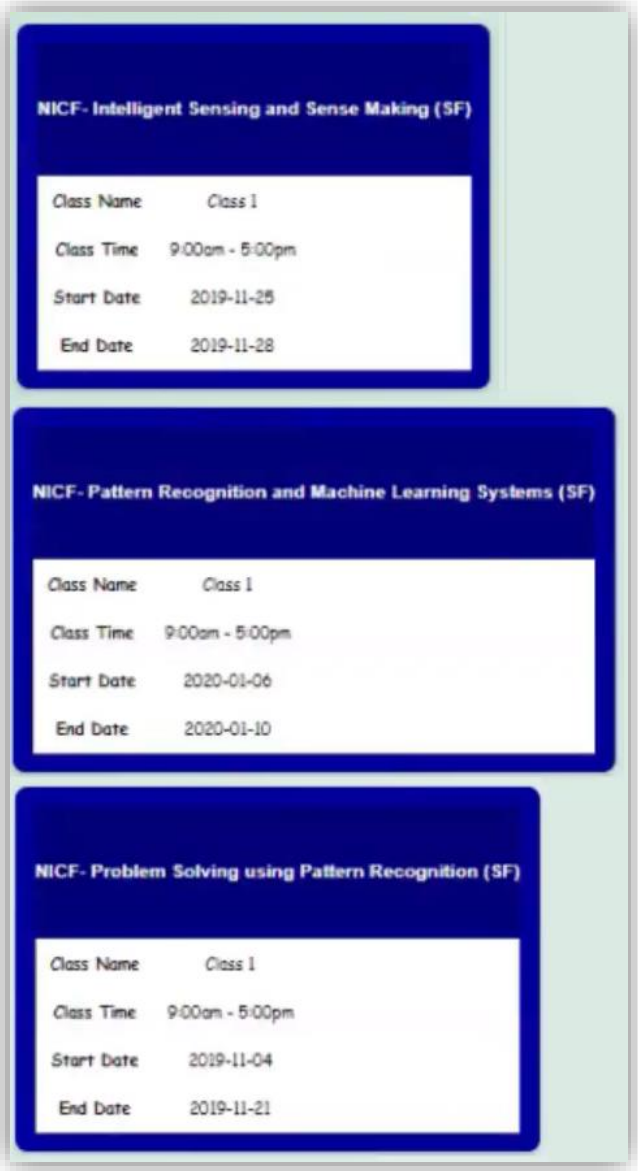
2.2.3 Exercise

[Logistics] Intelligent Rapid Shuttle (IRS) System



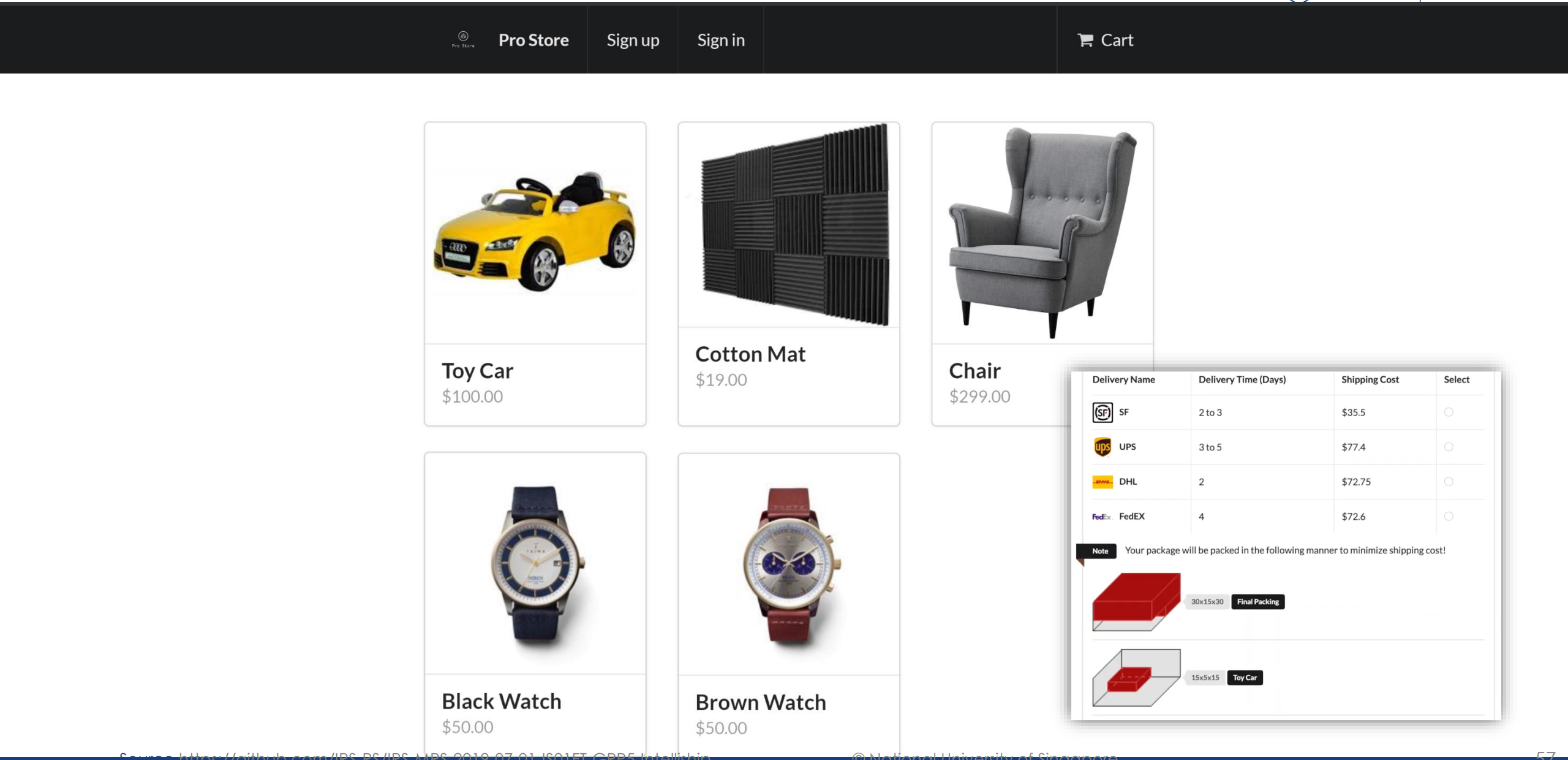
Source https://github.com/IRS-RS/IRS-RS-2019-03-09-IS1PT-GRP-aiVoyagers-irs-Intelligent-Rapid-Shuttle

[Education] ISS Course Recommender



Source: https://github.com/IRS-RS/IRS-MRS-2019-07-01-IS01FT-GRP1-ISS-COURSE-RECOMMENDER-SYSTEM

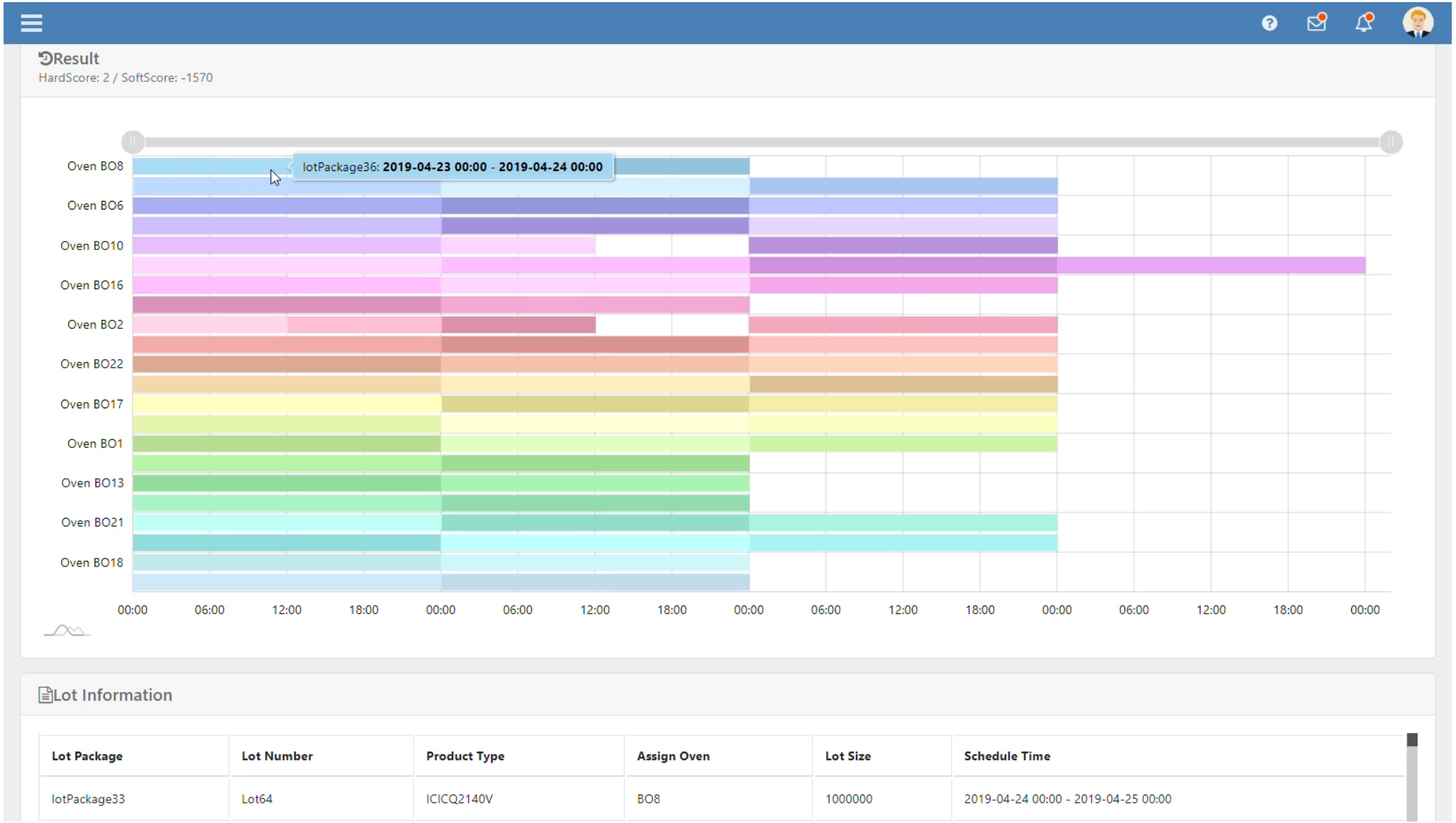
[E-Commerce] Shipping and Packing Optimizer



Source: https://github.com/IRS-RS/IRS-MRS-2019-07-01-IS01FT-GRP5-Intelliship

[FinTech] Co-branded Petrol Credit Card

[Manufacturing] Integrated Circuits Lot-Oven Scheduling and Dispatch Optimizer



Source: https://github.com/IRS-RS/IRS-RS-2019-03-09-IS01PT-GRP-Hex-Sdot_Scheduling_and_Dispatch_Optimization_Tool

ENTERPRISE KNOWLEDGE GRAPH SYSTEM

(KNOWLEDGE GRAPH SOLUTION THAT LEADS TO ENTERPRISE AI)

Team 2 Members: **YANG Lu Yi, YIN Tian Shi, YU Yu**

Source: https://github.com/IRS-PM/IRS-PM-2020-01-18-IS02PT-GRP2-Enterprise-Knowledge-Graph

YANG LU YI

Integrated AI: Hybrid Reasoning Systems

Intelligent Assistant: Mycroft (open source)

https://github.com/MycroftAI

Source https://www.kickstarter.com/projects/aiforeveryone/mycroft-an-open-source-artificial-intelligence-for

[Health] Meal Planner for Diabetics (MP-D)

Source: https://github.com/IRS-RS/IRS-MRS-2019-07-01-IS01FT-GRP9-Meal-Planner-for-Diabetics

Meal Planner for Diabetics

- **Problem:**

  Diabetes Mellitus is a chronic condition in which the body has elevated glucose levels. It is a global public health problem and Type 2 diabetes accounts for around 90% of diabetes cases worldwide. Singapore ranks second in the proportion of diabetics among developed countries, behind US, with 10.63% of the population having this condition. Disturbingly, the prevalence of diabetes in Singapore is set to increase from 7.3% in 1990 to 15% in 2050, as people lead more sedentary lifestyles and consume high-energy diets.

  Type 2 diabetes is a costly disease that can cause complications like blindness and lower limb amputation if not properly managed. Dietary intervention is key to maintaining blood glucose levels and managing the disease but people with the condition may not necessarily have the knowledge to make better food choices or they may find meal planning to be too time-consuming.

- **Solution:**

  Our product team developed a meal planning system to help plan suitable meals and make the solution easier to manage user's diabetic condition.

Meal Planner for Diabetics

- **What does a meal plan (solution for diabetic user) look like?**

- **For a team of 3-4, then provide your team based answers in this online sheet:**

ISSGZ, THIS IS YOUR PERSONALISED MENU FOR THE WEEK

WITH AN ESTIMATED 1730.0 KCAL/DAY

More Details

*The numbers on the right of the menu show calories (kcal) for the meal

DAY 1

breakfast

Mr Bean grass jelly soya milk 138.80

Mr Bean chicken mayo pancake 291.00

mid-morning snack

Chinese pear 53.71

lunch

Black pearl cereal beverage 129.96

Uttapam 262.40

mid-afternoon snack

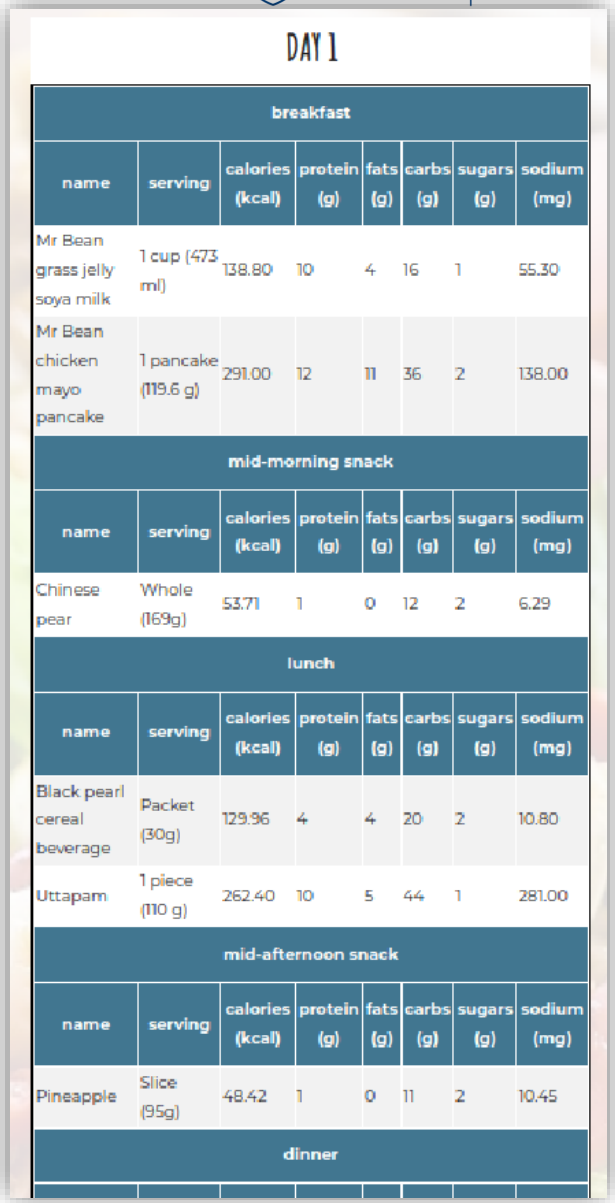
Pineapple 48.42

dinner

Soyabean milk without sugar 88.00

Fried hong kong egg noodles 693.21

DAY 2



DAY 1

breakfast							
name	serving	calories (kcal)	protein (g)	fats (g)	carbs (g)	sugars (g)	sodium (mg)
Mr Bean grass jelly soya milk	1 cup (473 ml)	138.80	10	4	16	1	55.30
Mr Bean chicken mayo pancake	1 pancake (119.6 g)	291.00	12	11	36	2	138.00
mid-morning snack							
name	serving	calories (kcal)	protein (g)	fats (g)	carbs (g)	sugars (g)	sodium (mg)
Chinese pear	Whole (169g)	53.71	1	0	12	2	6.29
lunch							
name	serving	calories (kcal)	protein (g)	fats (g)	carbs (g)	sugars (g)	sodium (mg)
Black pearl cereal beverage	Packet (30g)	129.96	4	4	20	2	10.80
Uttapam	1 piece (110 g)	262.40	10	5	44	1	281.00
mid-afternoon snack							
name	serving	calories (kcal)	protein (g)	fats (g)	carbs (g)	sugars (g)	sodium (mg)
Pineapple	Slice (95g)	48.42	1	0	11	2	10.45
dinner							

Count of Food Name	Column Labels							
Row Labels	Breakfast	Drink	Fruits	Main	Salad	Snack	(blank)	Grand Total
chinese	28	4		283			50	365
indian	5			65				70
japanese				45			5	50
KFC	6			5			20	31
korean				25			2	27
malay	1			68			15	84
MrBean	5							5
peranakan							1	1
Popeyes							2	2
thai				6				6
vietnamese				4			3	7
western	42			132	8	1	11	194
(blank)	51	31	17		11	10	3151	3271
Grand Total	**138**	**35**	**17**	**633**	**19**	**11**	**3260**	**4113**

Count of Food Name	Column Labels							
Row Labels	Breakfast	Drink	Fruits	Main	Salad	Snack	(blank)	Grand Total
chinese	21	4		194				219
indian	5			49				54
japanese				42				42
malay	1			61				62
MrBean	5							5
western	17			58	6			81
(blank)	8	28	13			7		56
Grand Total	**57**	**32**	**13**	**404**	**6**	**7**		**519**

Meal Planner for Diabetics

- **What is a meal plan representation behind user interface?**

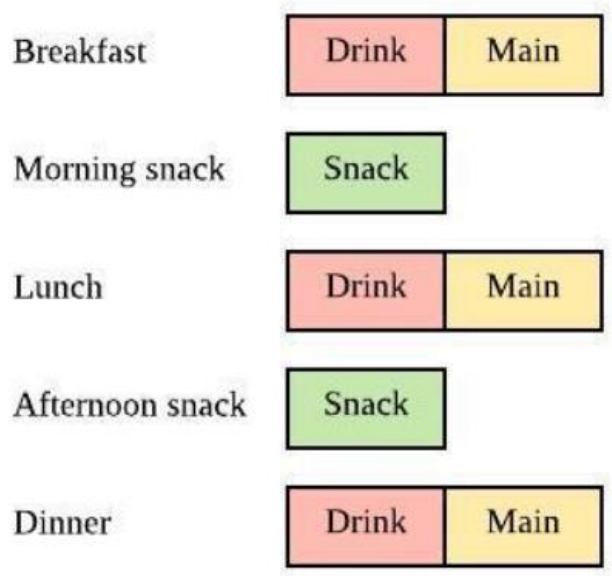
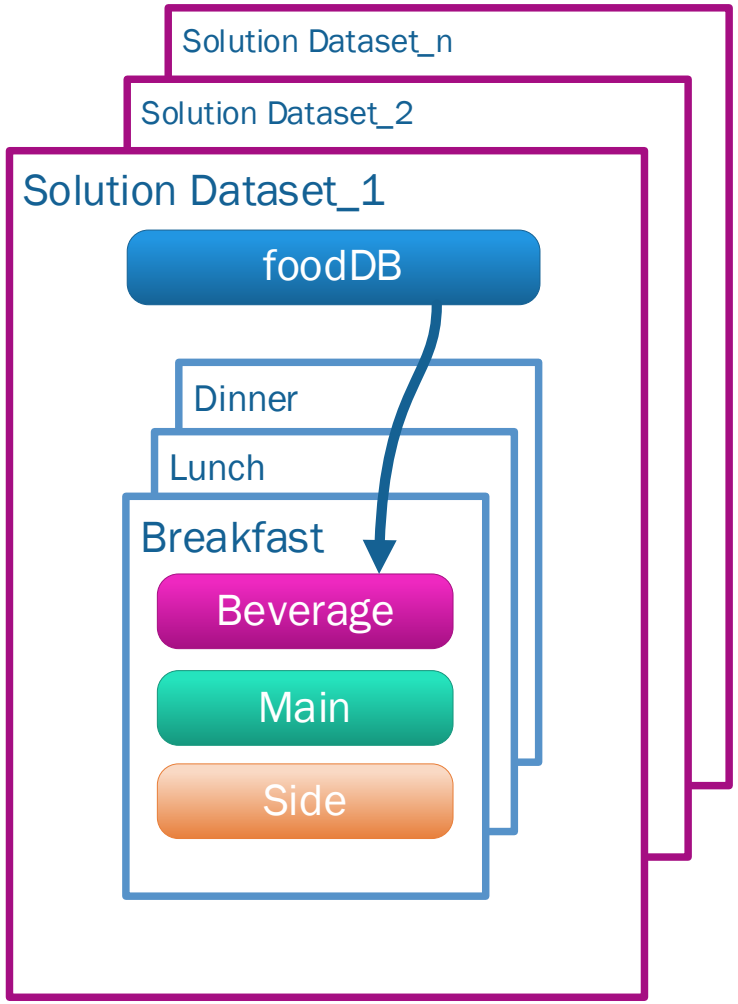
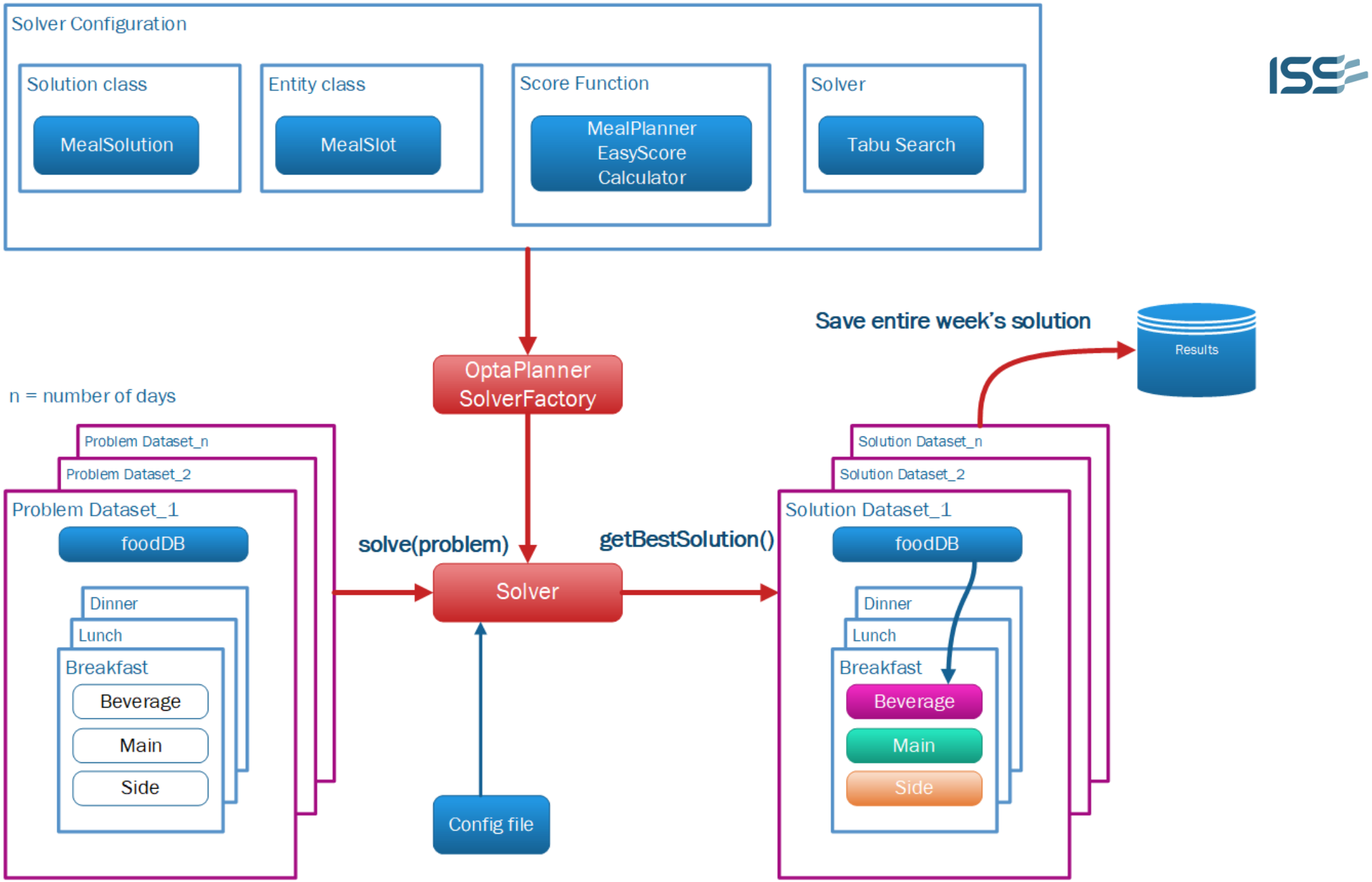


Figure 6 Meal plan structure for a day




Solver Configuration
Solution class
MealSolution
Entity class
MealSlot
Score Function
MealPlanner EasyScore Calculator
Solver
Tabu Search
OptaPlanner SolverFactory
Save entire week's solution
Results
n = number of days
Problem Dataset_n
Problem Dataset_2
Problem Dataset_1
foodDB
Dinner
Lunch
Breakfast
Beverage
Main
Side
solve(problem)
Solver
Config file
getBestSolution()
Solution Dataset_n
Solution Dataset_2
Solution Dataset_1
foodDB
Dinner
Lunch
Breakfast
Beverage
Main
Side

2.2 Search Based Reasoning Applications

2.2.1 Reasoning System Case Studies

2.2.2 Reasoning System Demos

2.2.3 Exercise

Meal Planner for Diabetics

- **Install and run MP-D.**

2.2 Search Based Reasoning Applications

2.2.1 Reasoning System Case Studies

2.2.2 Reasoning System Demos

2.2.3 Exercise

Meal Planner for Diabetics

- **Create your customized meal plan, using your student ID (or full name) during new user registration.**

2.3 Search Reasoning [Workshop]

2.3.1 Informed Search

Thanks to Geet Jethwani (A0215395B) for his contribution.

2.3.2 Reasoning System Exploration

2.3.3 Workshop Submission

2.3 Search Reasoning [Workshop]

2.3.1 Informed Search

Thanks to Geet Jethwani (A0215395B) for his contribution.

2.3.2 Reasoning System Exploration

2.3.3 Workshop Submission

Informed Search: Constraint Satisfaction Problem

- Understand what is ***Constraint Satisfaction Problem***
- How to formally define a CSP
- CSP Examples
- Resolving CSP using backtracking technique

Constraint Satisfaction Problem

- Constraint satisfaction problems (CSPs) are mathematical questions defined as a set of objects whose state must satisfy a number of constraints or limitations.

- CSPs represent the entities in a problem as a homogeneous collection of finite constraints over variables, which is solved by constraint satisfaction methods. -wiki

Define CSP

- CSP contains 3 components mainly:
- Variables , domains and constraints, where :
  - `V = {V1,V2….Vn} is a set of variables`
  - `D = {D1,D2….Dn} is a set of domains`
  - `C  = {C1,C2…Cn} is a set of constraints`
- Where ever constraint Ci is in turn a pair of (Tj , Rj)
- Tj is set of variables that are part of the constraint
- Rj is a relation that defines the values that variable can take

Solve CSP

Problems that can be modeled as a constraint satisfaction problem include:

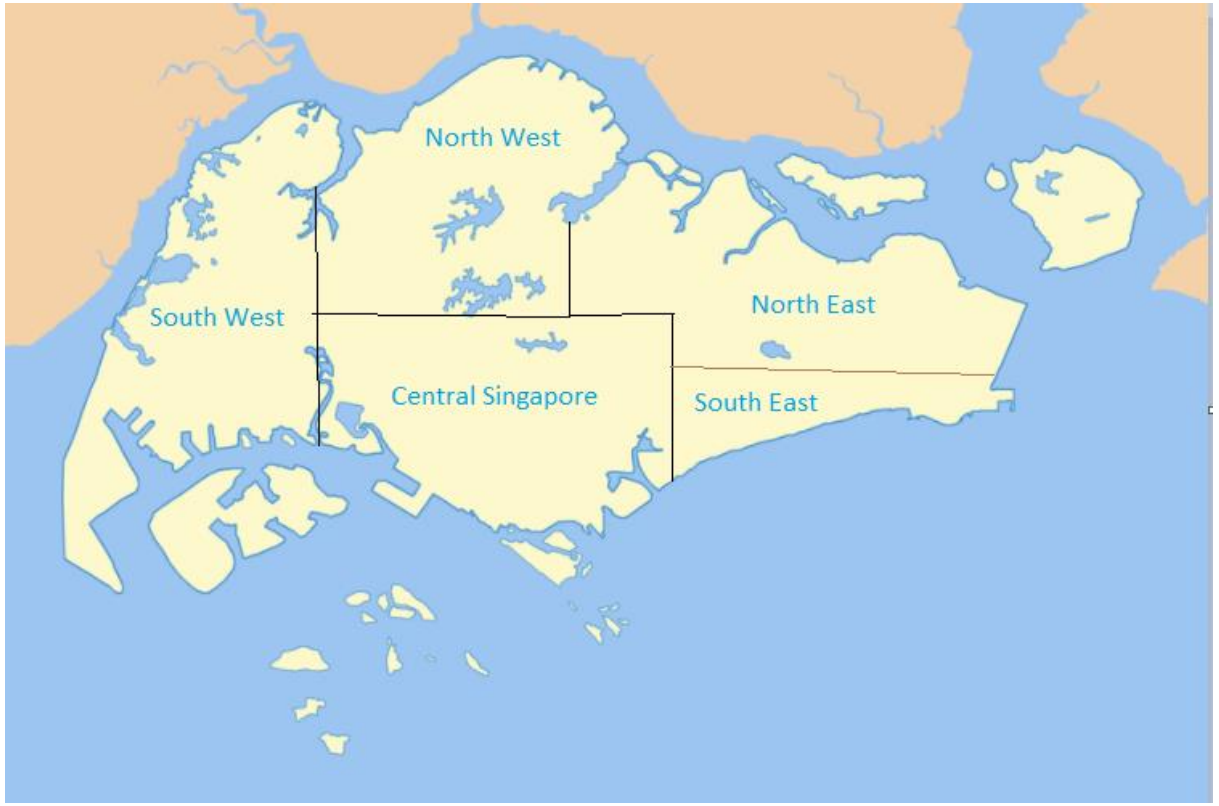
- Map coloring problem
- Cryptography Problem
- N Queens Problem
- TSP/VRP and many more

Typical search algorithm to solve CSP:

- **Backtracking** is a general algorithm for finding all (or some) solutions to some computational problems, notably constraint satisfaction problems, that incrementally builds candidates to the solutions, and abandons a candidate ("backtracks") as soon as it determines that the candidate cannot possibly be completed to a valid solution.

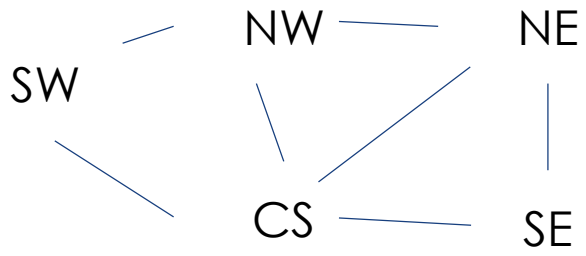
Singapore Map coloring problem

- Variables: SW , NW , CS , NE , SE
- Domains Di = {red , green, blue}
- Constraint: adjacent regions must have different color (SW!=NW)

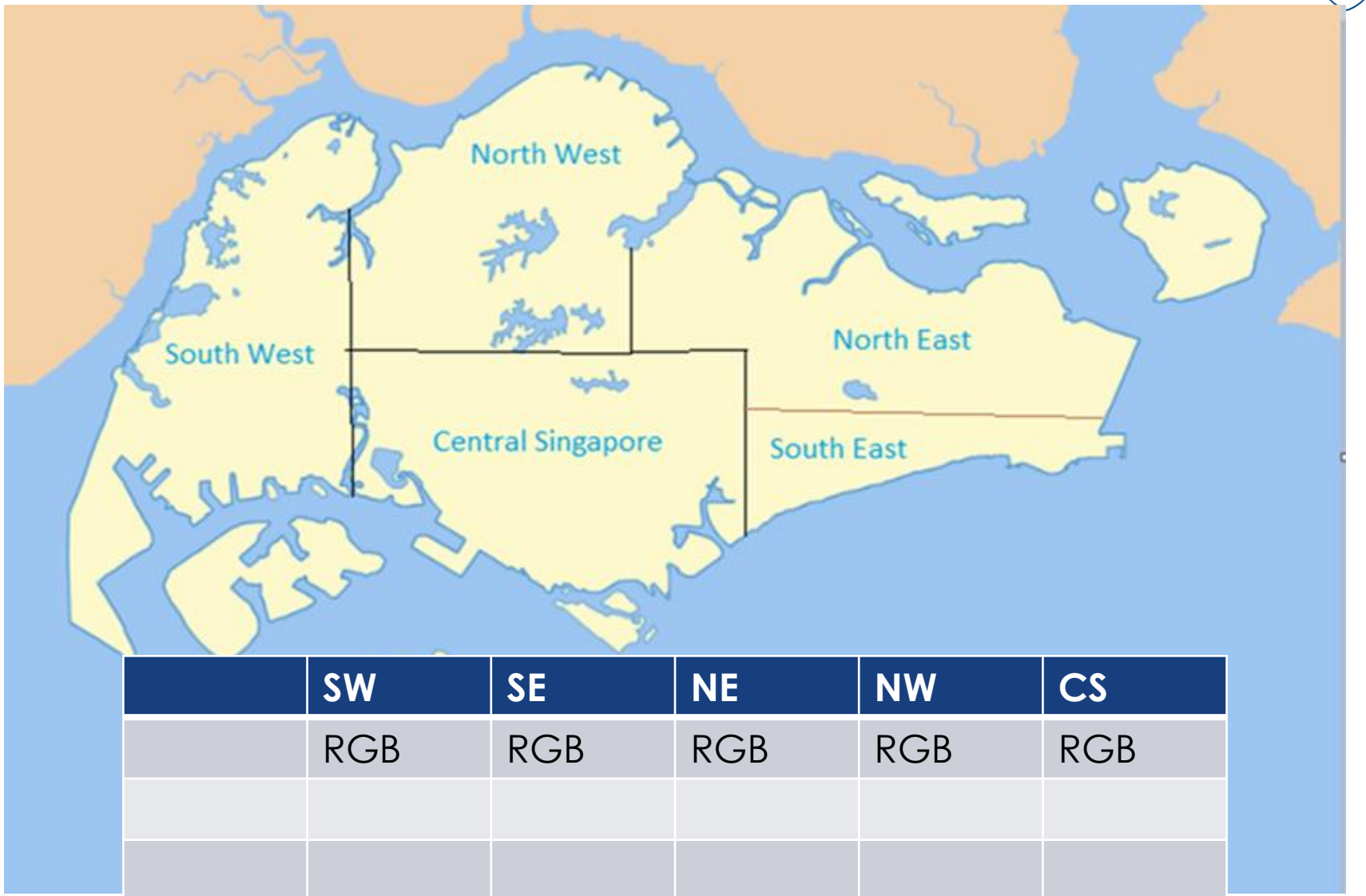


Constraint Graph

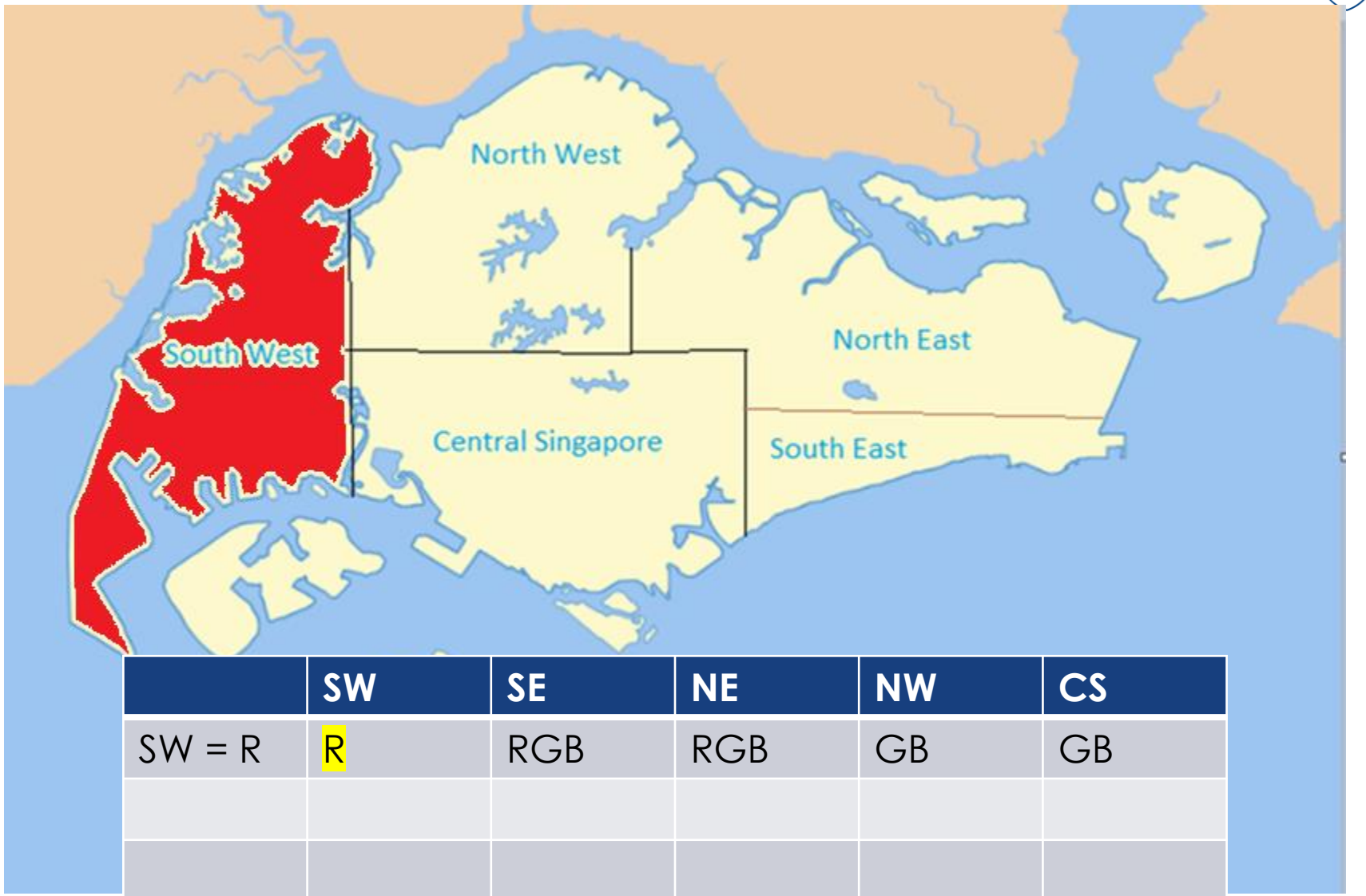
- Constraint Graph



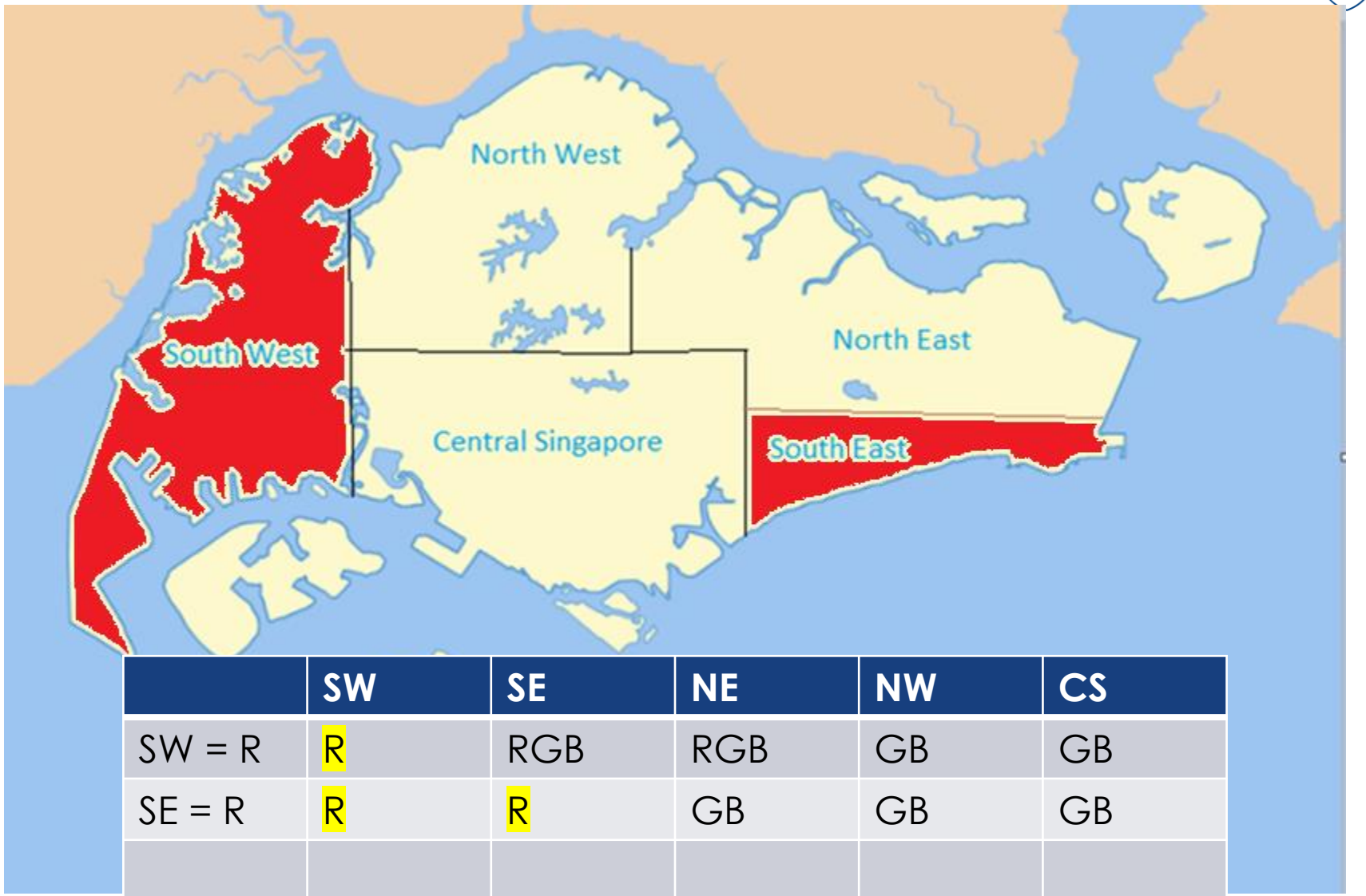
- Binary CSP: each constraint relates at most two variables
- Constraint Graph: nodes are available, arcs show constraints



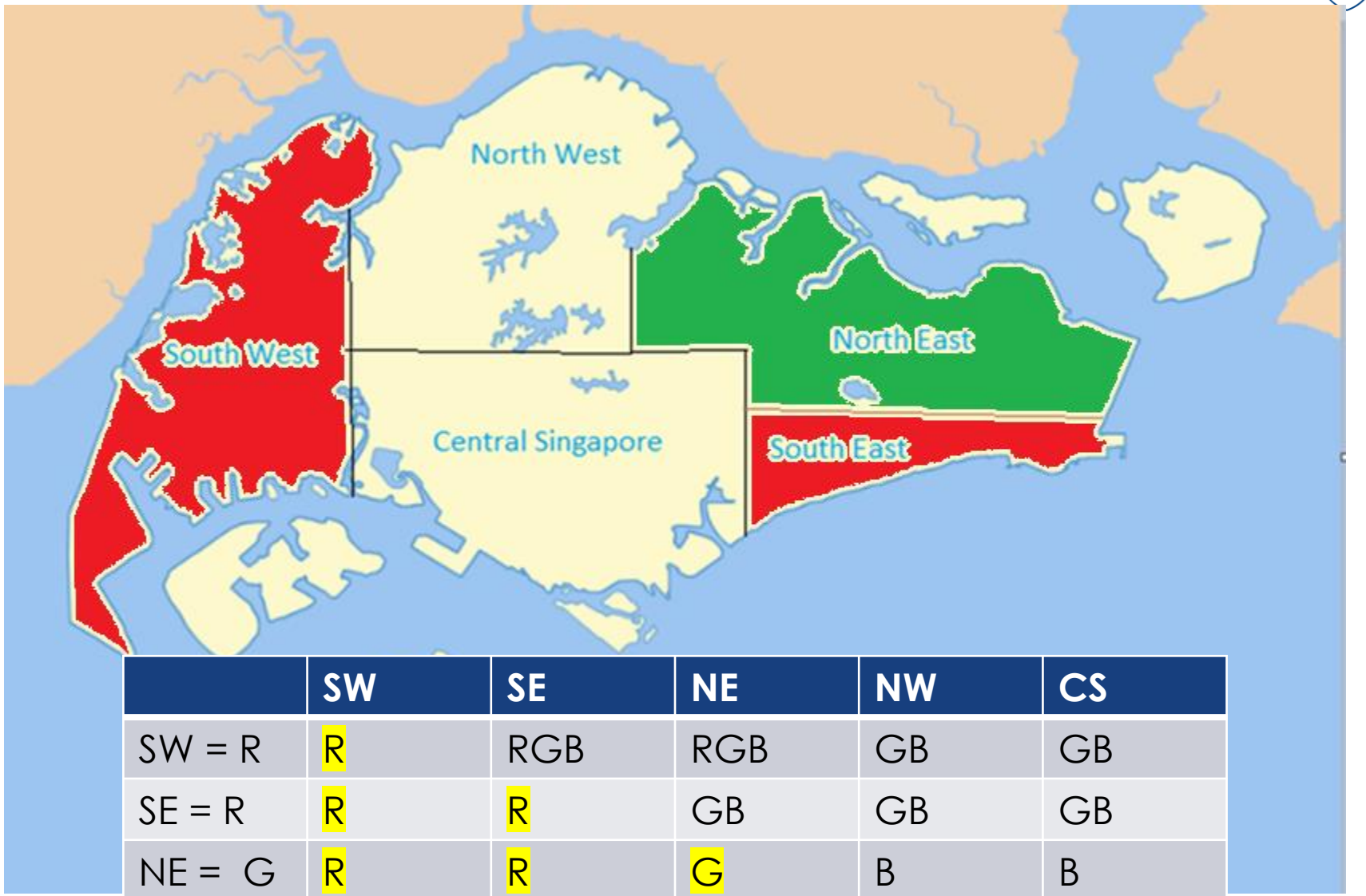
	SW	SE	NE	NW	CS
	RGB	RGB	RGB	RGB	RGB



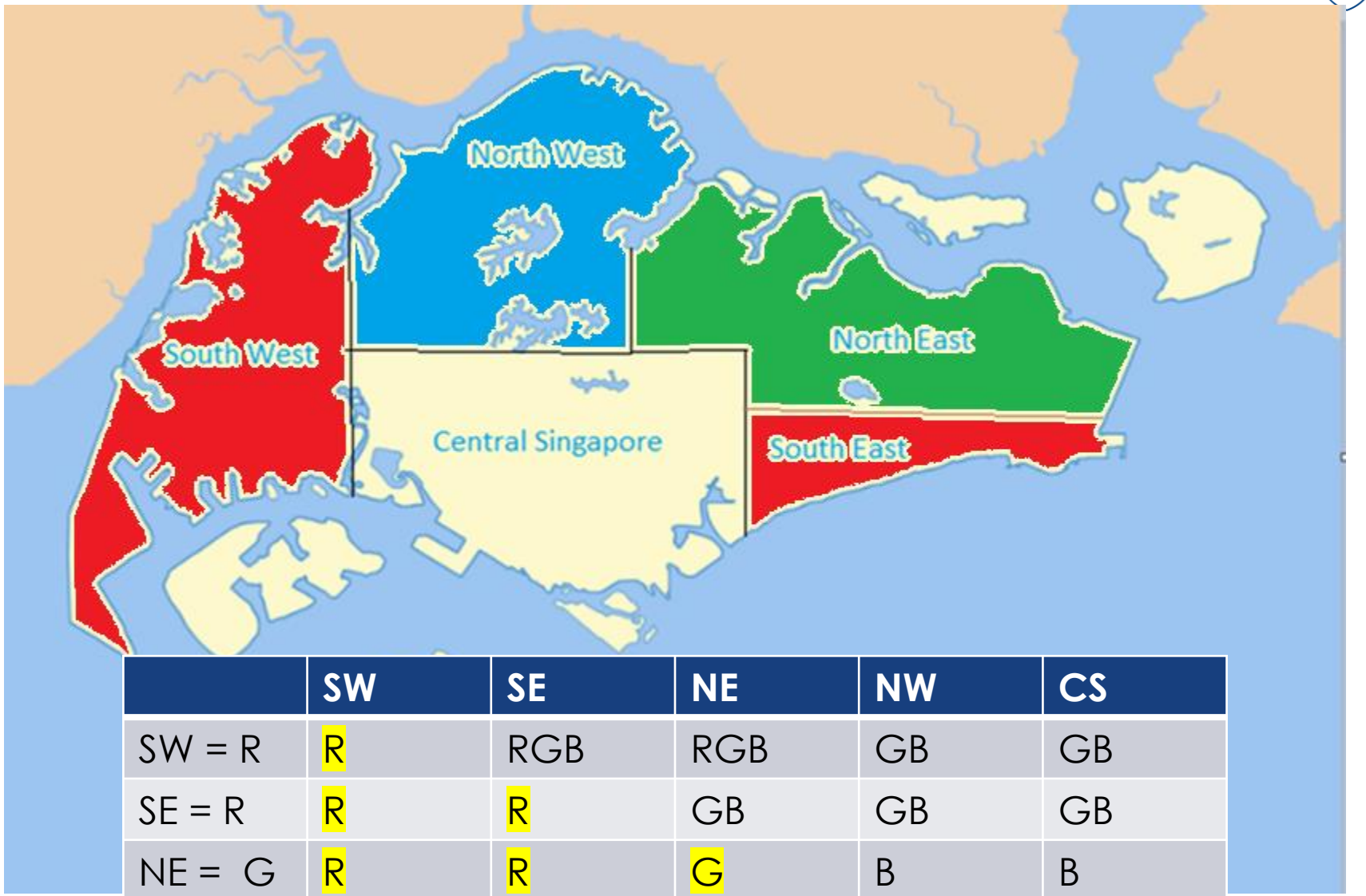
	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB



	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R	R	GB	GB	GB

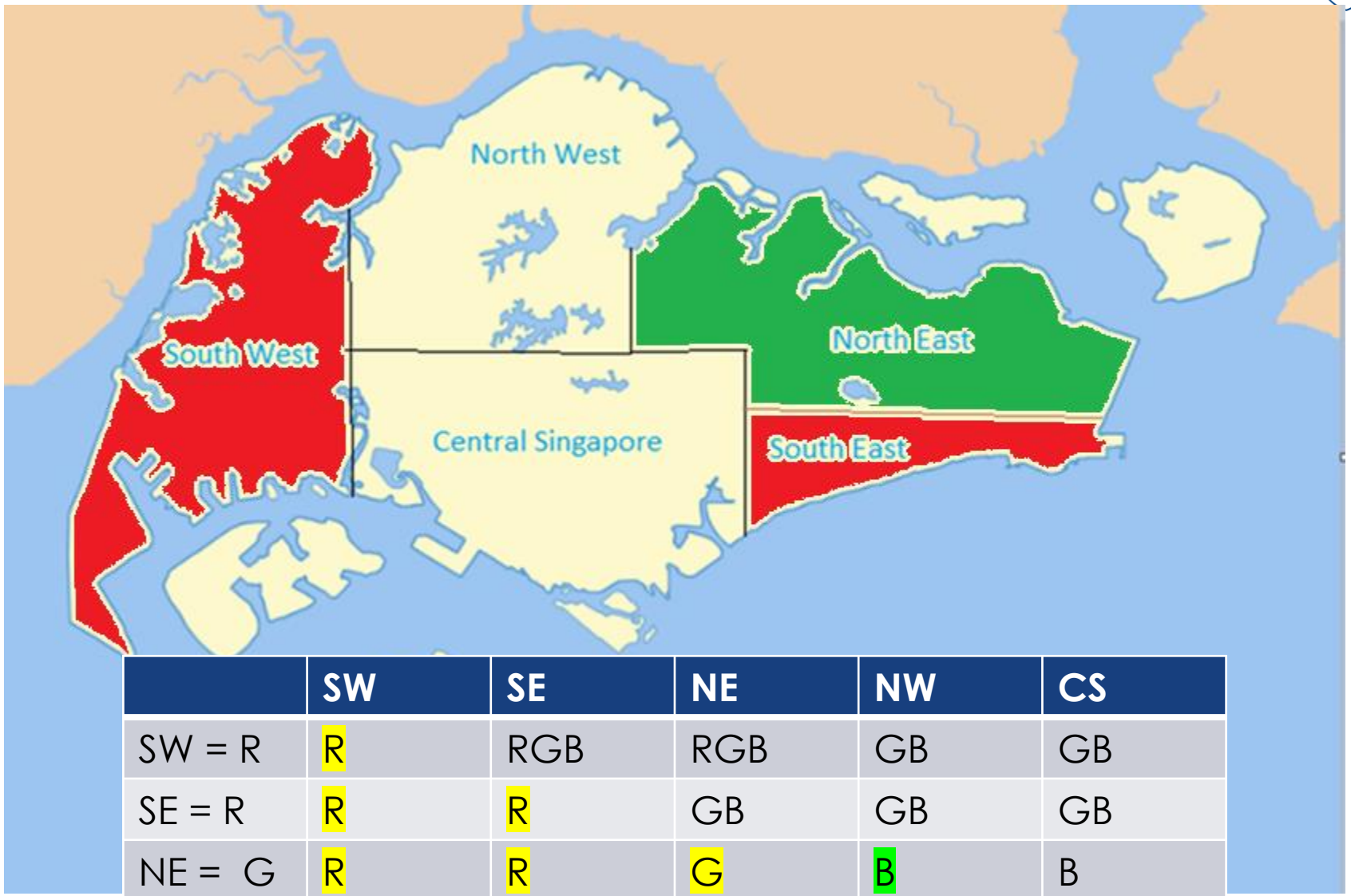


	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R	R	GB	GB	GB
NE = G	R	R	G	B	B



	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R	R	GB	GB	GB
NE = G	R	R	G	B	B
NW = B	R	R	G	B	✖

Backtrack

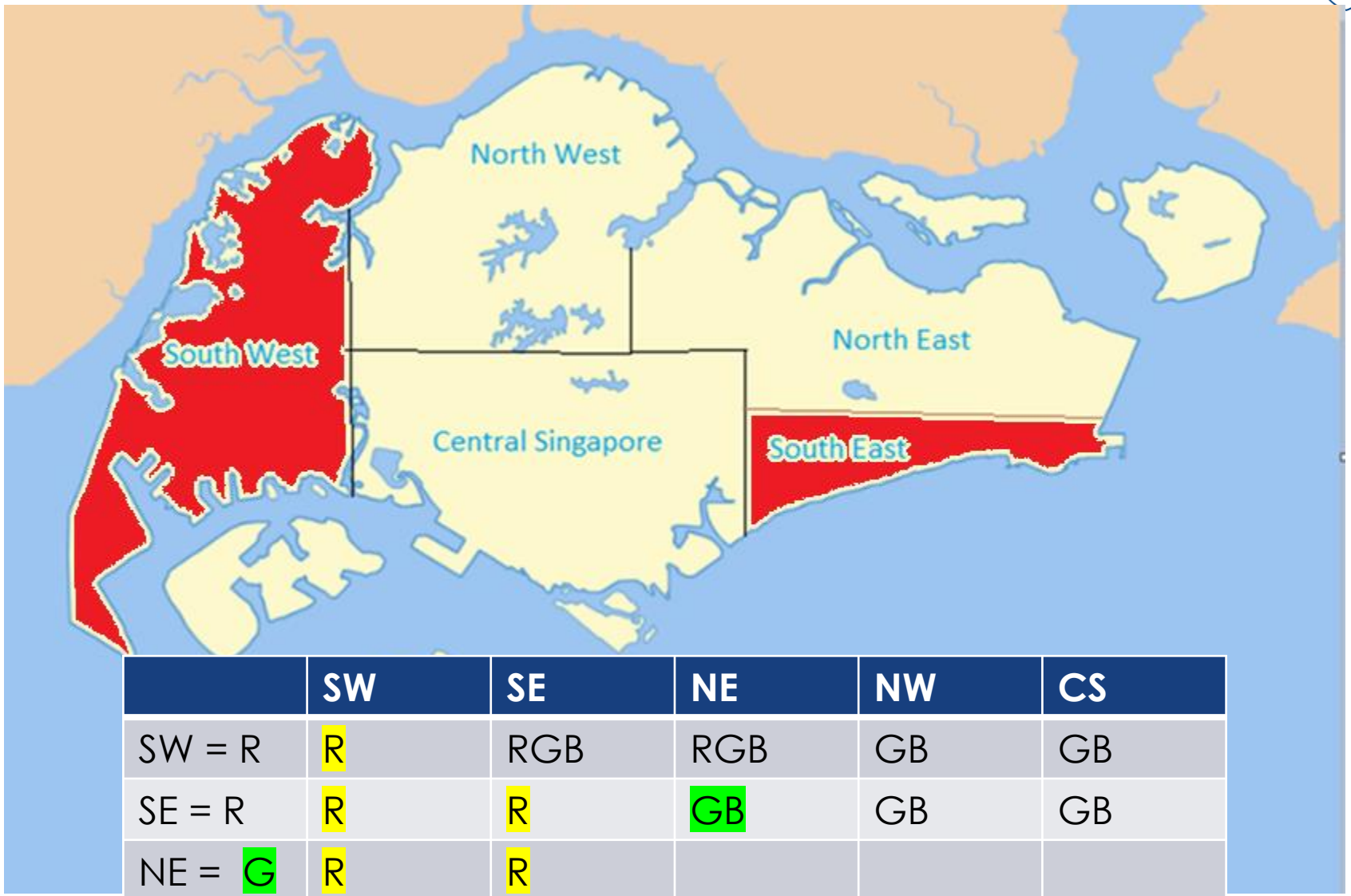


	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R	R	GB	GB	GB
NE = G	R	R	G	B	B
NW = B	R	R	G		

Backtrack

Any other possible values for ***NW***?

None; Then backtrack further to ***NE;***



	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R	R	GB	GB	GB
NE = G	R	R			

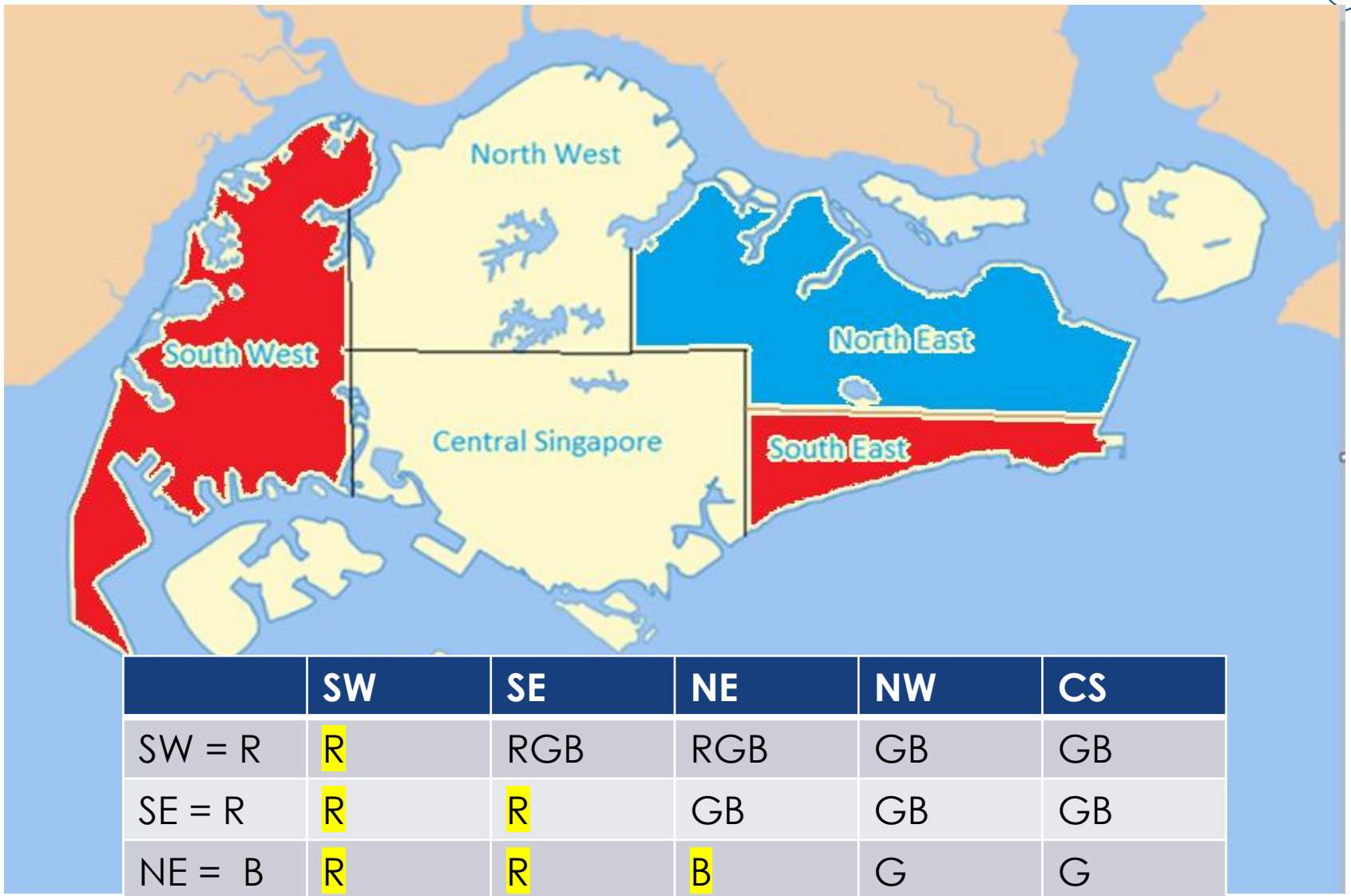
Backtrack

Any other possible values for ***NE***?

Yes: Blue;

Update:

NE = B to resume search;



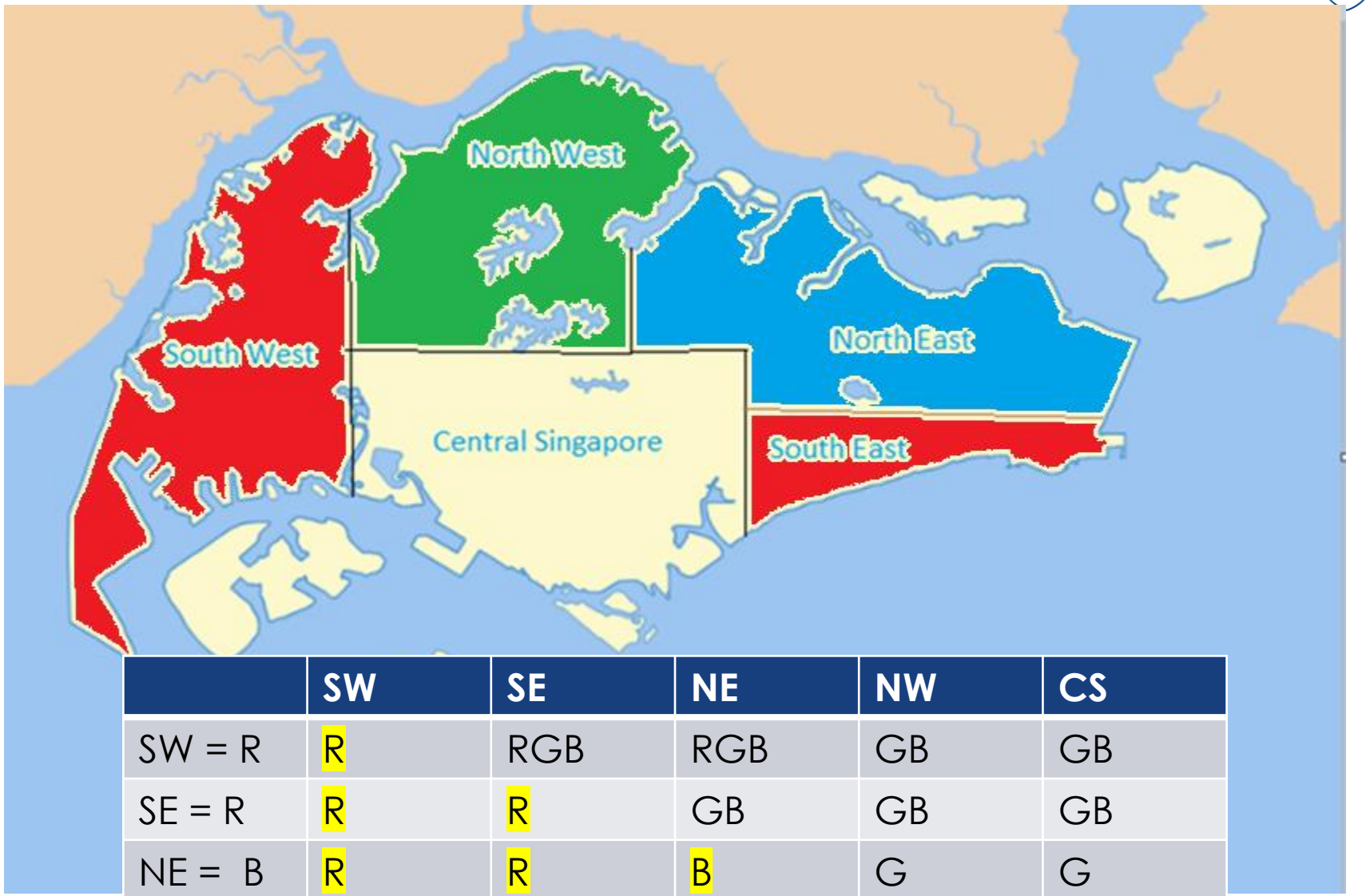
	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R	R	GB	GB	GB
NE = B	R	R	B	G	G

Any other possible values for ***NE***?

Yes: Blue;

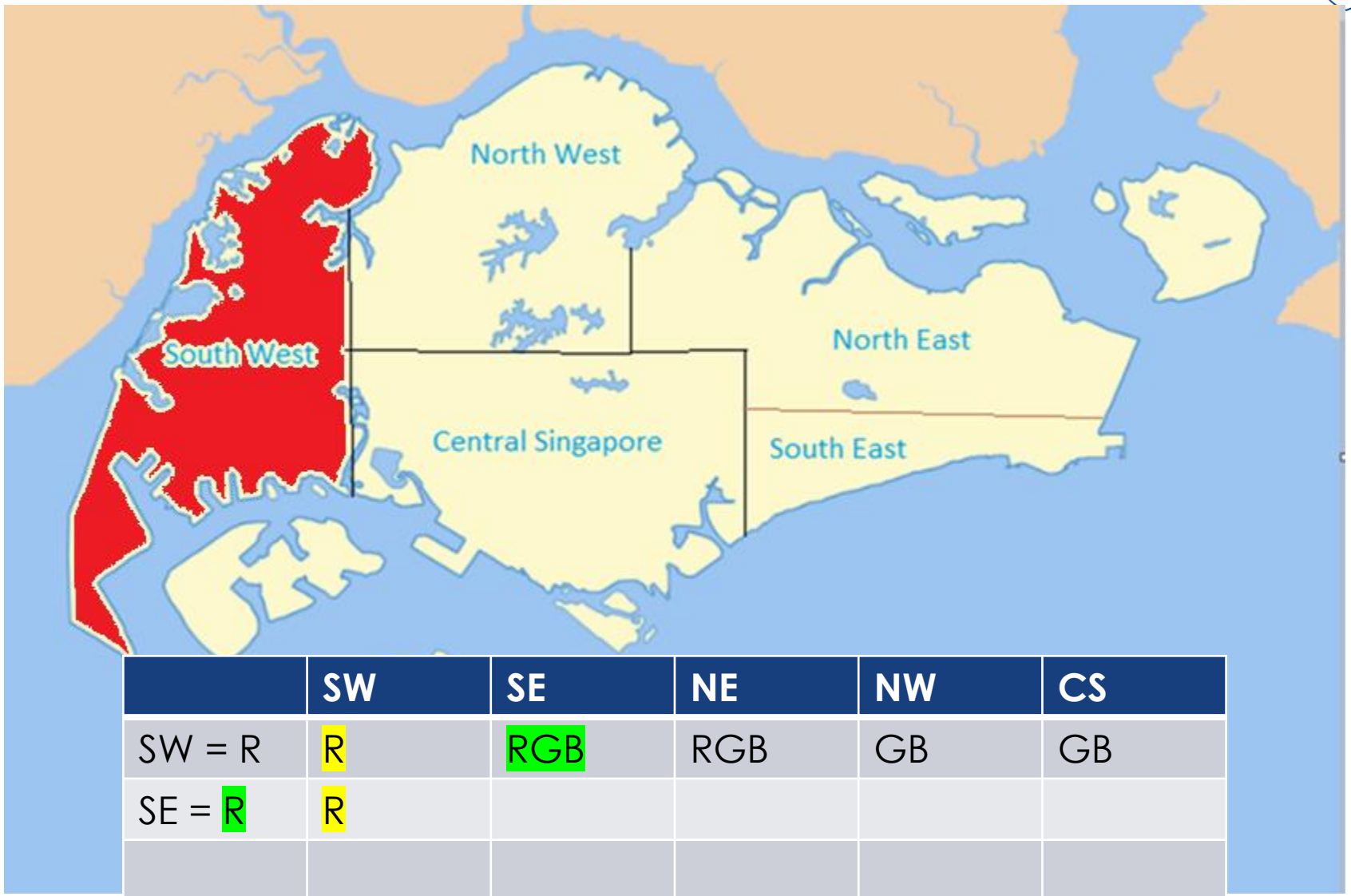
Update:

NE = B to resume search;



	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R	R	GB	GB	GB
NE = B	R	R	B	G	G
NW = G	R	R	B	G	✖

Backtrack



	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = R	R				

Backtrack

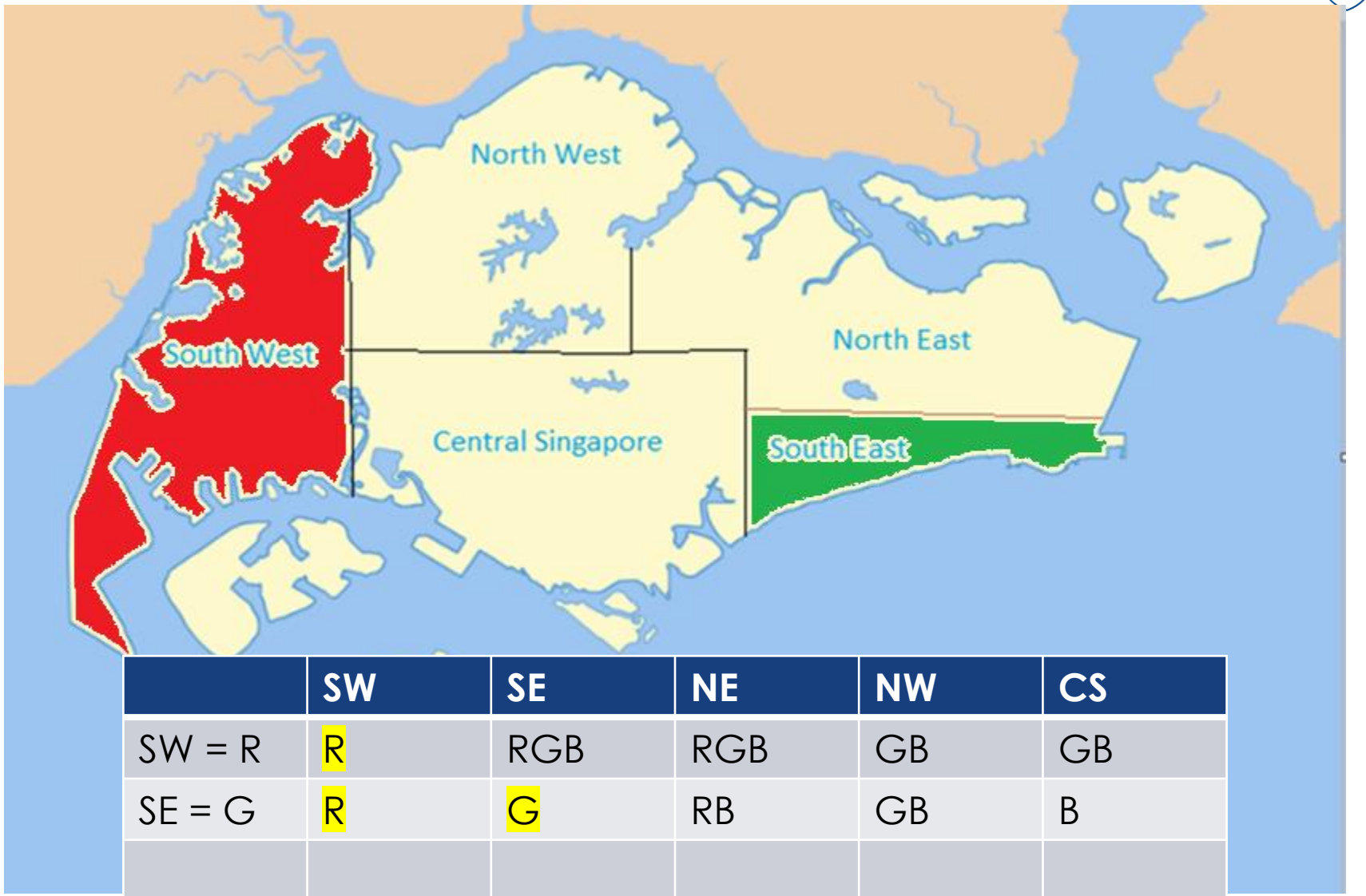
Since none works:

NE = G;

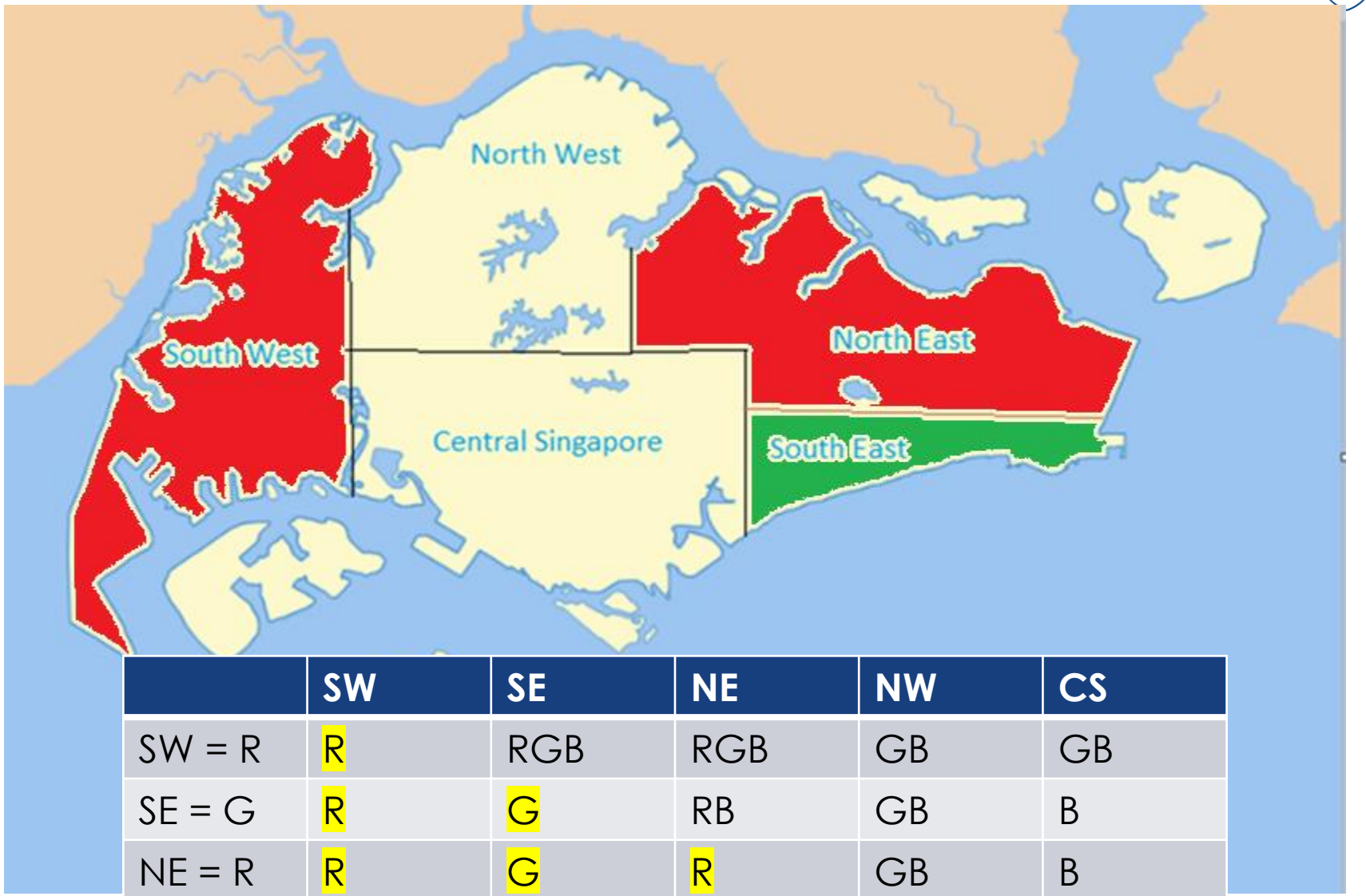
NE = B;

Then backtrack to ***SE***;

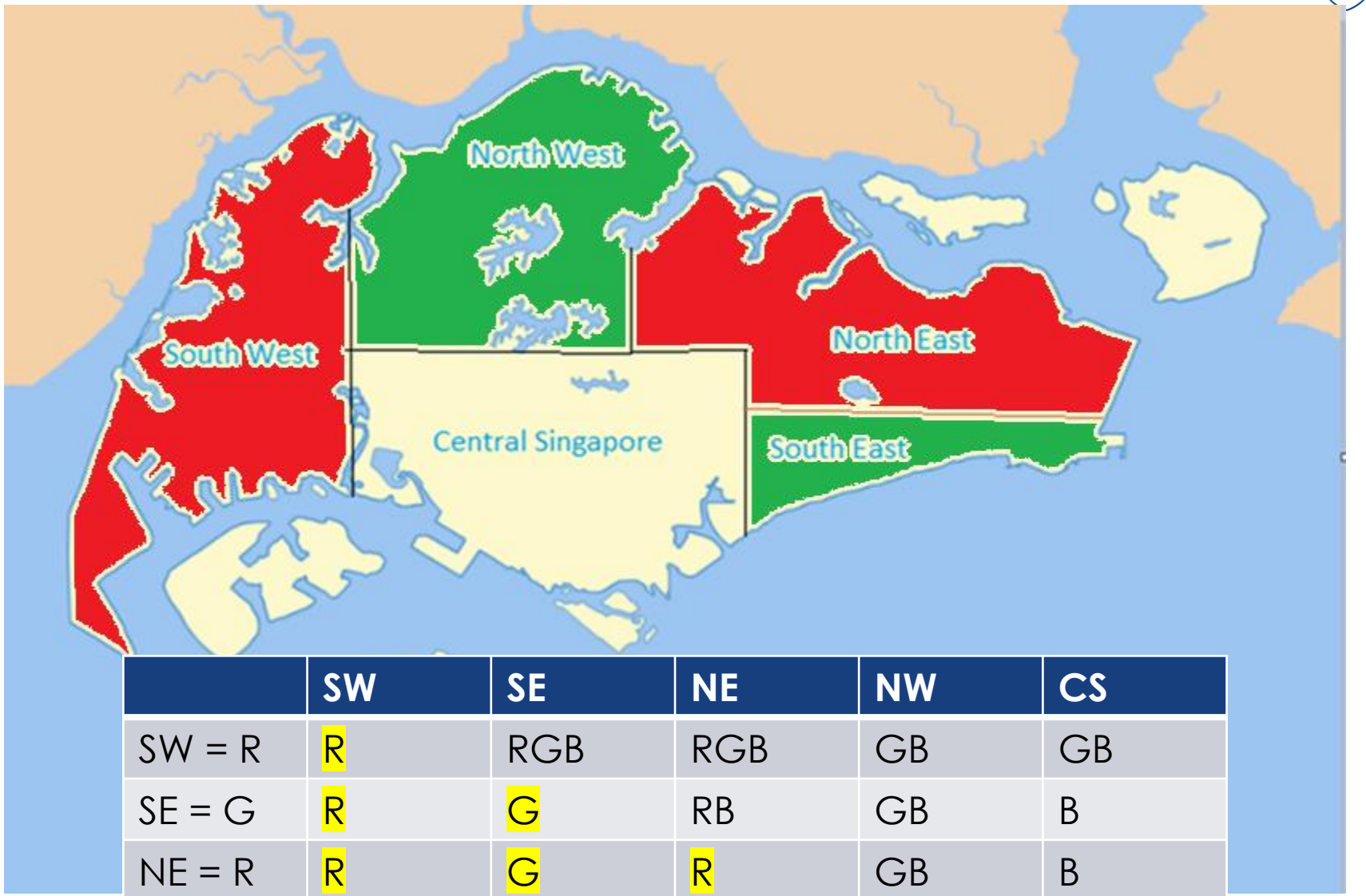
Update: ***SE*** = G



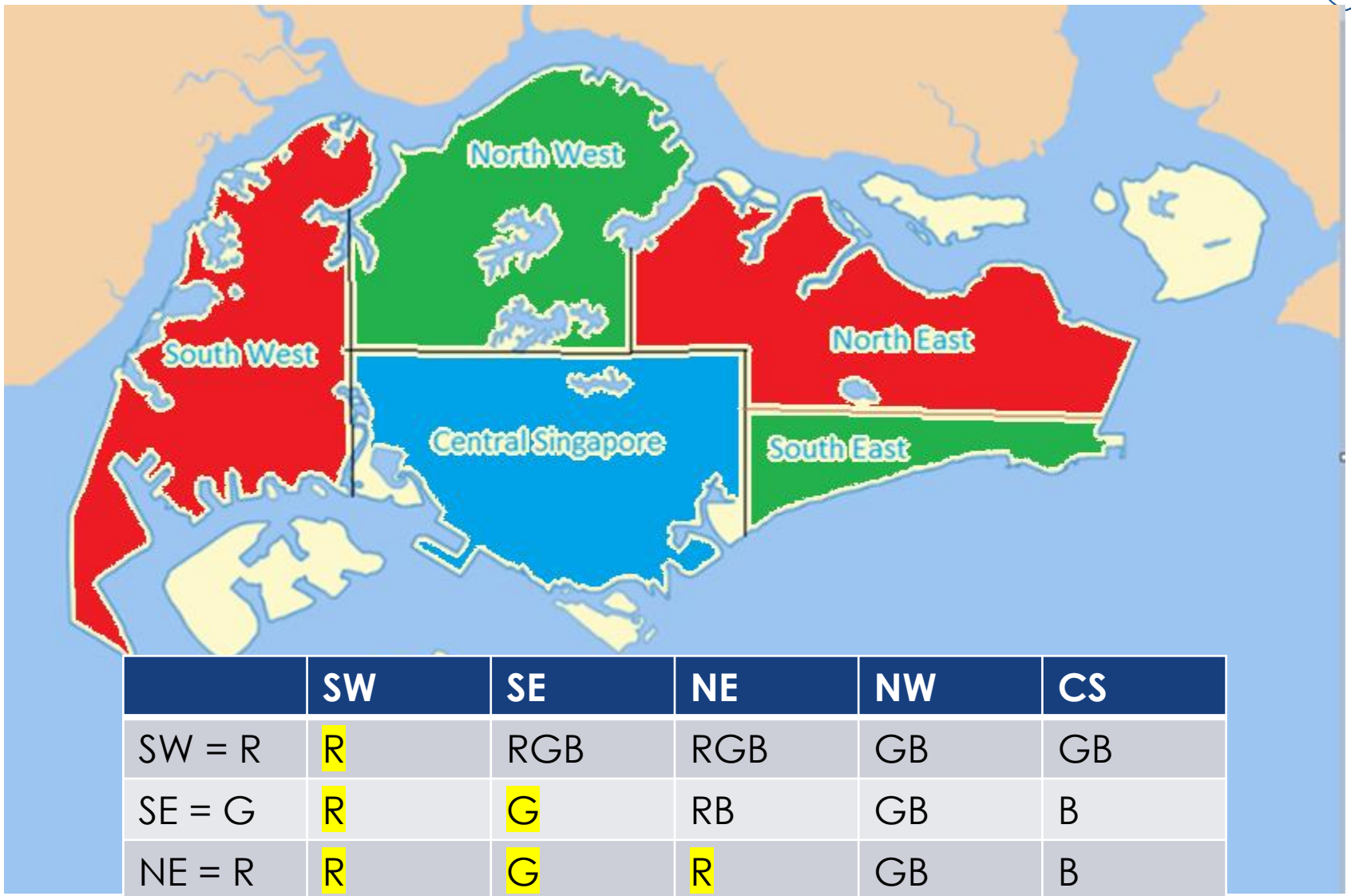
	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = G	R	G	RB	GB	B



	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = G	R	G	RB	GB	B
NE = R	R	G	R	GB	B



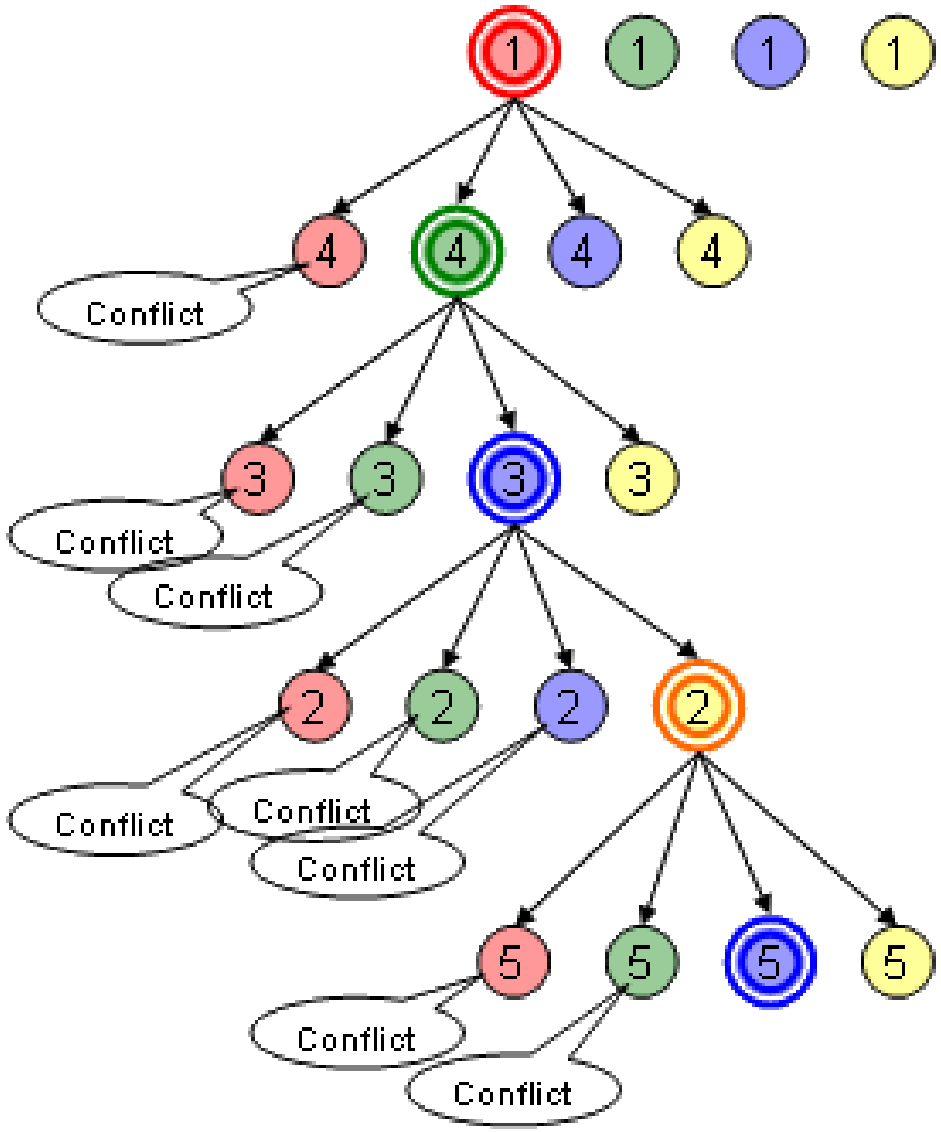
	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = G	R	G	RB	GB	B
NE = R	R	G	R	GB	B
NW = G	R	G	R	G	B

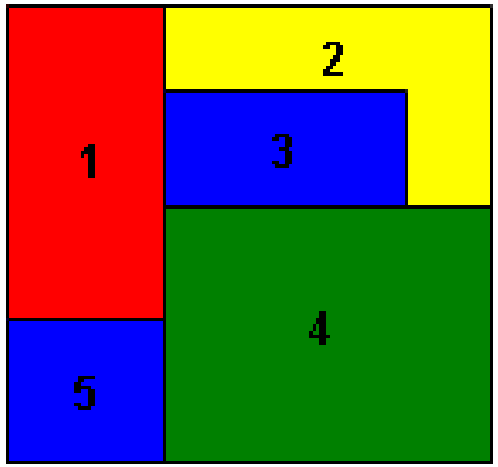


	SW	SE	NE	NW	CS
SW = R	R	RGB	RGB	GB	GB
SE = G	R	G	RB	GB	B
NE = R	R	G	R	GB	B
NW = G	R	G	R	G	B
CS = B	R	G	R	G	B

All constraints are satisfied;

Solution is found;





Cryptography Problem

- In the following pattern :
- replace each letter by a distinct digit so that the resulting sum is correct

```
  S E N D
  M O R E
=========
M O N E Y
```

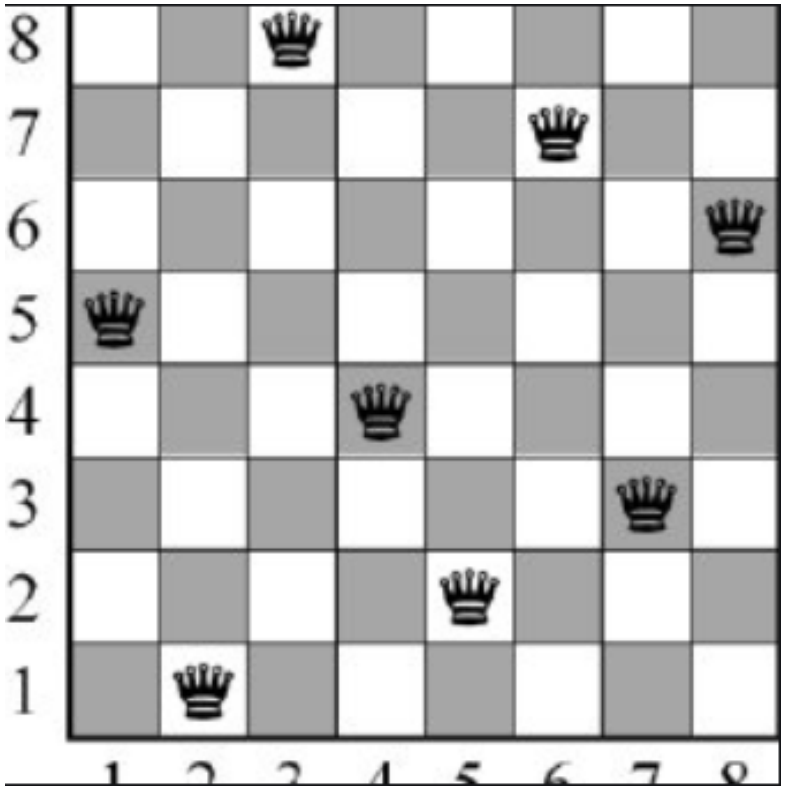
- What would be the Variables , Domains and Constraints in this case?

Sudoku Problem

5	3	1	2	7	6	8	9	4
6	2	4	1	9	5	2		
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

N queens Problem

- The local condition is that no two queens attack each other, i.e. are on the same row, or column, or diagonal.



N queens Problem (Using CSP Tools)

- Introduction to Google OR-Tools
- Tool installation
- CSP example: N-Queen workshop

Google OR-Tools: Introduction

- OR-Tools is an open source software suite for optimization, tuned for tackling the world's toughest problems in vehicle routing, flows, integer and linear programming, and constraint programming. (**O**perational **R**esearch domain)

OR-Tools

OR-Tools includes solvers for:

- Constraint Programming: A set of techniques for finding feasible solutions to a problem expressed as constraints.
- Linear and mixed integer programming: finding the optimal value of a linear objective function, given a set of linear inequalities as constraints (e.g., assigning people to jobs)
- Vehicle Routing: identifying best vehicle routes based on given constraints.
- Graph Algorithms: Code for finding shortest paths in graphs, min-cost flows, max flows, and linear sum assignments.

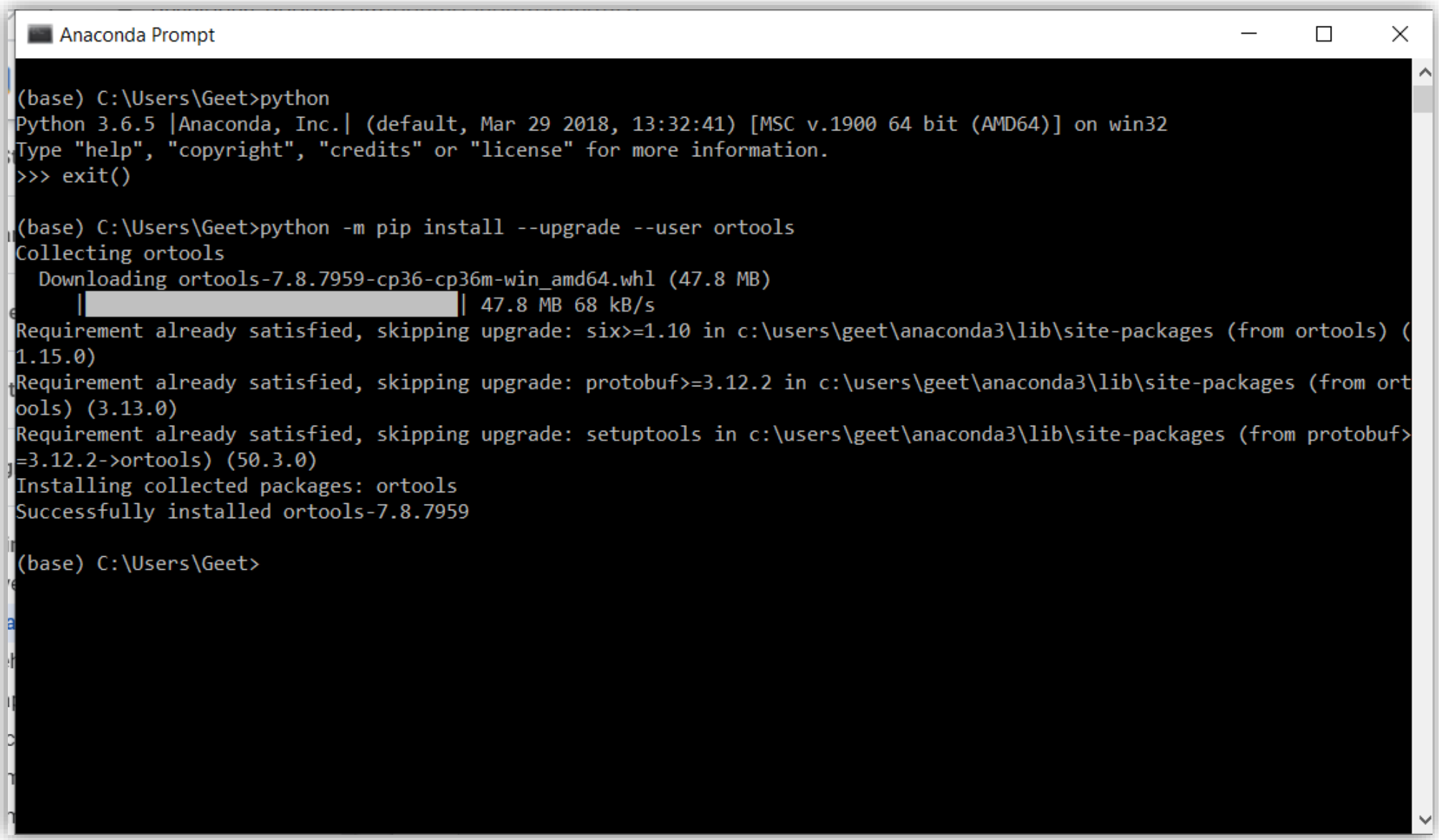
Google OR-Tools Installation

- Install OR-Tools for Python:
- The fastest way to get OR-Tools is to install the Python binary version. If you already have Python (version 3.5+ on Linux, or 3.6+ on Mac OS or Windows), and the Python package manager PIP, you can install OR-Tools as follows:

```
python -m pip install --upgrade --user ortools
```

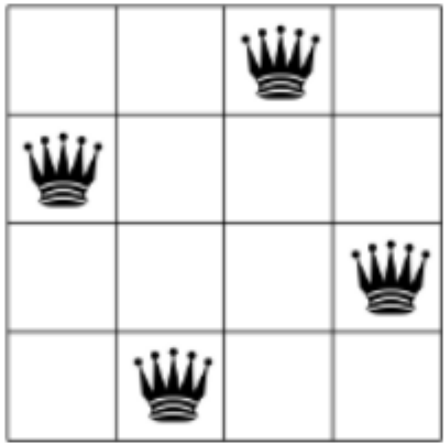
- For more details check out the installation guide to OR-Tools:
  https://developers.google.com/optimization/install
- Additionally , install Microsoft Visual C++ Redistributable for visual studio

Google OR-Tools Installation



N queens Problem

- How can N queens be placed on an NxN chessboard so that no two of them attack each other?
- For our example we'll start with 4 queens

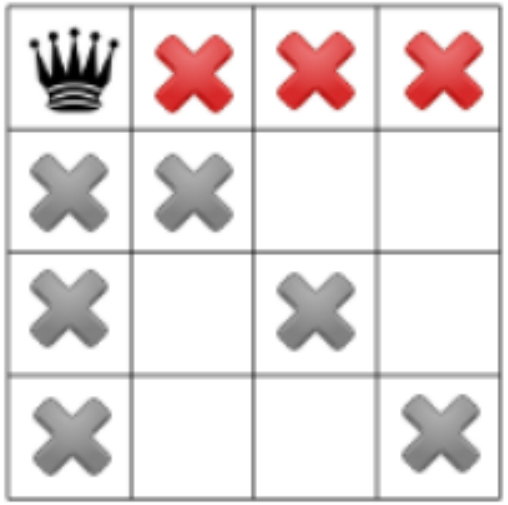


CSP on Google OR: solving N-Queens

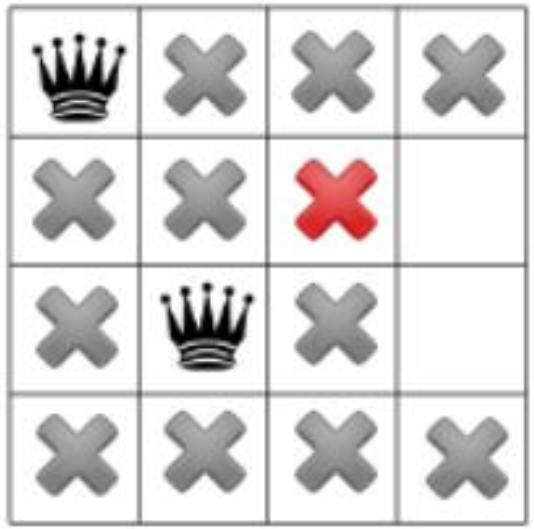
- Solving N Queens through Propagation and Backtracking :
- Propagation: Adding restrictions on each step :
  - E.g. Each time the solver places a queen, it can't place any other queens on the row and diagonals the current queen is on.
- Backtracking: When the solver gets stuck and cat assign a value to the next variable.
  - E.g. moving a queen to a new square on the current column.

Constraints:

- Only one queen in the same row
- Only one queen in the same column
- No other queen in the diagonal

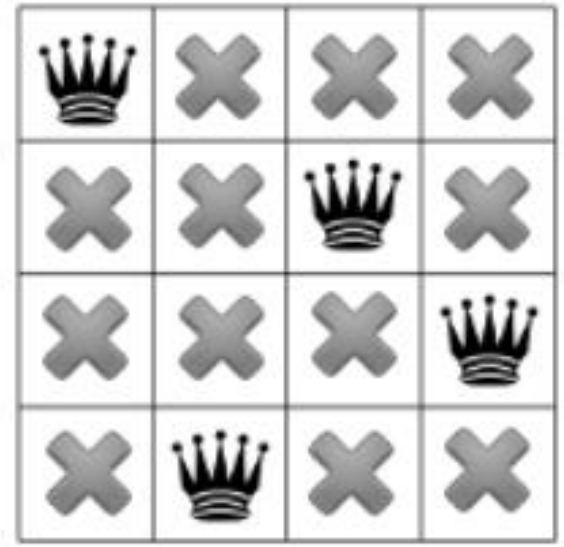


- After propagating the diagonal constraint, we can see that it leaves no available squares in either the third column or last row:



- Now try to Backtrack and solve

- Finally The solution comes up as follows :



Solving Using CP Solver

- Declare the Solver

- from ortools.constraint_solver import pywrapcp
- def main(board_size):
  # Creates the solver.
- solver = pywrapcp.Solver("n-queens")

- Create the variables
- IntVar Creates Variables for the problem as an array neamed queens

```
queens = [solver.IntVar(0, board_size - 1, "x%i" % i) for i in
range(board_size)]
```

- Create the Constraints :
- All rows,columns must be different.
- No two queens can be on the same diagnol

```
solver.Add(solver.AllDifferent(queens))
solver.Add(solver.AllDifferent([queens[i] + i for i in range(board_size)]))
solver.Add(solver.AllDifferent([queens[i] - i for i in range(board_size)]))
```

- Add decision buider

```
db = solver.Phase(queens,
                  solver.CHOOSE_FIRST_UNBOUND,
                  solver.ASSIGN_MIN_VALUE)
```

- Set the board value = n (4, 8, 64, 128, etc.)

Workshop

- Try implementing the N queens code.
- What results are obtained when n = 64?
- Display the results
- What results are obtained when n = 128?
- Compare results. What is the exponential growth?
- Is there a large difference? Why?
- Export your python code or ipynb file, and other relevant artifacts, then zip and submit to NUS (Canvas) under proper naming convention ***StudentID_FullName RS2WS2.3.1.zip***;

2.3 Search Reasoning [Workshop]

2.3.1 Informed Search

Thanks to Geet Jethwani (A0215395B) for his contribution.

2.3.2 Reasoning System Exploration

2.3.3 Workshop Submission

Reasoning System Exploration

- **[No workshop submission] Explore & run one or more past learner project of your choice from below:**
  1. https://github.com/IRS-PM/IRS-PM-2022-01-29-IS04FT-GRP8-OhMyFish
  2. https://github.com/IRS-PM/IRS-PM-2021-07-05-IS03FT-Group-17-CoVid-Diagnosis-and-Symptoms-Helper
  3. https://github.com/IRS-PM/IRS-PM-2021-07-05-IS03FT-Group-8-SnapYummy-Cooking-Assistant
  4. https://github.com/IRS-PM/IRS-PM-2021-01-16-IS03PT-GRP-Invigilator-Allocation-System
  5. https://github.com/IRS-PM/IRS-PM-2020-01-18-IS02PT-GRP2-Enterprise-Knowledge-Graph
  6. https://github.com/IRS-RS/IRS-MRS-2019-07-01-IS01FT-GRP7-Trippingo-SG-Travel-Planner
  7. https://github.com/IRS-RS/IRS-RS-2019-03-09-IS01PT-GRP-Hex-Sdot_Scheduling_and_Dispatch_Optimization_Tool
  8. https://github.com/IRS-PM/Workshop-Project-Submission-Template-GA-Optimizer
- **[Optional action for more able learners] Create *PULL* request to update the github repo if you fixed any compatibility issue during installation using latest packages.**

2.3 Search Reasoning [Workshop]

2.3.1 Informed Search

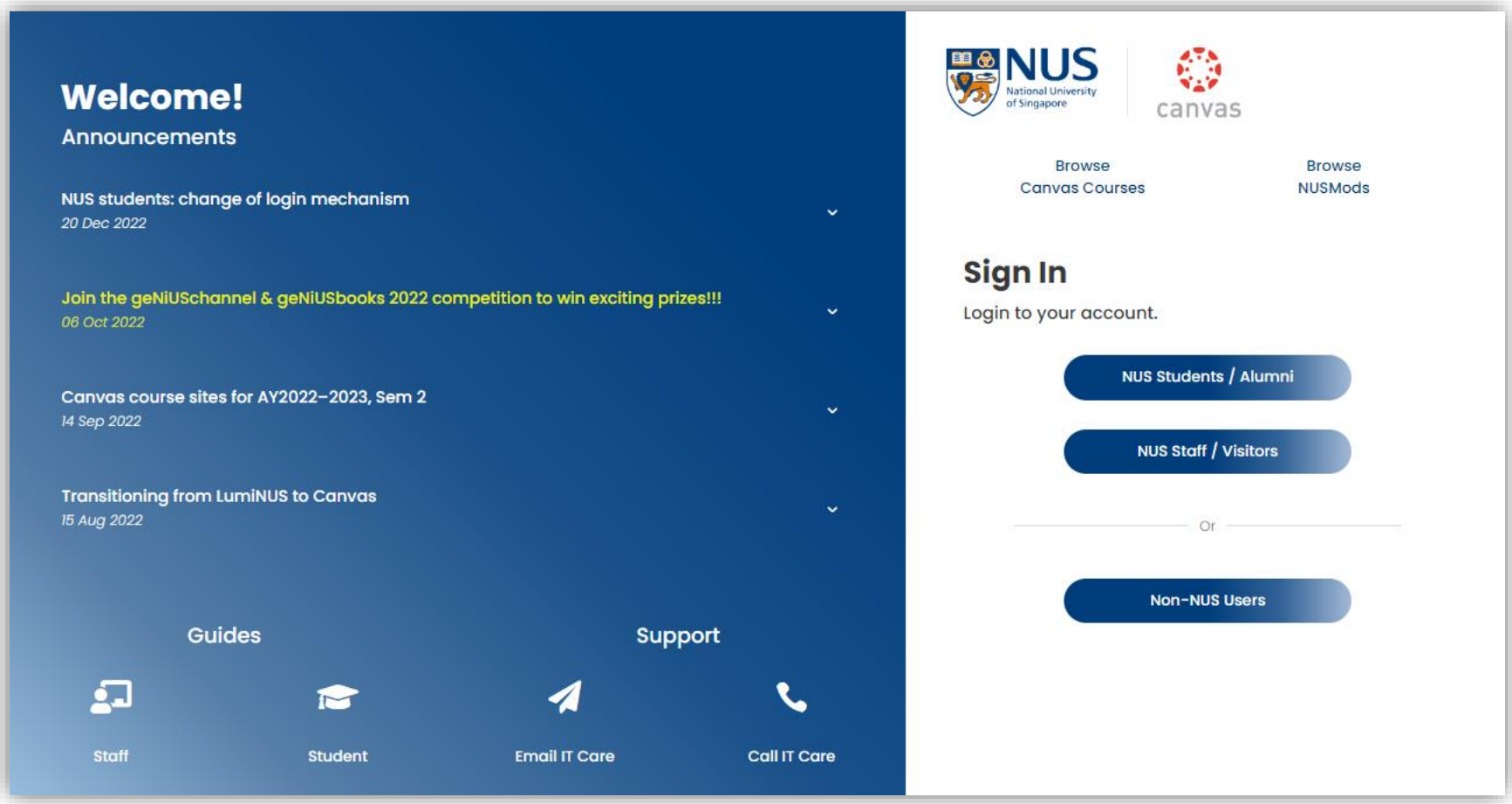
Thanks to Geet Jethwani (A0215395B) for his contribution.

2.3.2 Reasoning System Exploration

2.3.3 Workshop Submission

Workshop Submission

- **Naming convention: StudentID YourFullName**
- **Use zip to a single file, then rename, if you plan to submit multiple files.**



END OF NOTES

APPENDICES

The right A.I. for the job

One Artificial Intelligence algorithm does not fit all use cases.

Vector Space Model

Full text search

"cat"

The secret life of felines
felines.pdf
Felines, or **cats** as they are more commonly known, are carnivorous ...

Other use cases include:
recommendations, similarties, ...

Implemented by:

Lucene

Neural Net

Image recognition

"Dog"

Other use cases include:
voice recognition, machine translation, ...

Implemented by:
TensorFlow, Deeplearning4j

Constraint Solver

Vehicle routing problem

15% less driving time

Other use cases include:
employee rostering, job scheduling, ...

Implemented by:
OptaPlanner

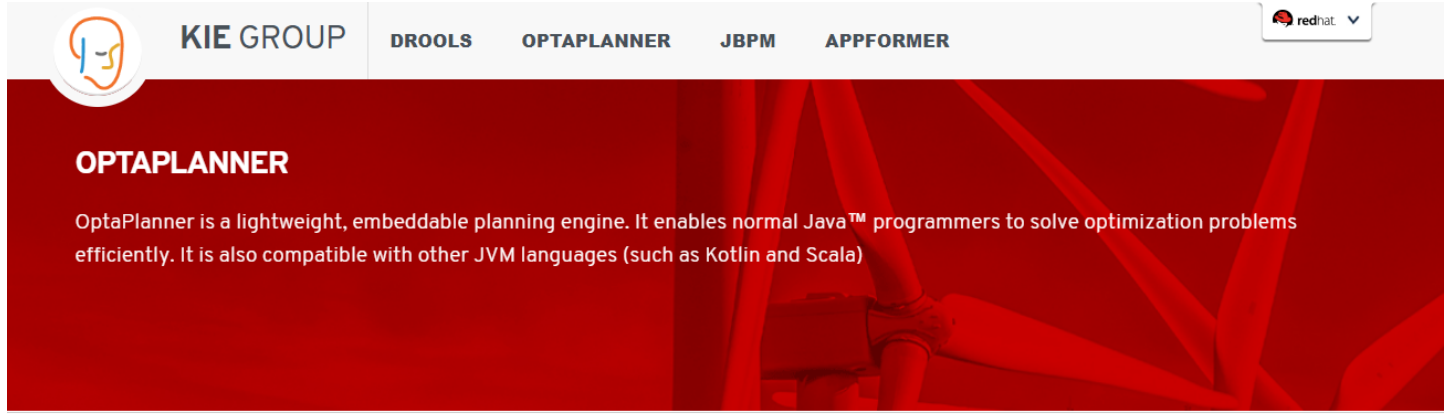
Other algorithms for other use cases:
A* Search for pathfinding, Rete/Phreak for production rule systems, k-means for cluster analysis, ...

Source http://www.optaplanner.org/learn/slides.html

[Optional] Search Reasoning [Workshop]

[Optional] Search Reasoning [Workshop]

KIE OptaPlanner Tutorial



DROOLS

Drools is a business rule management system with a forward-chaining and backward-chaining inference based rules engine, allowing fast and reliable evaluation of business rules and complex event processing.

Read more →

OPTAPLANNER

OptaPlanner is a constraint solver that optimizes use cases such as employee rostering, vehicle routing, task assignment and cloud optimization.

Read more →

JBPM

jBPM is a flexible Business Process Management suite allowing you to model your business goals by describing the steps that need to be executed to achieve those goals.

Read more →

APPFORMER

AppFormer is a low code platform to develop modern applications. It's a powerful tool for developers that can easily build applications by mashing up components and connect them to other Red Hat modules and software.

We make building apps looks easy.

Read more →

JBoss KIE

http://www.kiegroup.org/

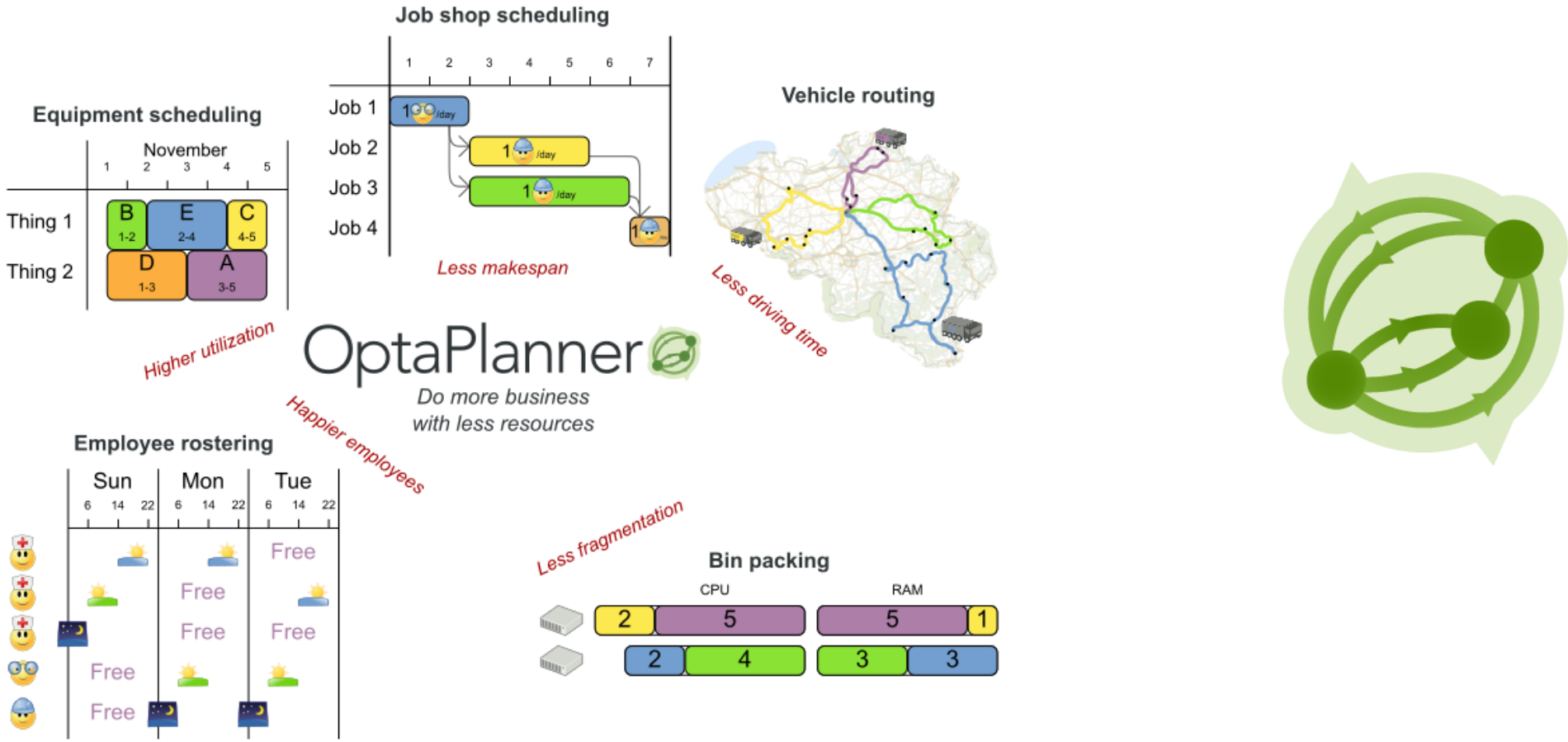
JBoss KIE OptaPlanner

http://www.optaplanner.org/

[Optional] Search Reasoning [Workshop]

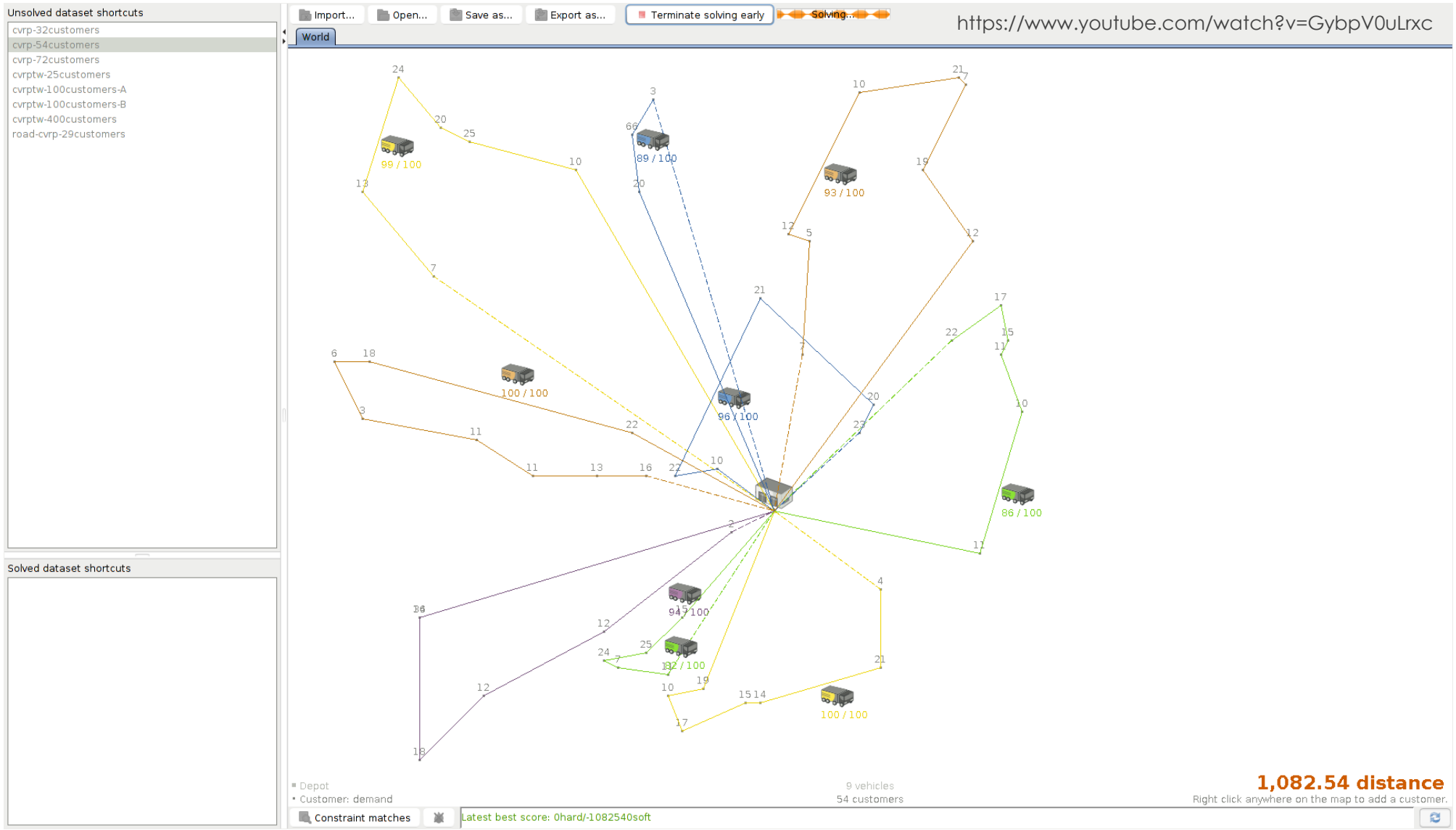
KIE OptaPlanner Tutorial

- **Constrain Satisfaction: Business Resource Optimizer**



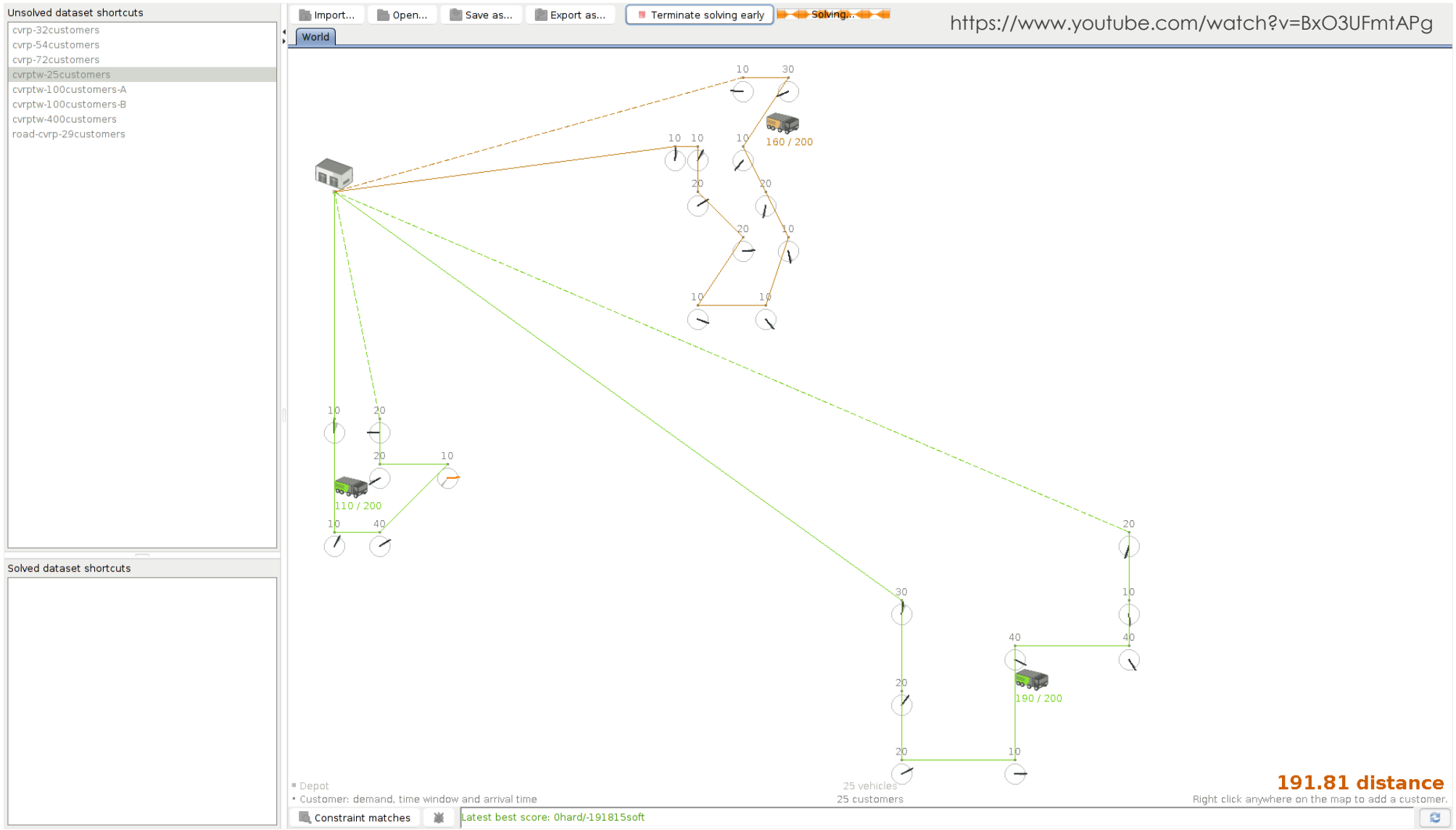
[Optional] Search Reasoning [Workshop]

KIE OptaPlanner Tutorial – VRP: Customer demand (vehicle load)



[Optional] Search Reasoning [Workshop]

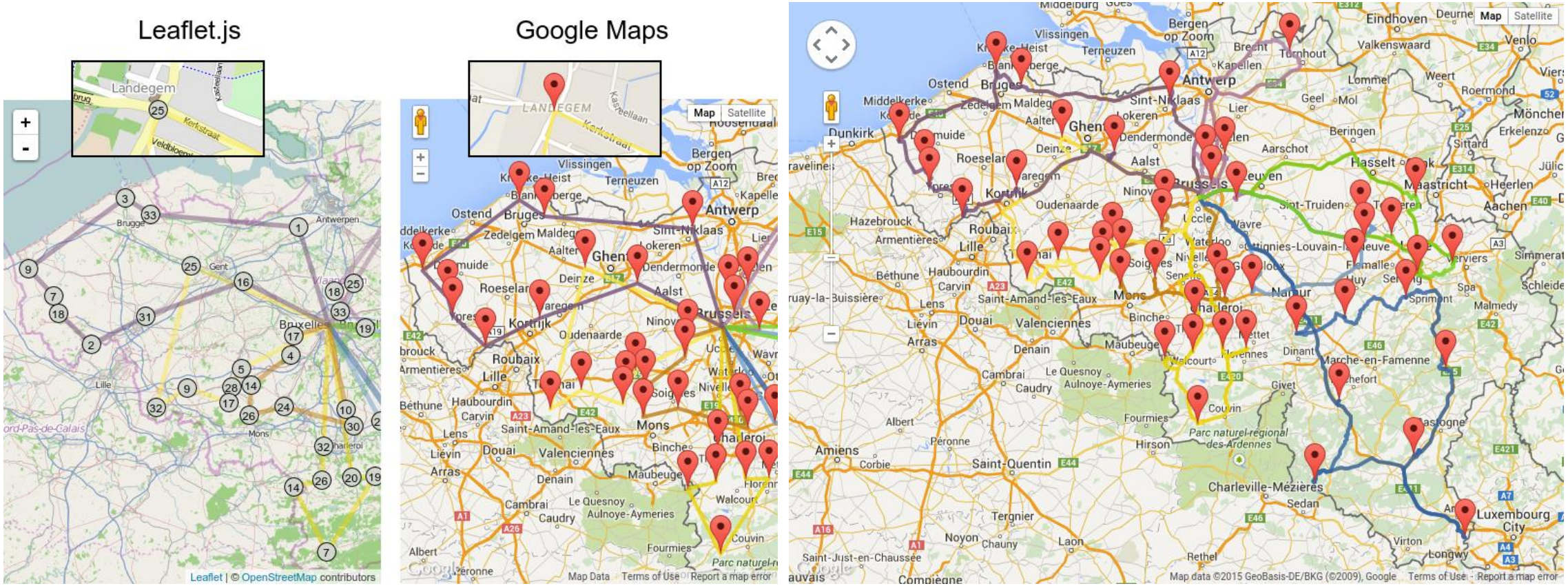
KIE OptaPlanner Tutorial – VRP: Customer demand, Time window



[Optional] Search Reasoning [Workshop]

KIE OptaPlanner Tutorial – VRP with map integration

Visualizing Vehicle Routing with Leaflet and Google Maps



https://www.optaplanner.org/blog/2015/03/10/VisualizingVehicleRoutingWithLeafletAndGoogleMaps.html

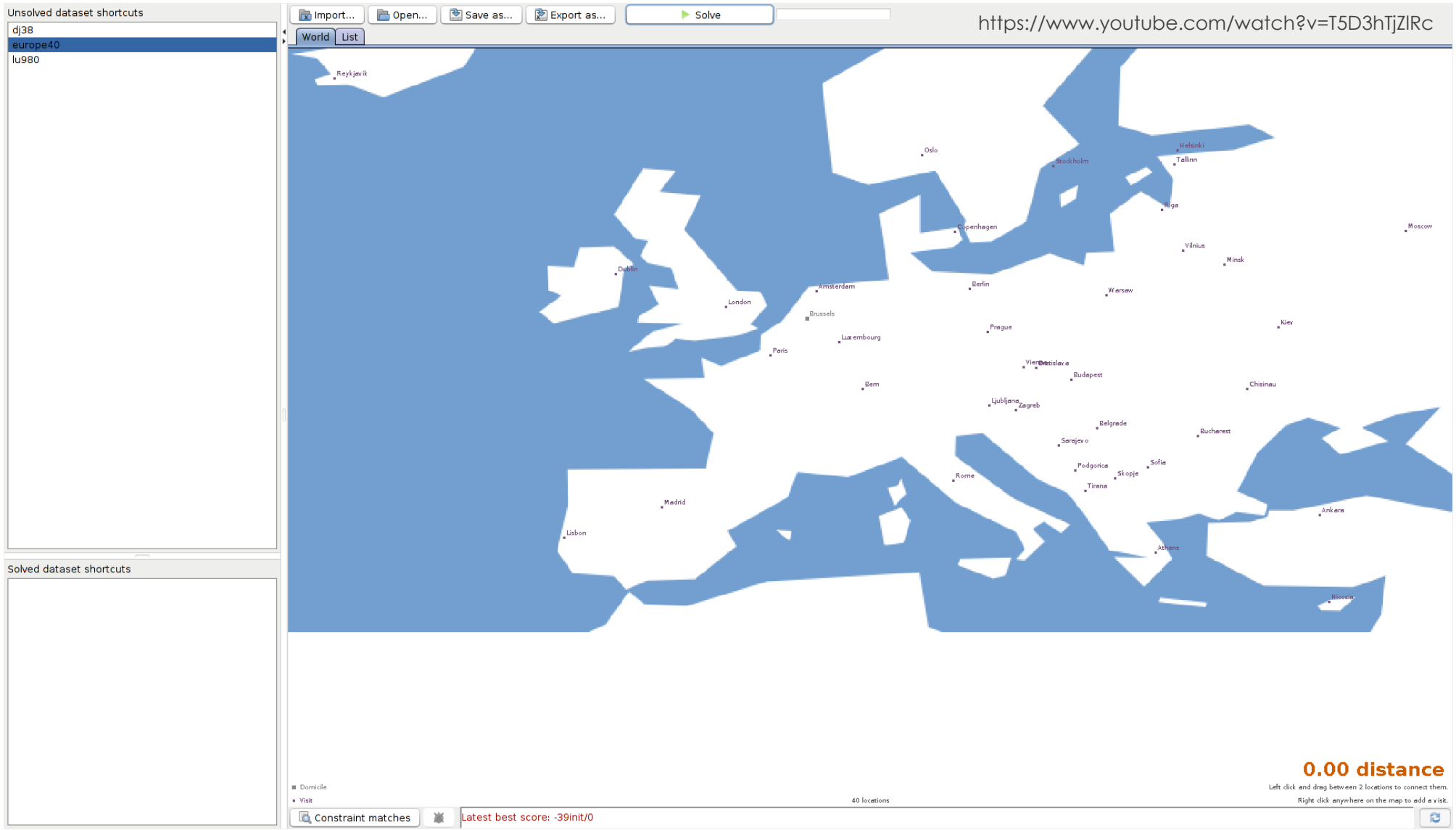
[Optional] Search Reasoning [Workshop]

KIE OptaPlanner Tutorial – TSP: Europe cities

https://www.youtube.com/watch?v=T5D3hTjZlRc

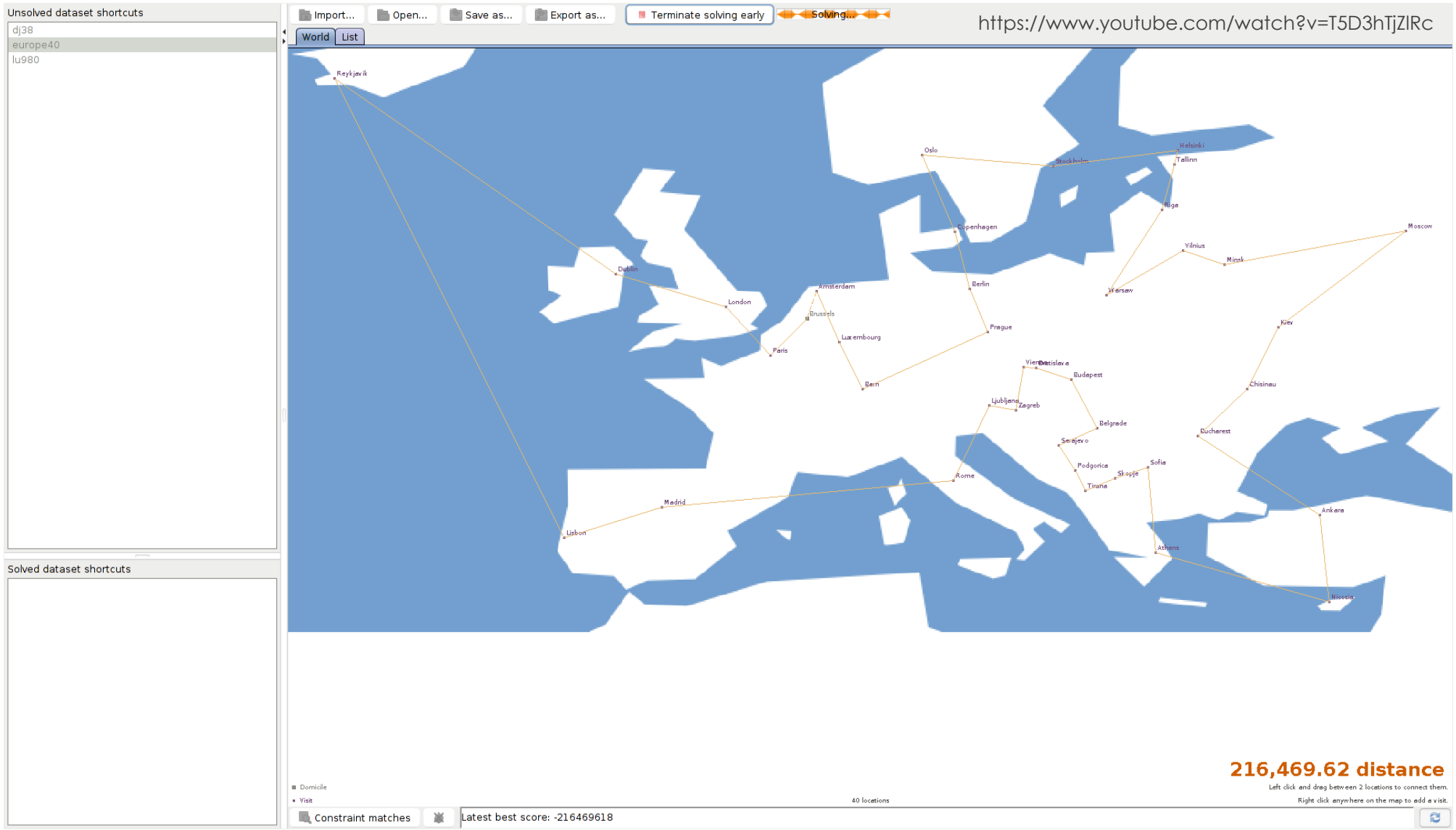
[Optional] Search Reasoning [Workshop]

KIE OptaPlanner Tutorial – TSP: Europe cities



[Optional] Search Reasoning [Workshop]

KIE OptaPlanner Tutorial – TSP: Europe cities



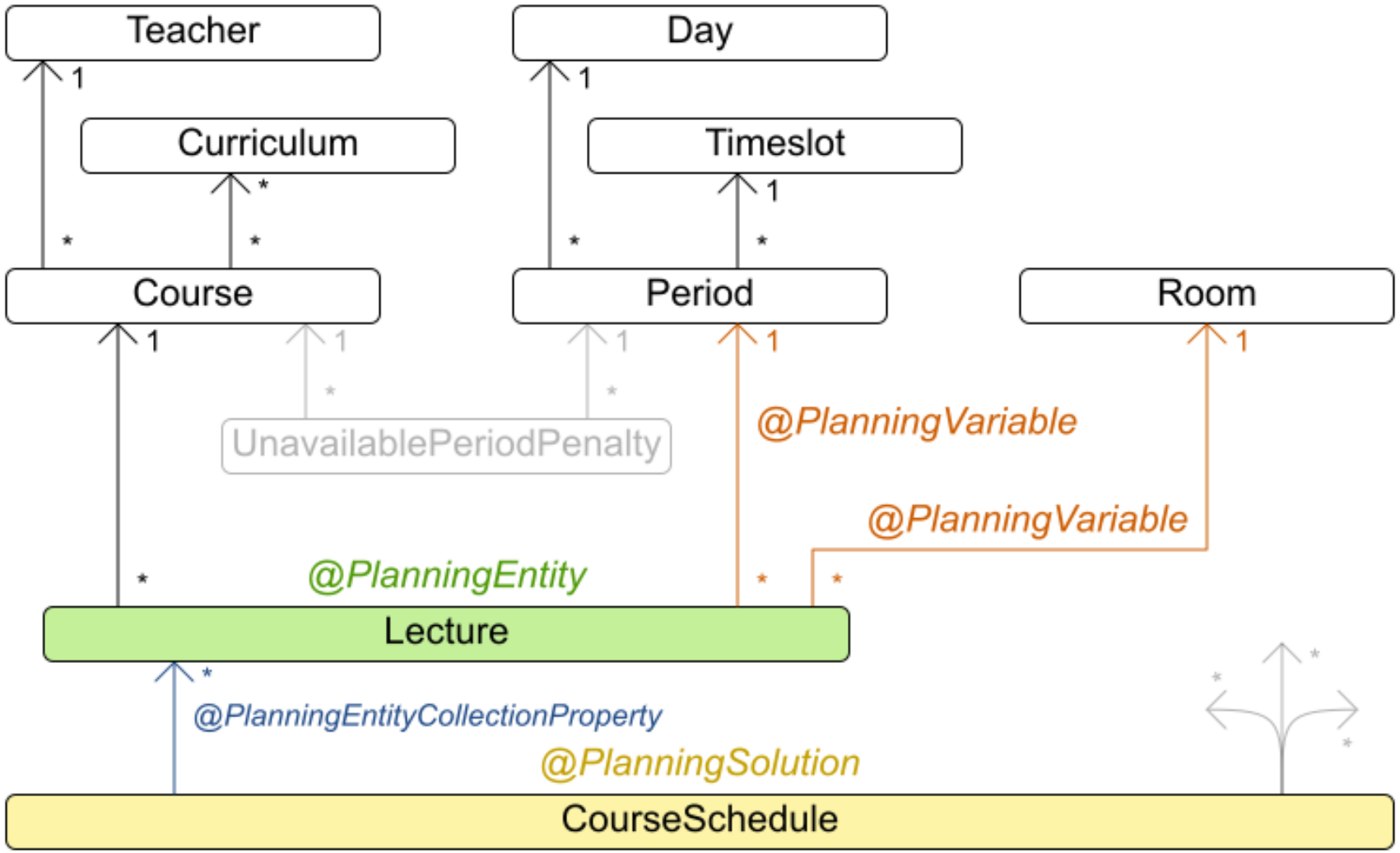
[Optional] Search Reasoning [Workshop]

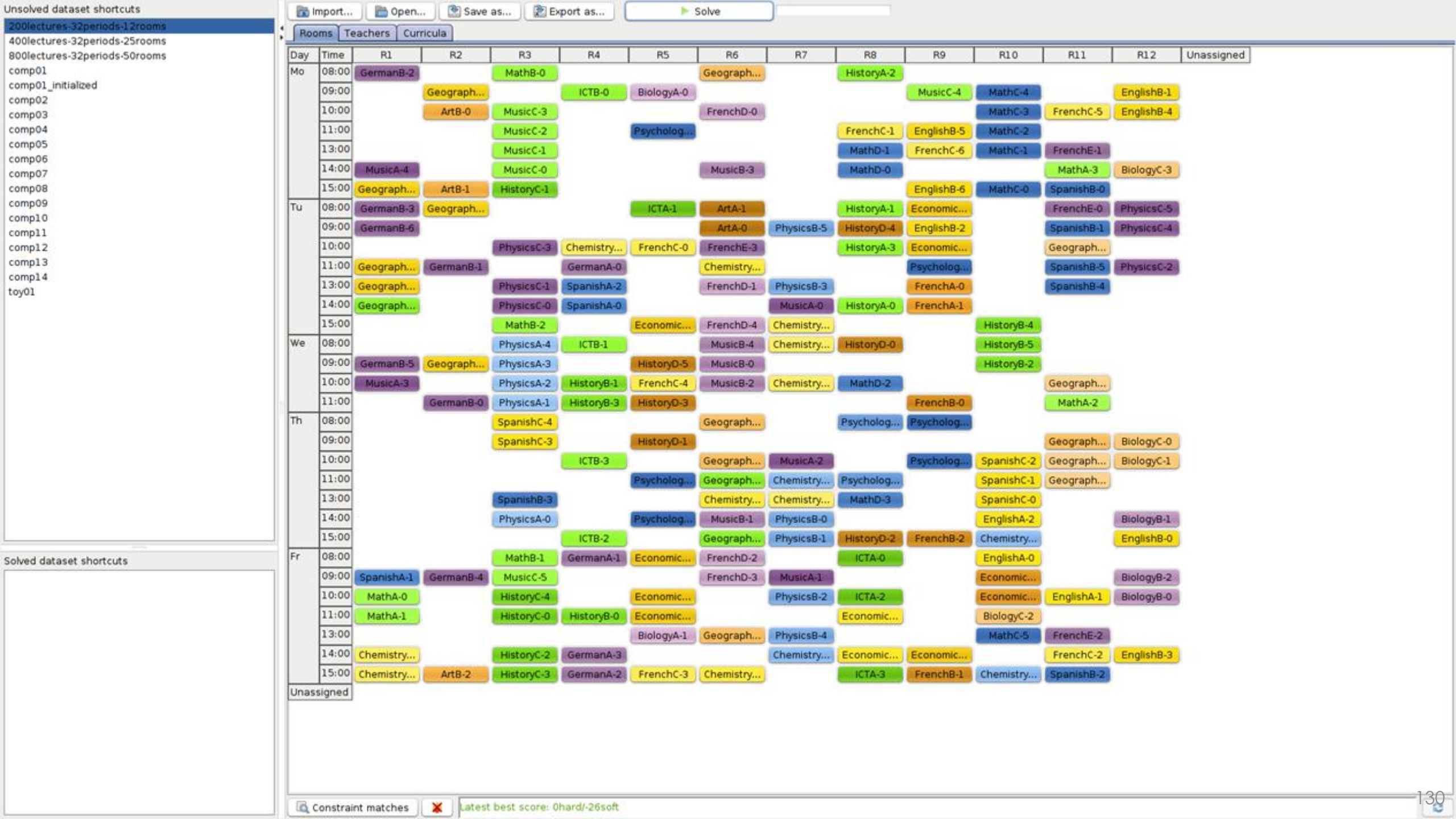
Use Case: Curriculum Course Scheduling

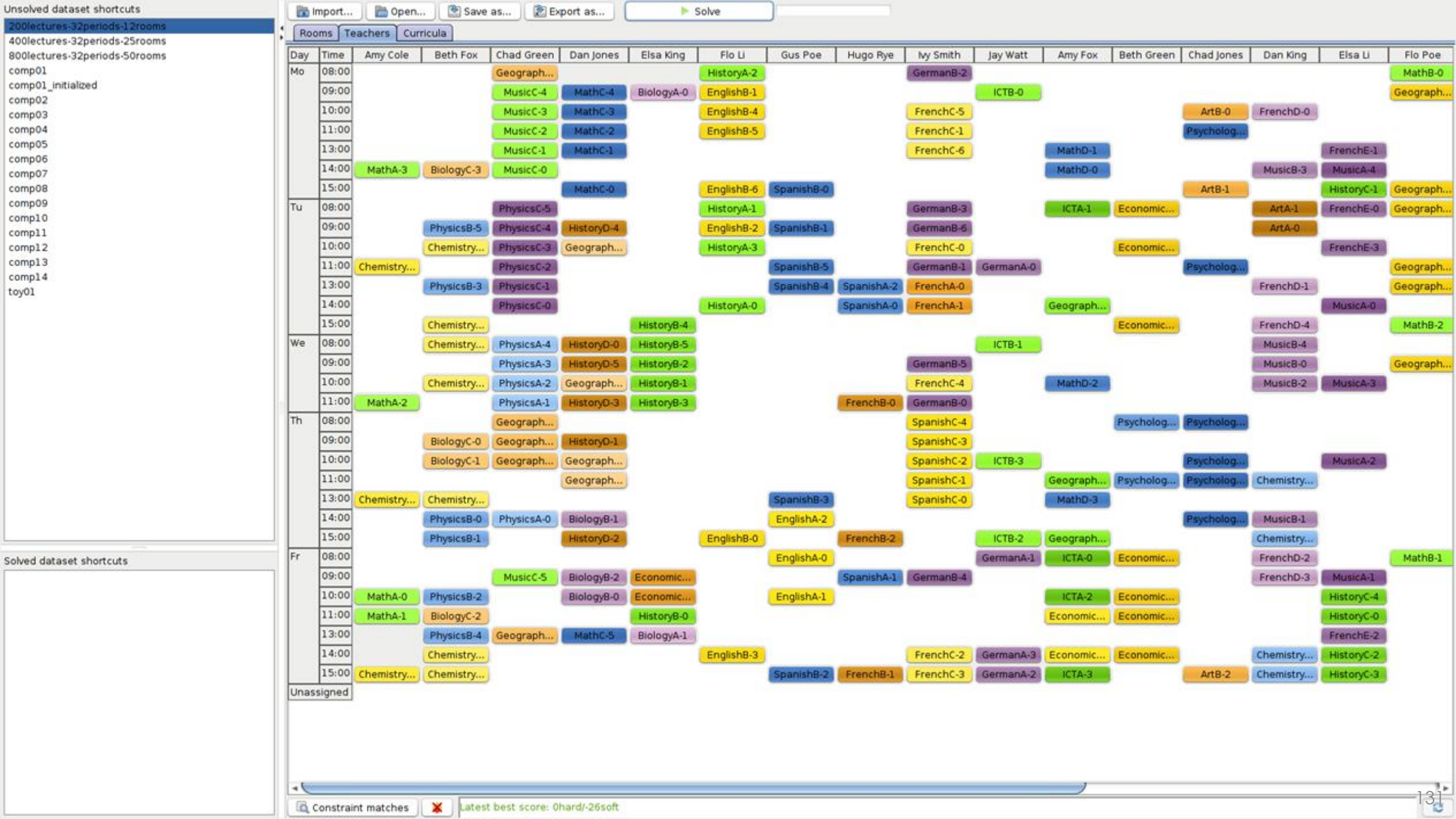
- **Business Scenario / Problem Description**
- **Schedule each lecture into a timeslot and into a room.**
- **Hard constraints:**
  - Teacher conflict: A teacher must not have two lectures in the same period.
  - Curriculum conflict: A curriculum must not have two lectures in the same period.
  - Room occupancy: two lectures must not be in the same room in the same period.
  - Unavailable period (specified per dataset): A specific lecture must not be assigned to a specific period.
- **Soft constraints:**
  - Room capacity: A room's capacity should not be less than the number of students in its lecture.
  - Minimum working days: Lectures of the same course should be spread out into a minimum number of days.
  - Curriculum compactness: Lectures belonging to the same curriculum should be adjacent to each other (so in consecutive periods).
  - Room stability: Lectures of the same course should be assigned to the same room.
- **The problem is defined by the International Timetabling Competition 2007 track 3.**

  http://www.cs.qub.ac.uk/itc2007/curriculmcourse/course_curriculm_index.htm

Curriculum course class diagram




Unsolved dataset shortcuts
200lectures-32periods-12rooms
400lectures-32periods-25rooms
800lectures-32periods-50rooms
comp01
comp01_initialized
comp02
comp03
comp04
comp05
comp06
comp07
comp08
comp09
comp10
comp11
comp12
comp13
comp14
toy01
Solved dataset shortcuts
Import...
Open...
Save as...
Export as...
Solve
Rooms
Teachers
Curricula
Day Time R1 R2 R3 R4 R5 R6 R7 R8 R9 R10 R11 R12 Unassigned
Mo 08:00 GermanB-2 MathB-0 Geograph... HistoryA-2
09:00 Geograph... ICTB-0 BiologyA-0 MusicC-4 MathC-4 EnglishB-1
10:00 ArtB-0 MusicC-3 FrenchD-0 MathC-3 FrenchC-5 EnglishB-4
11:00 MusicC-2 Psycholog... FrenchC-1 EnglishB-5 MathC-2
13:00 MusicC-1 MathD-1 FrenchC-6 MathC-1 FrenchE-1
14:00 MusicA-4 MusicC-0 MusicB-3 MathD-0 MathA-3 BiologyC-3
15:00 Geograph... ArtB-1 HistoryC-1 EnglishB-6 MathC-0 SpanishB-0
Tu 08:00 GermanB-3 Geograph... ICTA-1 ArtA-1 HistoryA-1 Economic... FrenchE-0 PhysicsC-5
09:00 GermanB-6 ArtA-0 PhysicsB-5 HistoryD-4 EnglishB-2 SpanishB-1 PhysicsC-4
10:00 PhysicsC-3 Chemistry... FrenchC-0 FrenchE-3 HistoryA-3 Economic... Geograph...
11:00 Geograph... GermanB-1 GermanA-0 Chemistry... Psycholog... SpanishB-5 PhysicsC-2
13:00 Geograph... PhysicsC-1 SpanishA-2 FrenchD-1 PhysicsB-3 FrenchA-0 SpanishB-4
14:00 Geograph... PhysicsC-0 SpanishA-0 MusicA-0 HistoryA-0 FrenchA-1
15:00 MathB-2 Economic... FrenchD-4 Chemistry... HistoryB-4
We 08:00 PhysicsA-4 ICTB-1 MusicB-4 Chemistry... HistoryD-0 HistoryB-5
09:00 GermanB-5 Geograph... PhysicsA-3 HistoryD-5 MusicB-0 HistoryB-2
10:00 MusicA-3 PhysicsA-2 HistoryB-1 FrenchC-4 MusicB-2 Chemistry... MathD-2 Geograph...
11:00 GermanB-0 PhysicsA-1 HistoryB-3 HistoryD-3 FrenchB-0 MathA-2
Th 08:00 SpanishC-4 Geograph... Psycholog... Psycholog...
09:00 SpanishC-3 HistoryD-1 Geograph... BiologyC-0
10:00 ICTB-3 Geograph... MusicA-2 Psycholog... SpanishC-2 Geograph... BiologyC-1
11:00 Psycholog... Geograph... Chemistry... Psycholog... SpanishC-1 Geograph...
13:00 SpanishB-3 Chemistry... Chemistry... MathD-3 SpanishC-0
14:00 PhysicsA-0 Psycholog... MusicB-1 PhysicsB-0 EnglishA-2 BiologyB-1
15:00 ICTB-2 Geograph... PhysicsB-1 HistoryD-2 FrenchB-2 Chemistry... EnglishB-0
Fr 08:00 MathB-1 GermanA-1 Economic... FrenchD-2 ICTA-0 EnglishA-0
09:00 SpanishA-1 GermanB-4 MusicC-5 FrenchD-3 MusicA-1 Economic... BiologyB-2
10:00 MathA-0 HistoryC-4 Economic... PhysicsB-2 ICTA-2 Economic... EnglishA-1 BiologyB-0
11:00 MathA-1 HistoryC-0 HistoryB-0 Economic... Economic... BiologyC-2
13:00 BiologyA-1 Geograph... PhysicsB-4 MathC-5 FrenchE-2
14:00 Chemistry... HistoryC-2 GermanA-3 Chemistry... Economic... Economic... FrenchC-2 EnglishB-3
15:00 Chemistry... ArtB-2 HistoryC-3 GermanA-2 FrenchC-3 Chemistry... ICTA-3 FrenchB-1 Chemistry... SpanishB-2
Unassigned
Constraint matches
Latest best score: 0hard/-26soft


Unsolved dataset shortcuts
200lectures-32periods-12rooms
400lectures-32periods-25rooms
800lectures-32periods-50rooms
comp01
comp01_initialized
comp02
comp03
comp04
comp05
comp06
comp07
comp08
comp09
comp10
comp11
comp12
comp13
comp14
toy01
Solved dataset shortcuts
Import...
Open...
Save as...
Export as...
Solve
Rooms
Teachers
Curricula
Day Time Amy Cole Beth Fox Chad Green Dan Jones Elsa King Flo Li Gus Poe Hugo Rye Ivy Smith Jay Watt Amy Fox Beth Green Chad Jones Dan King Elsa Li Flo Poe
Mo 08:00 Geograph... HistoryA-2 GermanB-2 MathB-0
09:00 MusicC-4 MathC-4 BiologyA-0 EnglishB-1 ICTB-0 Geograph...
10:00 MusicC-3 MathC-3 EnglishB-4 FrenchC-5 ArtB-0 FrenchD-0
11:00 MusicC-2 MathC-2 EnglishB-5 FrenchC-1 Psycholog...
13:00 MusicC-1 MathC-1 FrenchC-6 MathD-1 FrenchE-1
14:00 MathA-3 BiologyC-3 MusicC-0 MathD-0 MusicB-3 MusicA-4
15:00 MathC-0 EnglishB-6 SpanishB-0 ArtB-1 HistoryC-1 Geograph...
Tu 08:00 PhysicsC-5 HistoryA-1 GermanB-3 ICTA-1 Economic... ArtA-1 FrenchE-0 Geograph...
09:00 PhysicsB-5 PhysicsC-4 HistoryD-4 EnglishB-2 SpanishB-1 GermanB-6 ArtA-0
10:00 Chemistry... PhysicsC-3 Geograph... HistoryA-3 FrenchC-0 Economic... FrenchE-3
11:00 Chemistry... PhysicsC-2 SpanishB-5 GermanB-1 GermanA-0 Psycholog... Geograph...
13:00 PhysicsB-3 PhysicsC-1 SpanishB-4 SpanishA-2 FrenchA-0 FrenchD-1 Geograph...
14:00 PhysicsC-0 HistoryA-0 SpanishA-0 FrenchA-1 Geograph... MusicA-0
15:00 Chemistry... HistoryB-4 Economic... FrenchD-4 MathB-2
We 08:00 Chemistry... PhysicsA-4 HistoryD-0 HistoryB-5 ICTB-1 MusicB-4
09:00 PhysicsA-3 HistoryD-5 HistoryB-2 GermanB-5 MusicB-0 Geograph...
10:00 Chemistry... PhysicsA-2 Geograph... HistoryB-1 FrenchC-4 MathD-2 MusicB-2 MusicA-3
11:00 MathA-2 PhysicsA-1 HistoryD-3 HistoryB-3 FrenchB-0 GermanB-0
Th 08:00 Geograph... SpanishC-4 Psycholog... Psycholog...
09:00 BiologyC-0 Geograph... HistoryD-1 SpanishC-3
10:00 BiologyC-1 Geograph... Geograph... SpanishC-2 ICTB-3 Psycholog... MusicA-2
11:00 Geograph... SpanishC-1 Geograph... Psycholog... Psycholog... Chemistry...
13:00 Chemistry... Chemistry... SpanishB-3 SpanishC-0 MathD-3
14:00 PhysicsB-0 PhysicsA-0 BiologyB-1 EnglishA-2 Psycholog... MusicB-1
15:00 PhysicsB-1 HistoryD-2 EnglishB-0 FrenchB-2 ICTB-2 Geograph... Chemistry...
Fr 08:00 EnglishA-0 GermanA-1 ICTA-0 Economic... FrenchD-2 MathB-1
09:00 MusicC-5 BiologyB-2 Economic... SpanishA-1 GermanB-4 FrenchD-3 MusicA-1
10:00 MathA-0 PhysicsB-2 BiologyB-0 Economic... EnglishA-1 ICTA-2 Economic... HistoryC-4
11:00 MathA-1 BiologyC-2 HistoryB-0 Economic... Economic... HistoryC-0
13:00 PhysicsB-4 Geograph... MathC-5 BiologyA-1 FrenchE-2
14:00 Chemistry... EnglishB-3 FrenchC-2 GermanA-3 Economic... Economic... Chemistry... HistoryC-2
15:00 Chemistry... Chemistry... SpanishB-2 FrenchB-1 FrenchC-3 GermanA-2 ICTA-3 ArtB-2 Chemistry... HistoryC-3
Unassigned
Constraint matches
Latest best score: 0hard/-26soft

NICF- Reasoning Systems (SF)

Overview

Reference No	CRS-Q-0036478-ICT
Part of	-
Duration	5 days
Course Time	9:00am - 5:00pm
Enquiry	Please contact Ms. Jaymee TAN at tel: 65161206 or email isstfhj@nus.edu.sg for more details.

How can you capitalise on the use of Artificial Intelligence – Reasoning Systems to drive innovation and efficiency in your organisation? This 5-day course enables participants to understand the relevant knowledge needed to architect and/or lead teams executing intelligent system projects to reason, simulate, and optimise complex business problems. For example, how to predict future workload and staffing requirements by area, designation, skill and/or role; and how to dynamically redeploy workforce based on unplanned events (sick leave, workload and customer orders fluctuations) in real-time.

Through a mix of lectures reinforced by case examples, participants will acquire comprehensive knowledge of artificial intelligence (AI) techniques, including search, scheduling, optimisation, constraint satisfaction, evolutionary computation, and data mining. Participants will also get hands-on learning to integrate hybrid reasoning systems

This course is for business managers, data specialists, consultants, IT professionals and business professionals interested in learning how reasoning systems with AI optimization techniques can be applied into an organization to drive innovation, efficiency and identify competitive advantages.

This course is a part of the Artificial Intelligence and Graduate Certificate in Intelligent Reasoning Systems, which is a part of the Stackable Graduate Certificate Programme in Artificial Intelligent Systems (Masters Degree) offered by NUS-ISS.

Upcoming Classes

Class 1 12 Nov 2019 to 18 Nov 2019 (Full Time) −

Duration: 5 days

When:
Nov: 12, 13, 14, 15, 18

Time:
9:00am - 5:00pm

NICF- Problem Solving using Pattern Recognition (SF)

Overview

Reference No	CRS-Q-0035256-ICT
Part of	-
Duration	5 days
Course Time	9:00am - 5:00pm
Enquiry	Please contact Ms. Jaymee TAN at tel: 65161206 or email isstfhj@nus.edu.sg for more details.

Pattern recognition is one of the most important areas of Artificial Intelligence. It is a branch of machine learning that focuses on the recognition of patterns and regularities in data. Pattern recognition systems can be trained from labelled training data through supervised learning and or unlabelled data through unsupervised learning.

Pattern recognition has been widely used to solve many real-world problems such as image processing, speech recognition, data mining, business analytics, etc. There are many pattern recognition techniques available to perform different tasks such as regression, classification, clustering, etc. using various statistical and machine learning algorithms.

This course will be useful for participants to acquire pattern recognition knowledge. It will help participants analyse data more effectively by deriving useful hidden patterns in the data. Participants will also learn how to select and apply the most suitable pattern recognition techniques to solve the given problems and develop pattern recognition systems.

This course is part of the Artificial Intelligence and Graduate Certificate in Pattern Recognition Systems Series offered by NUS-ISS

Upcoming Classes

Class 1 04 Nov 2019 to 21 Nov 2019 (Full Time) −

Duration: 5 days

When:
Nov: 04, 05, 14, 15, 21

Time:
9.00am - 5.00pm

Real case:

Conflict of ISS course schedule

What are possible root causes?

Can you provide an AI solution?

Food for Thought

You are a manager of a team of AI engineers working on ISS course scheduling problem. One engineer, who is highly experienced in machine learning and deep learning, managed to collect past 20+ years of historical course schedules, carefully modelled to extract targets for supervised learning, e.g. the course schedule sequence/array in every 12 months (as ISS plans/publishes course schedule yearly), and then used robust supervised machine/deep learning algorithms to train and finally obtained a predictive model. Upon using the planning/predictive model, the first course schedule A was generated, and a long-time ISS admin course planner also parallelly created a course schedule B for comparison/evaluation. It's found that schedule A is about 10 times better (in predefined/business/practical/reasonable real world evaluation) than schedule B.

You as a manager plans to promote that engineer with a big annual bonus. During lunch time, you bragged this achievement to one AI guy, named Sam. But you were shocked: Sam recommended you to sack that engineer.

Now, you are wondering:

A) Sam is wrong, because _____ (other than Sam is a cynical joker.)

B) Sam is right, because _____ (other than you eager to see Sam slapped by that engineer.)

C) Sam is neither right or wrong, please suggest the next steps to confirm right/wrong _____

D) Leave this answer blank to downgrade one GPA level.

[Optional] Search Reasoning [Workshop]

Refer to ./optional workshop/S-RS Workshop Guide(s).pdf

END OF APPENDICES