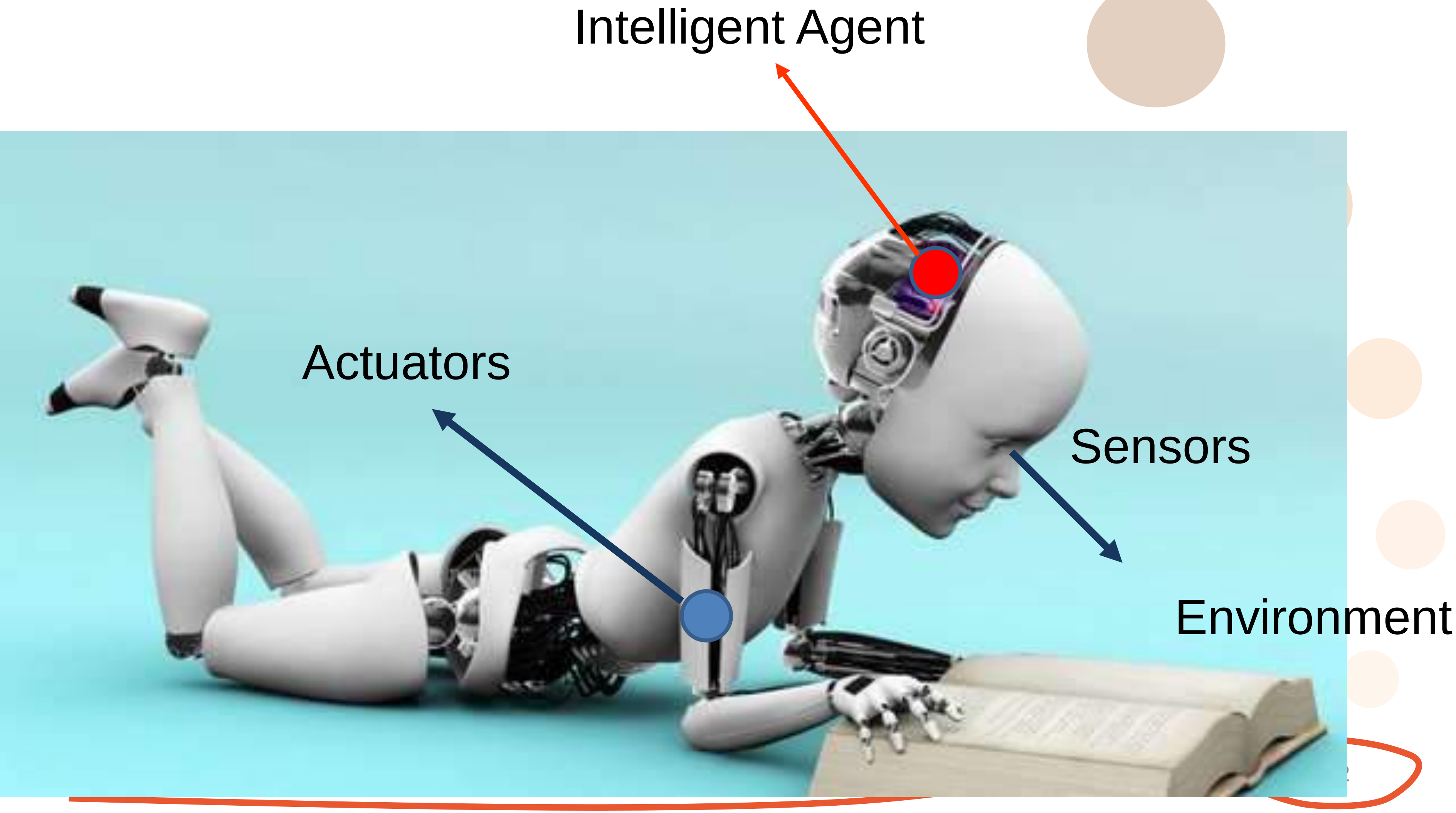
TARC
TUNKU ABDUL RAHMAN UNIVERSITY COLLEGE
BEYOND EDUCATION

Artificial Intelligence

Chapter 2 Problem-Definition & Problem Solving

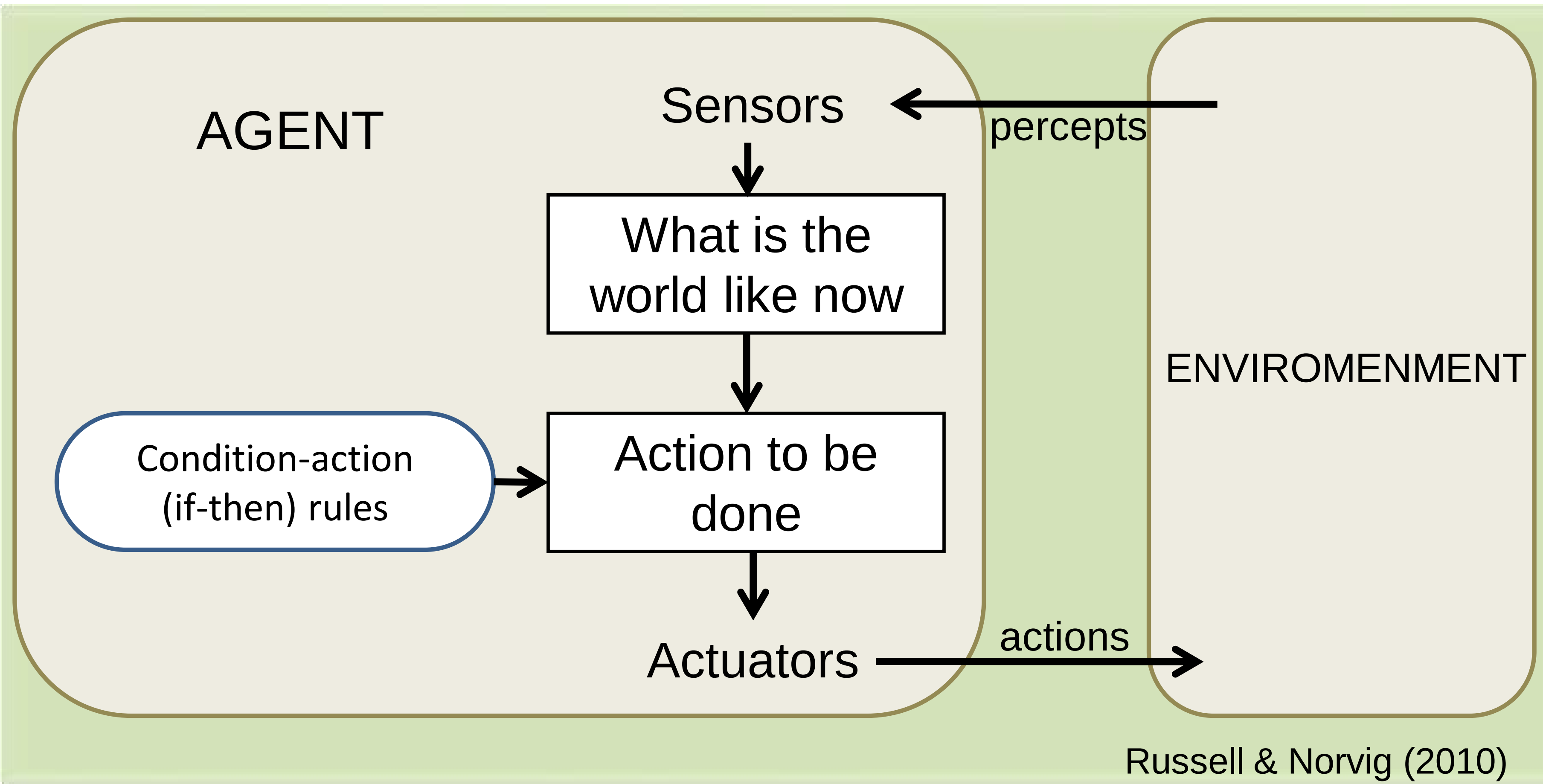
Chapter Outline

- Problem-solving concept
- Measuring problem-solving performance

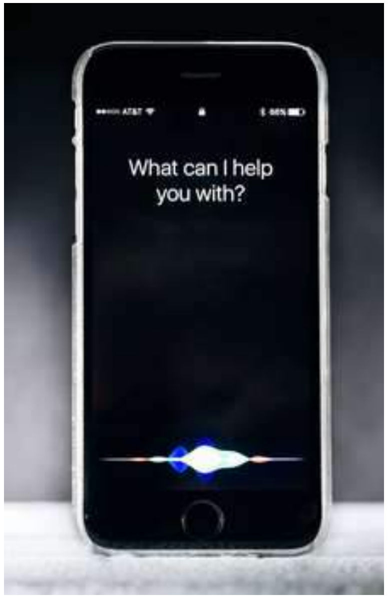

Intelligent Agent
Actuators
Sensors
Environment

Problem-solving agents

- They decide what to do by finding sequences of actions that lead to desirable states.



Examples of Intelligent Agent



Personal assistant in smartphones

Programs of the self-driving cars

Thermostat

Problem-solving concept

1st step in problem solving

Goal Formulation

Problem Formulation

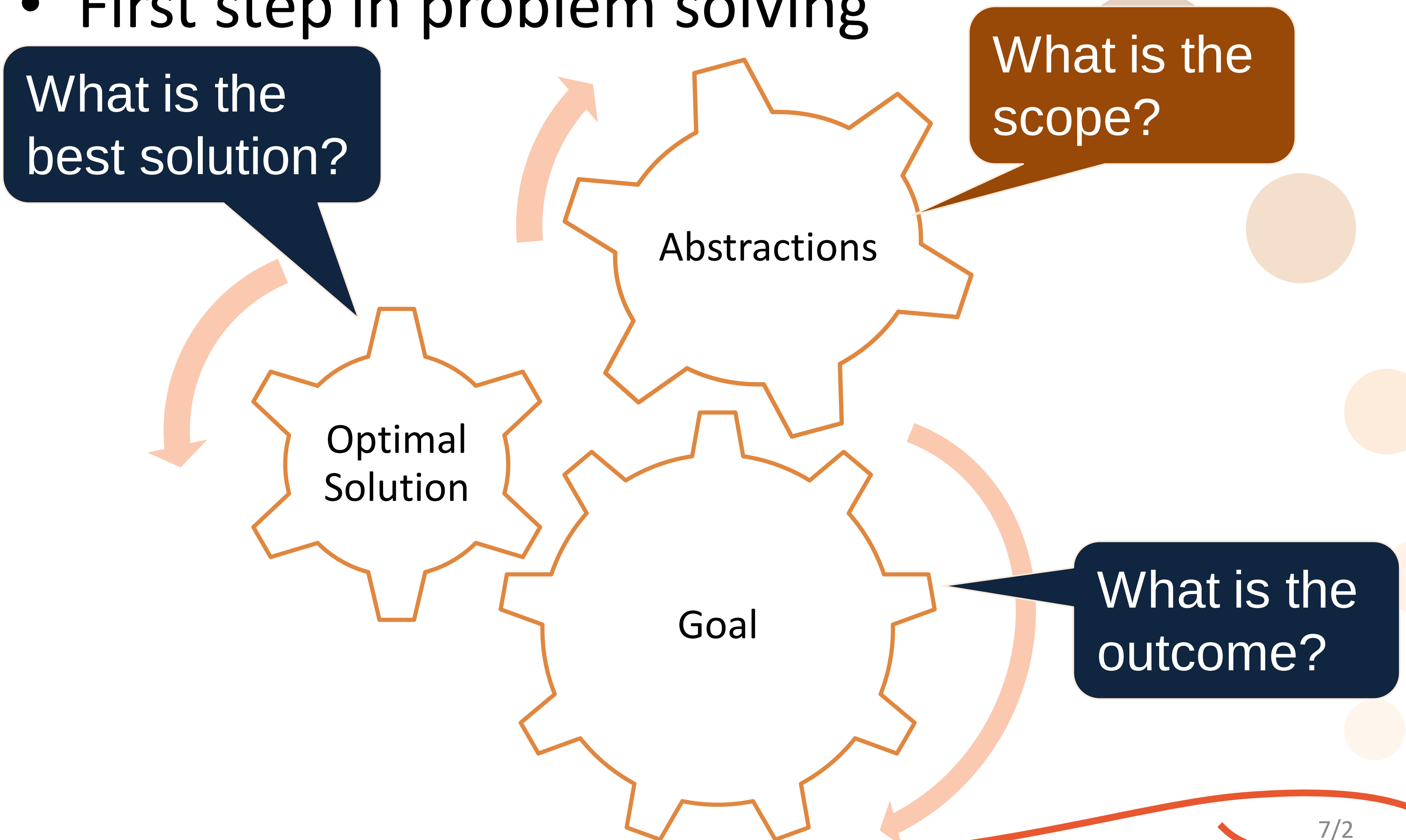
Process of deciding what actions and states to consider.

Search-solution-execution

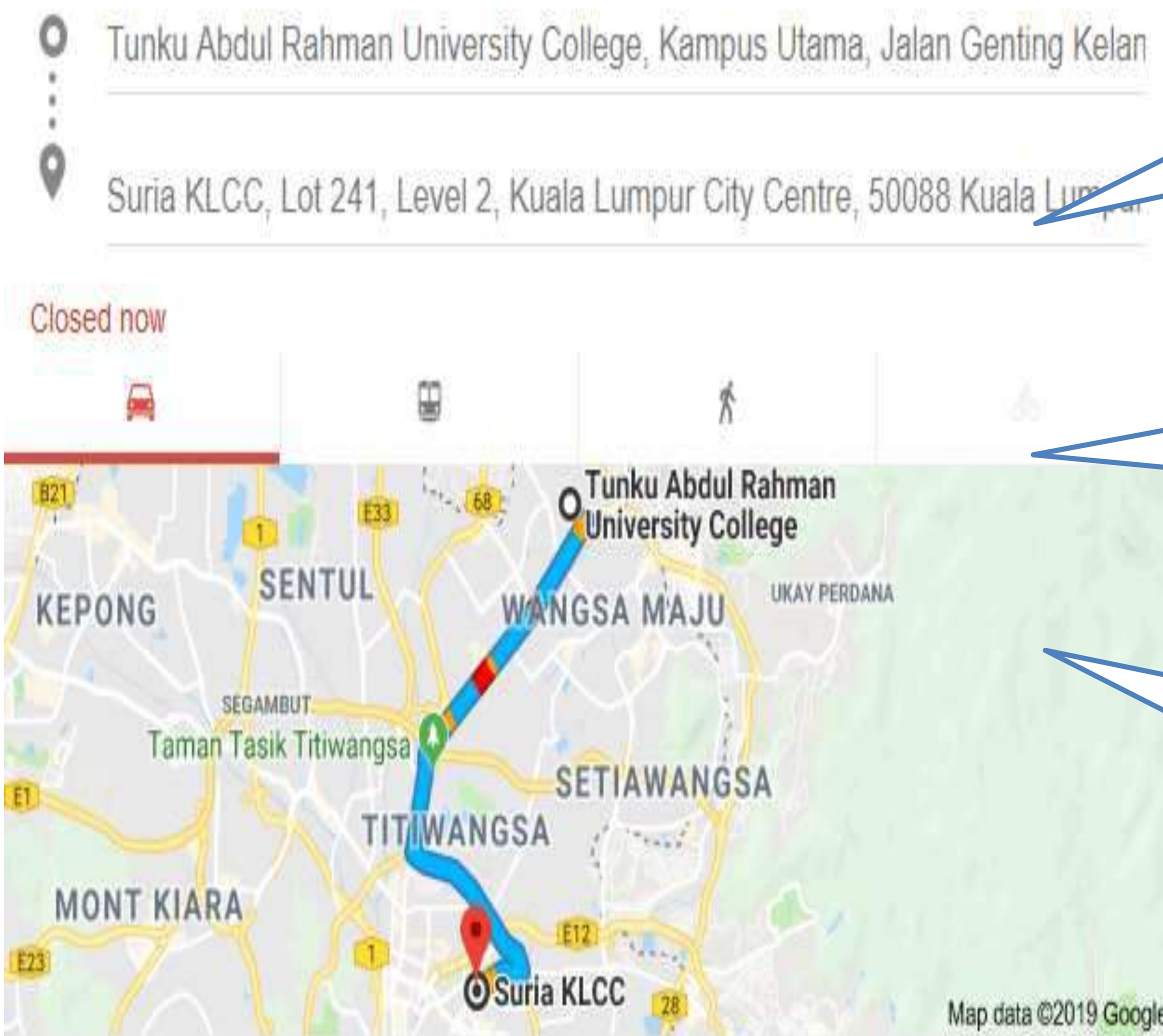
Process of looking for a sequence of actions.

Goal formulation

- First step in problem solving



Example: Travel



Goal?

Optimal Solution?

Abstractions?

Step 1: Formulating Goal

Goal

- To reach KLCC

Optimal Solution

- To get the shortest path to KLCC

Abstraction (Scope and remove unwanted details)

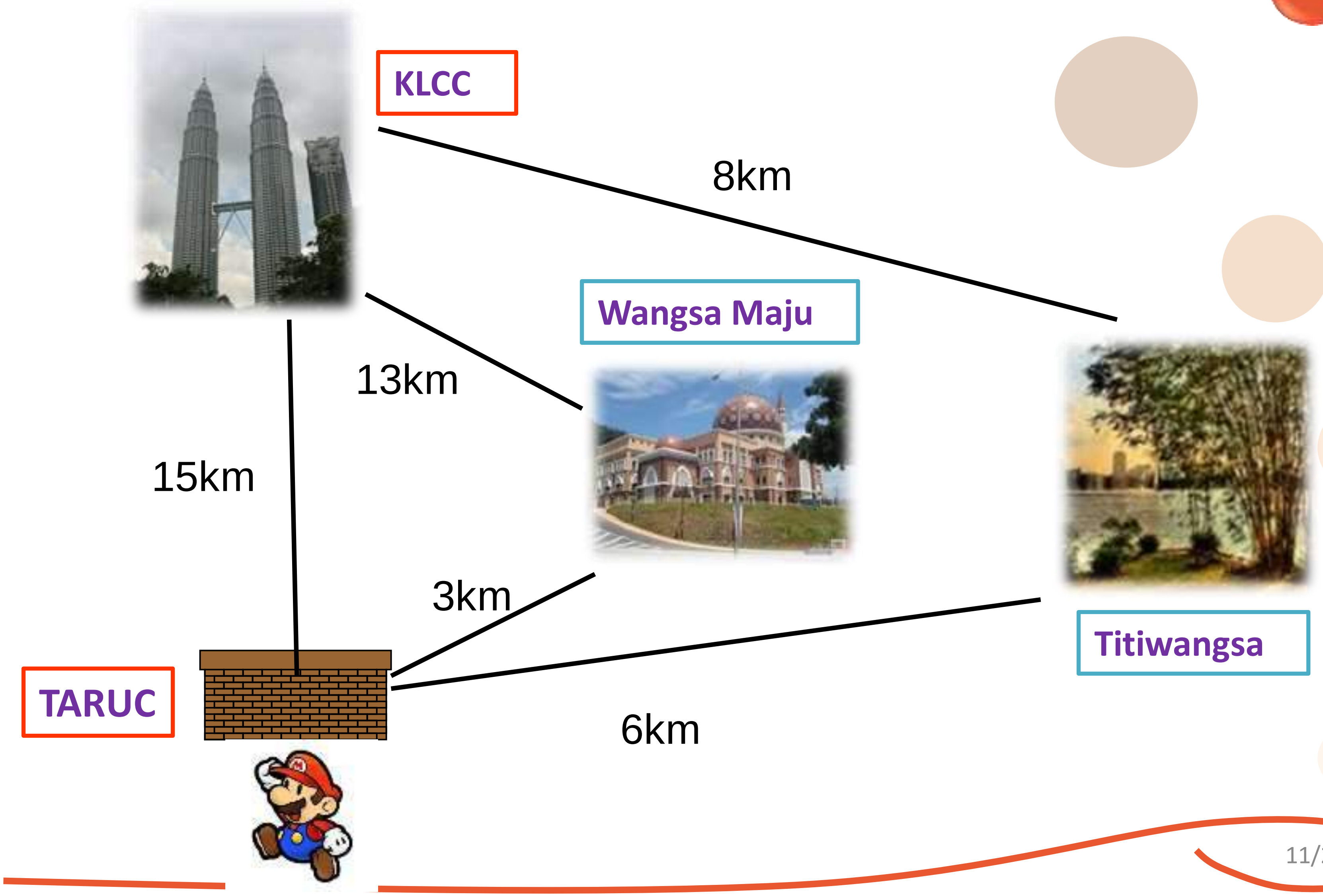
- E.g. We do not care about the time used to reach KLCC, which can be caused by unexpected factors, such as traffic jam

Problem definition

A problem can be defined by 4 components:

1. Initial State
   - How it starts?
2. Successor Function, State Space, Path
   - How is it solved?
3. Goal Test
   - Is it the desired outcome?
4. Step Cost, Path Cost
   - What is the cost?

Problem (1): Path Finding



Step 2: Formulating Problem (1)

1 **Initial state** (the state that the agent starts in)

– e.g. Go(_ , In(TARUC), 0)

Step 2: Formulating Problem (1)

2 **Successor function** (the possible Actions available to the agent, that will change the state)

For example, a function can be given as:

- *Go(To(child), In(Parent), State_Cost)*

E.g.:

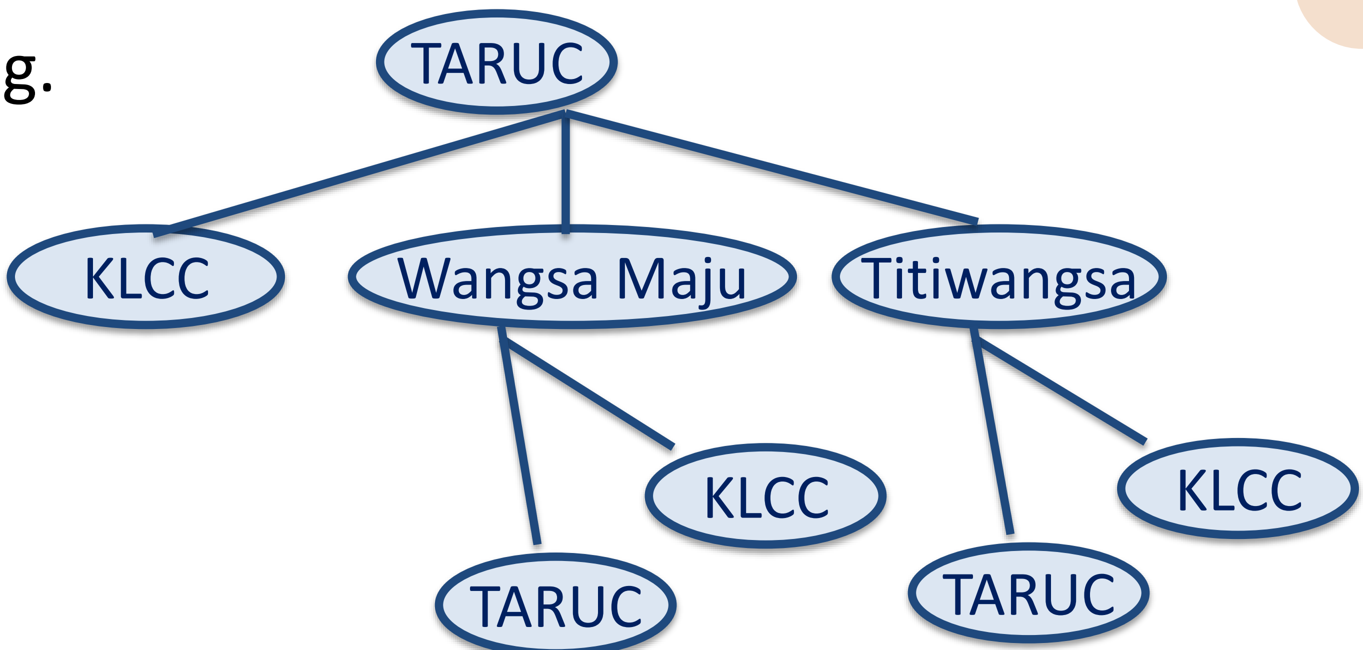
- *Go(To(Wangsa Maju, In(TARUC), 3)*
- *Go(To(KLCC), In(Wangsa Maju), 13),*

Step 2: Formulating Problem (1)

2 **State space** (the set of all states reachable from the initial state)

– Usually represented by a graph or a tree

– E.g.



Step 2: Formulating Problem (1)

2 **Path** (Sequence of states connected by a sequence of actions)

Step 2: Formulating Problem (1)

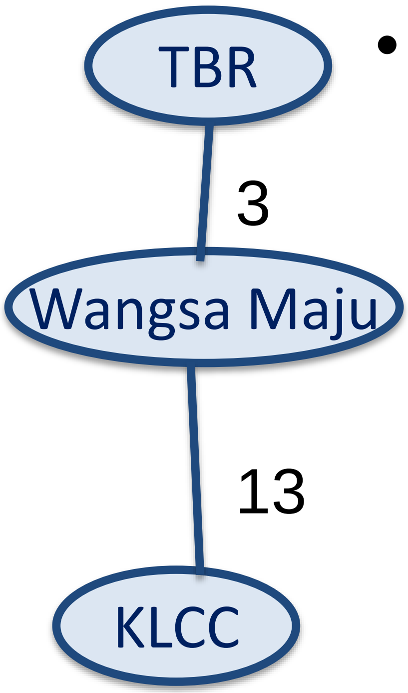
3 **Goal Test** (To determine whether a given state is a goal state)

– Sample of Goal Test Algorithm

```
If
      Go(To(KLCC), In( _ ), _ )
Then
      Go( _ , In(KLCC), _),
     show pathcost,
     return true
```

Step 2: Formulating Problem (1)

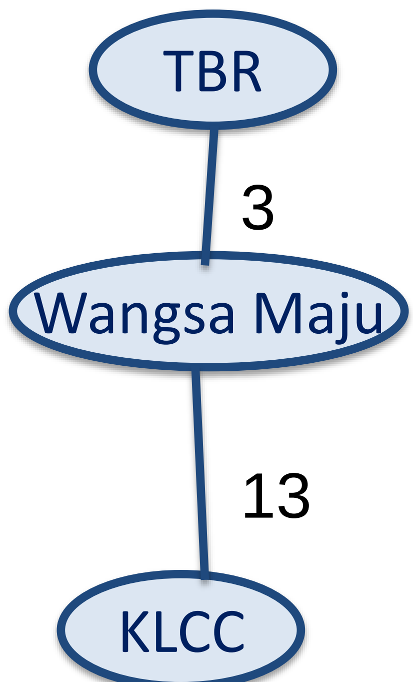
4 **Step cost** (route distance, from one state to another state)



- E.g. the step cost from TBR to Wangsa Maju = 3km

Step 2: Formulating Problem (1)

4 **Path cost (**Sum of the costs of the individual actions along the path)



16 = 3+13

Problem Solving Concept

- Goal Formulation
  - Goal
  - Optimal Solution
  - Abstraction

- Problem Formulation
  - Initial State
  - Successor Function
  - State Space
  - Path
  - Goal Test
  - Step/ path cost

Problem (2) Missionaries & Cannibals

The goal state (all on the right)

Boat can hold 1 or 2 people

missionaries >= cannibals

missionary

cannibal

Mission

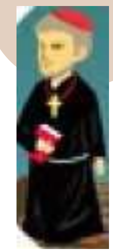
- The goal is given,
  1. What is the optimal solution?
  2. Suggest an abstraction.

Problem Formulation

- **Initial State**
  - characterize the state:
    - the no. of missionaries on the left bank, ML
    - the number of cannibals on the left bank, CL
    - the side the boat is on, B.
  - Representation in code:
    - [ML, CL, B]
    - **start([3, 3, left]).**
    - **goal([0, 0, right]).**

Successor function

- There are 8 possible moves:
  1. Move 1 missionary from the left bank
     - **move**([ML,CL,left],[MLNew,CL, right]).
     - E.g. **move**([3, 3, left], [2, 3, right]).

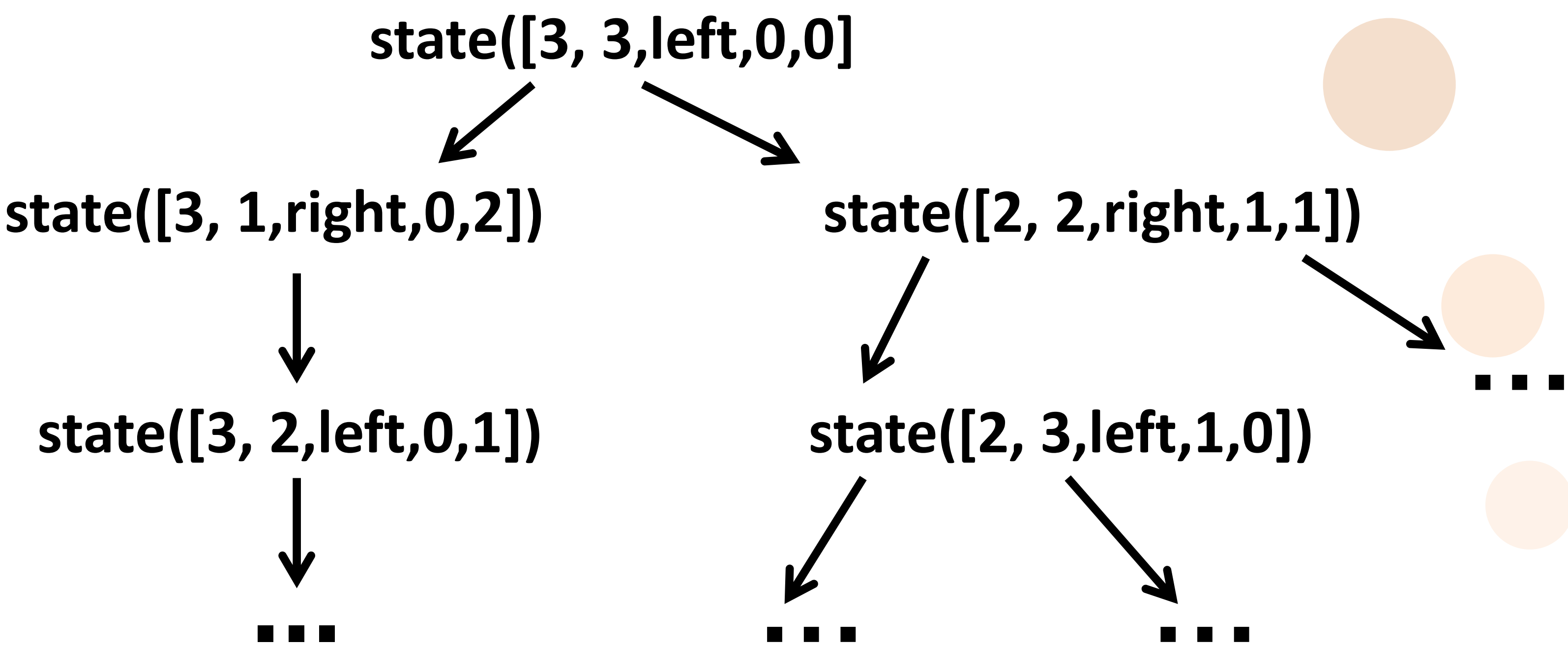


  2. Move 1 cannibal from the left bank
     - **move**([ML, CL, left], [ML, CLNew, right]).
     - E.g. **move**([3, 3, left], [3, 2, right]).

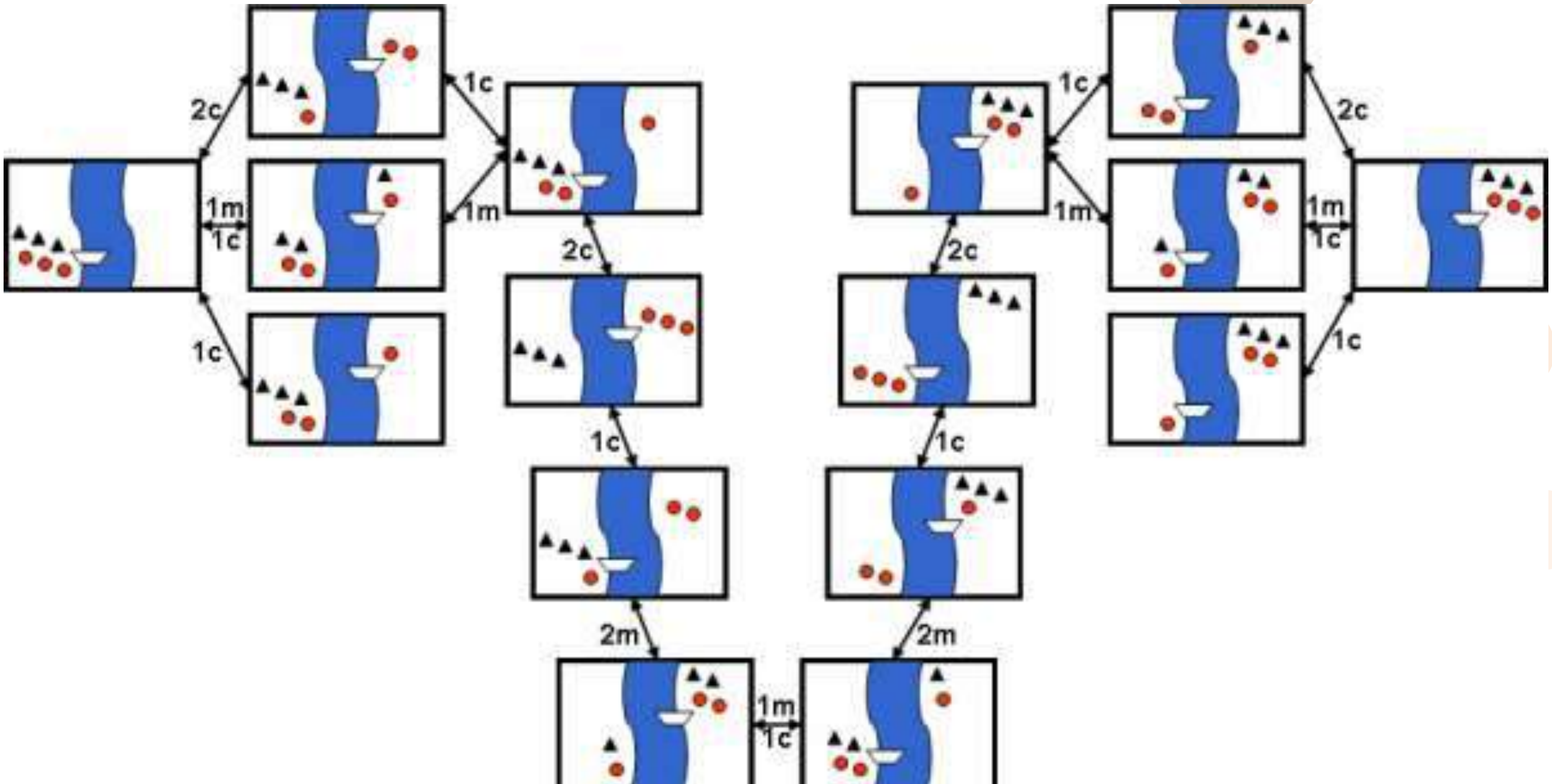
  3. And so on…
- Alternatively, we can represent in such a way
  – State([3,3, Left, 0,0])

State space

- The set of all states reachable from the initial state



Path



Path

- Sequence of states connected by a sequence of actions

```
file:///C:/Documents and Settings/sacha/Desktop/Codeproject/MissionariesAndCannibals/Missionar...
=================================================
  THE SEARCH HAS BEEN SETUP WITH THE
  FOLLOWING OPTIONS
=================================================
1. Missionaries must be equal or greater than Cannibals


As this is a breadth 1st search the higher up the
search tree the solutions are, the cheaper they will
be. So the 1st solutions found will be the optimal
ones. The most optimal solutions are shown below

=====FOUND SOLUTION [1]=====

This solution was found at level [12]

3M/3C <-BOAT LEFT     0M/0C
3M/1C   BOAT RIGHT->  0M/2C
3M/2C <-BOAT LEFT     0M/1C
3M/0C   BOAT RIGHT->  0M/3C
3M/1C <-BOAT LEFT     0M/2C
1M/1C   BOAT RIGHT->  2M/2C
2M/2C <-BOAT LEFT     1M/1C
0M/2C   BOAT RIGHT->  3M/1C
0M/3C <-BOAT LEFT     3M/0C
0M/1C   BOAT RIGHT->  3M/2C
0M/2C <-BOAT LEFT     3M/1C
0M/0C   BOAT RIGHT->  3M/3C
```

path

Goal Test

- It is a function to check whether everyone is carried to the left side of the river bank.

```
goal([0, 0, right]).
search([ML, CL, B], Solution):-
     goal(Solution).
```

Cost

- **Step Cost**
  – Whenever the parent expands the child, the state level will increase by one, **so the step cost is???**
- **Path Cost**
  – The path cost is the number of steps in the path
  – What do you think is the optimal path cost?

Example of Problem (3)

The 8-puzzle

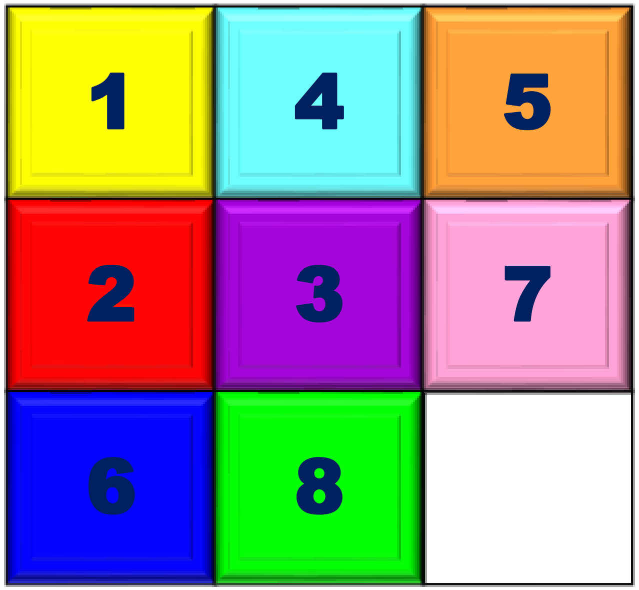


Figure 1

Figure 2

Formulating Problem (3)

1 Initial State

1. Suggest how to turn the initial state into a **code**.

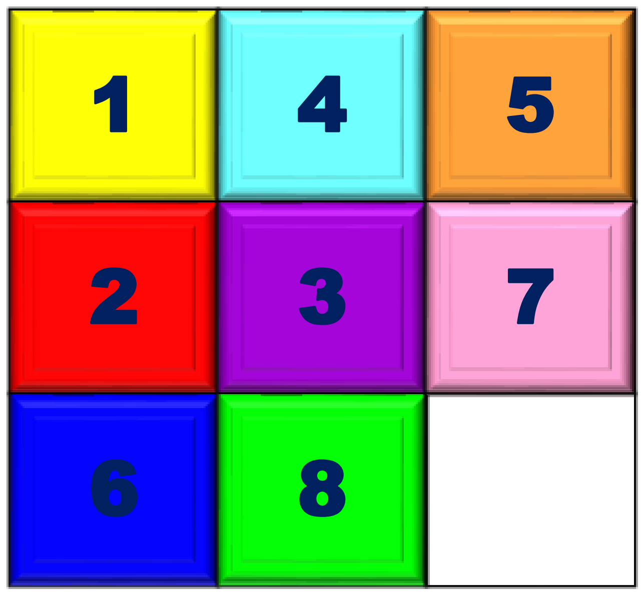
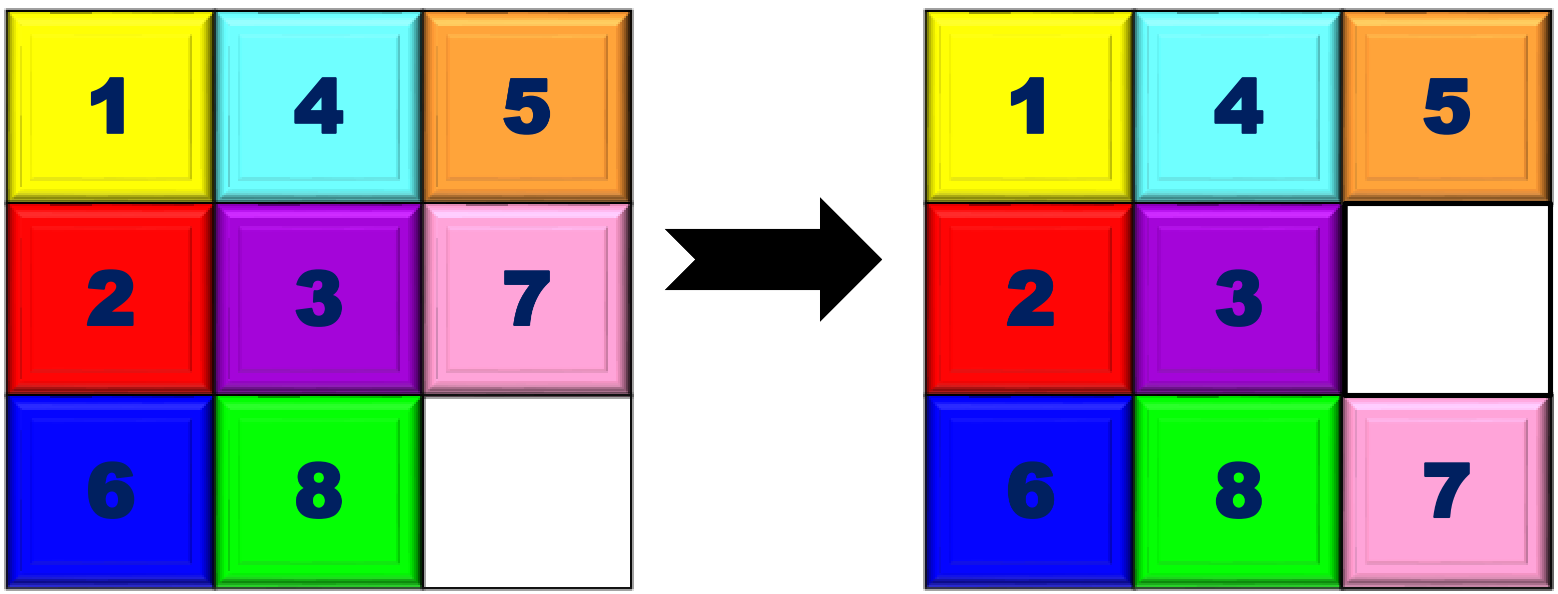


Figure 1

Formulating problem (3)

2 Successor function

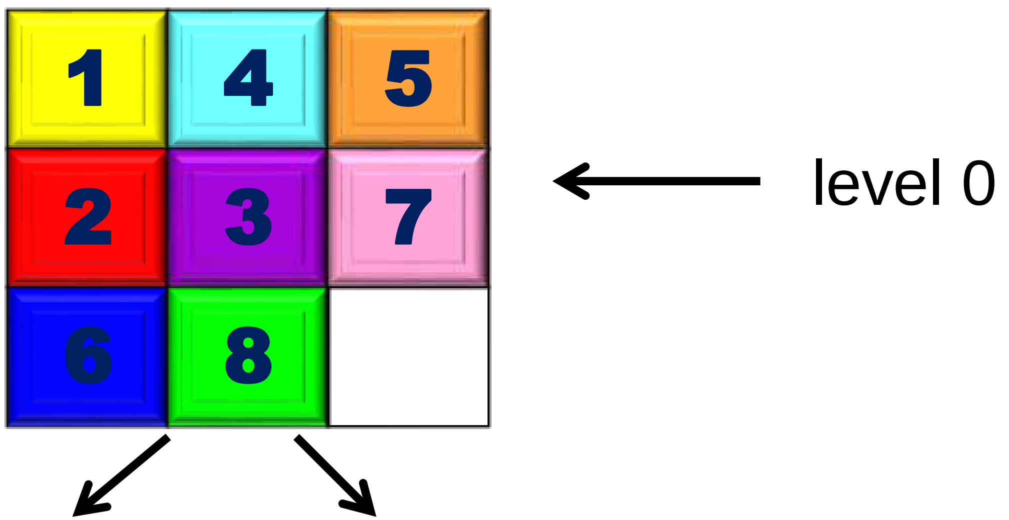
1. What is a successor function?
2. Suggest a successor function to represent:



Formulating problem (3)

2 **State Space**

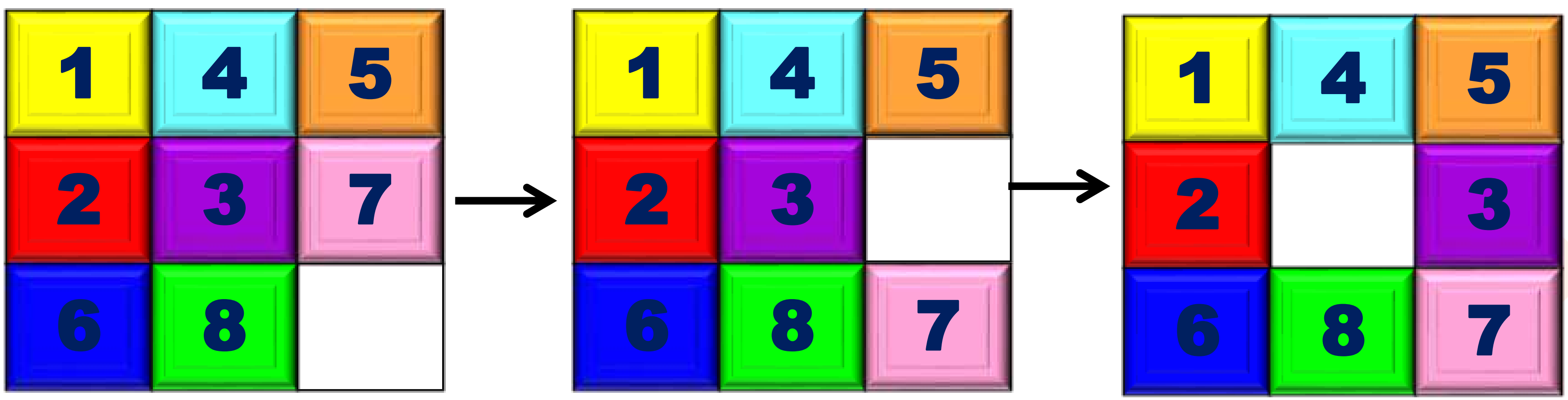
1. Define state space
2. Using a tree, illustrate the partial state space from the initial state until LEVEL 2.



Formulating problem (3)

2 Path

1. Define path



Formulating problem (3)

3 Goal Test

– This checks whether the state matches the goal.

– Example of function looks like this:

```
search(State,GoalState) :-
   goal(GoalState).
```

1. Suggest how do you represent `GoalState`.

Formulating problem (3)

4 Step cost:

1. Suggest the step cost of the 8-puzzle problem.

4 Path cost

1. Define path cost in the context of 8-puzzle problem.

Search-solution-execution

Search

- the process of looking for the best sequence of path

Solution

- A search algorithm takes a problem as input and returns a solution in the form of an action sequence

Execution

- Once a solution is found, the actions it recommends can be carried out.

Searching for Solutions

- Search tree
- Search graph
- Expanding/generating states
- Search strategy

Search Tree/Graph

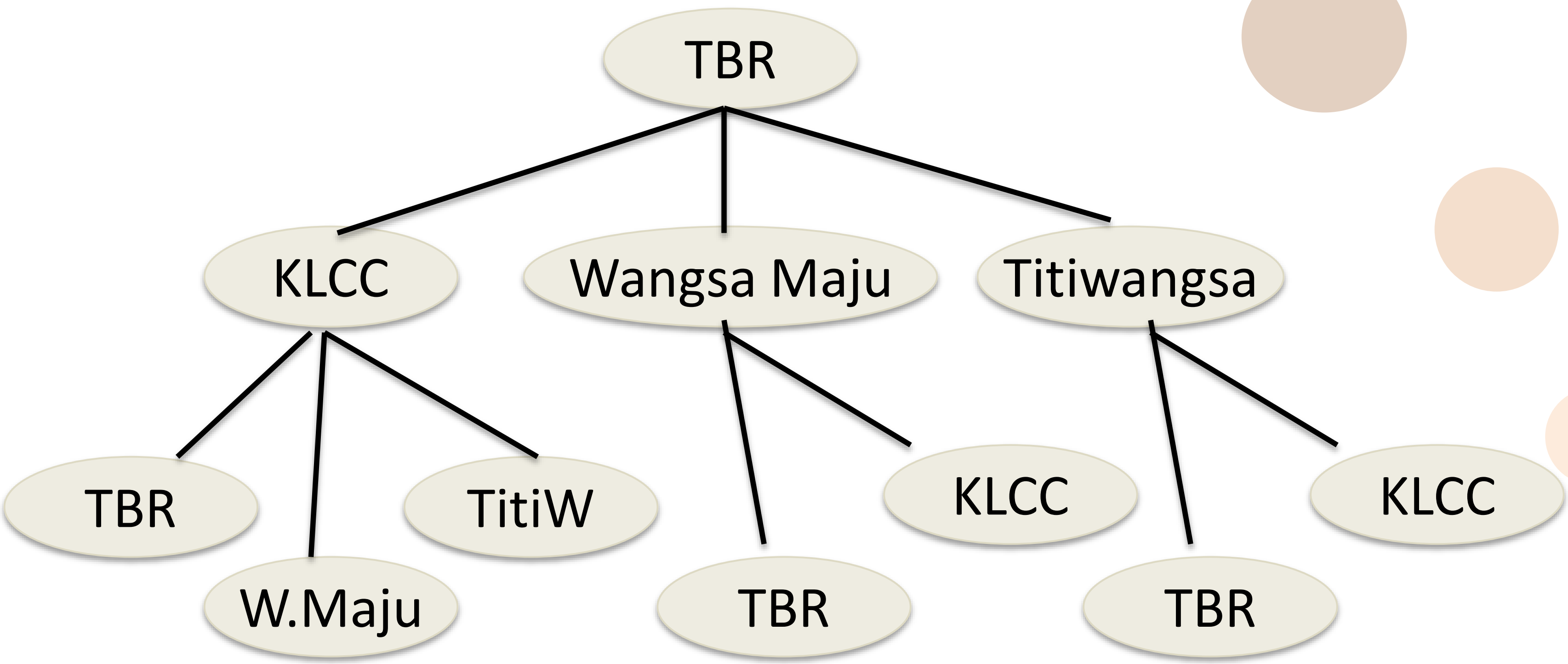
SEARCH TREE

- Generated by the initial state and the successor function that together define state space.

SEARCH GRAPH

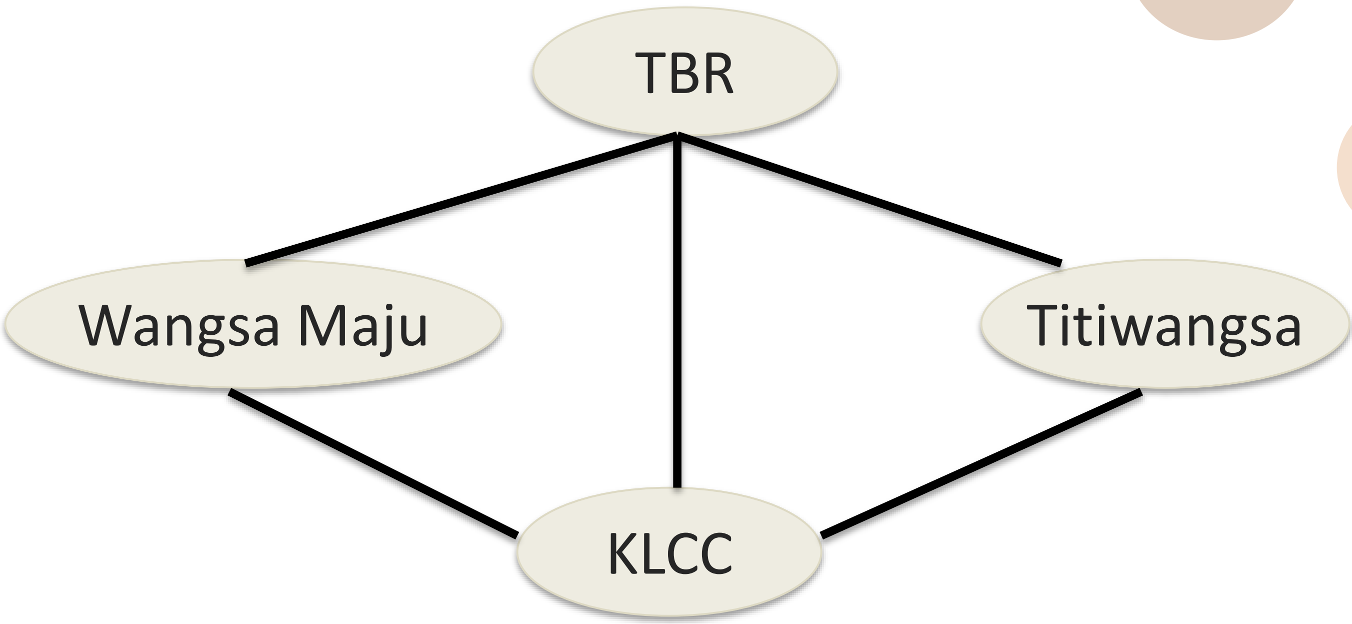
- The same state can be reached from multiple paths.

Search Tree

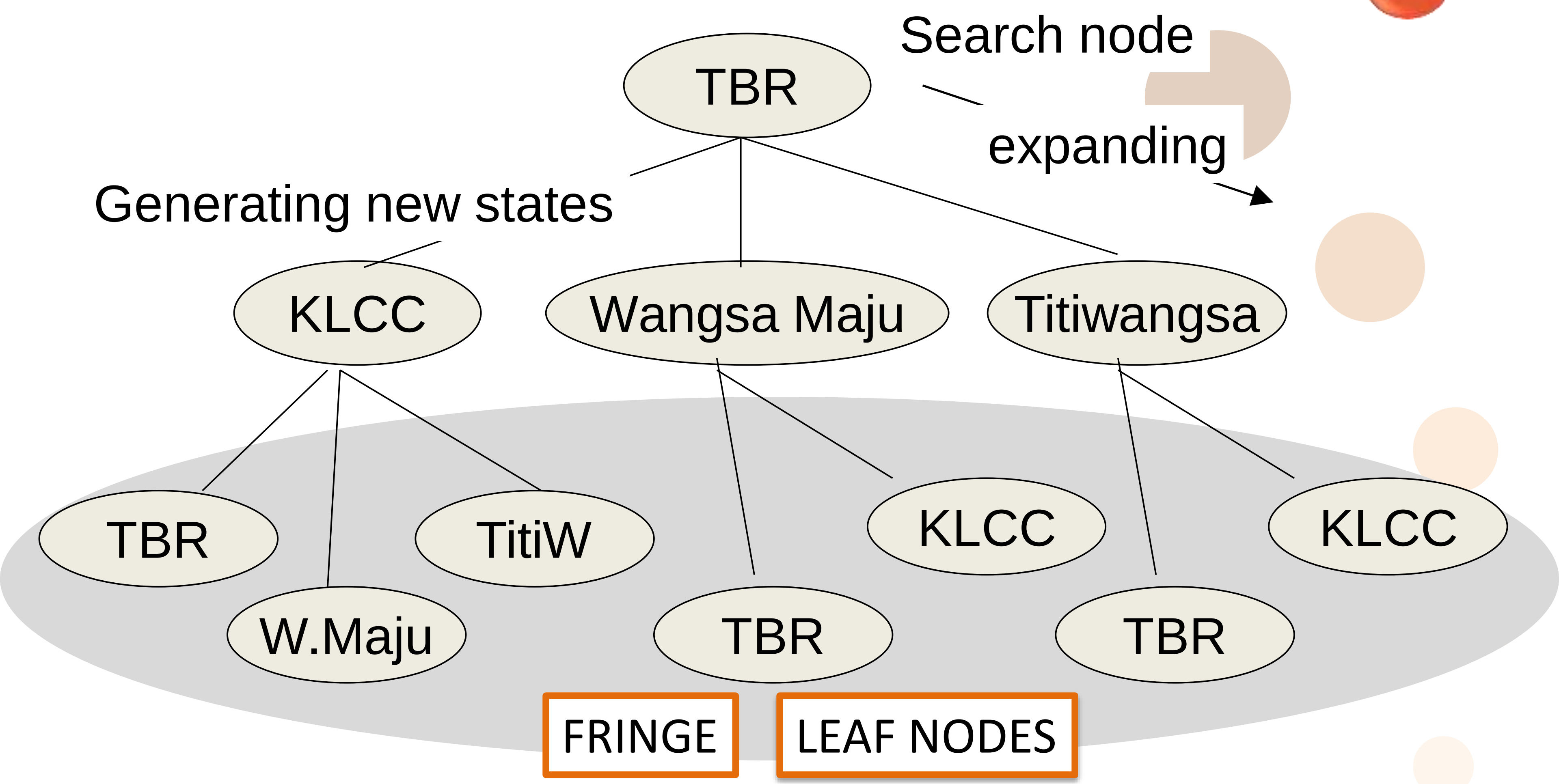


TBR
KLCC
Wangsa Maju
Titiwangsa
TBR
W.Maju
TitiW
TBR
KLCC
TBR
KLCC

Search Graph



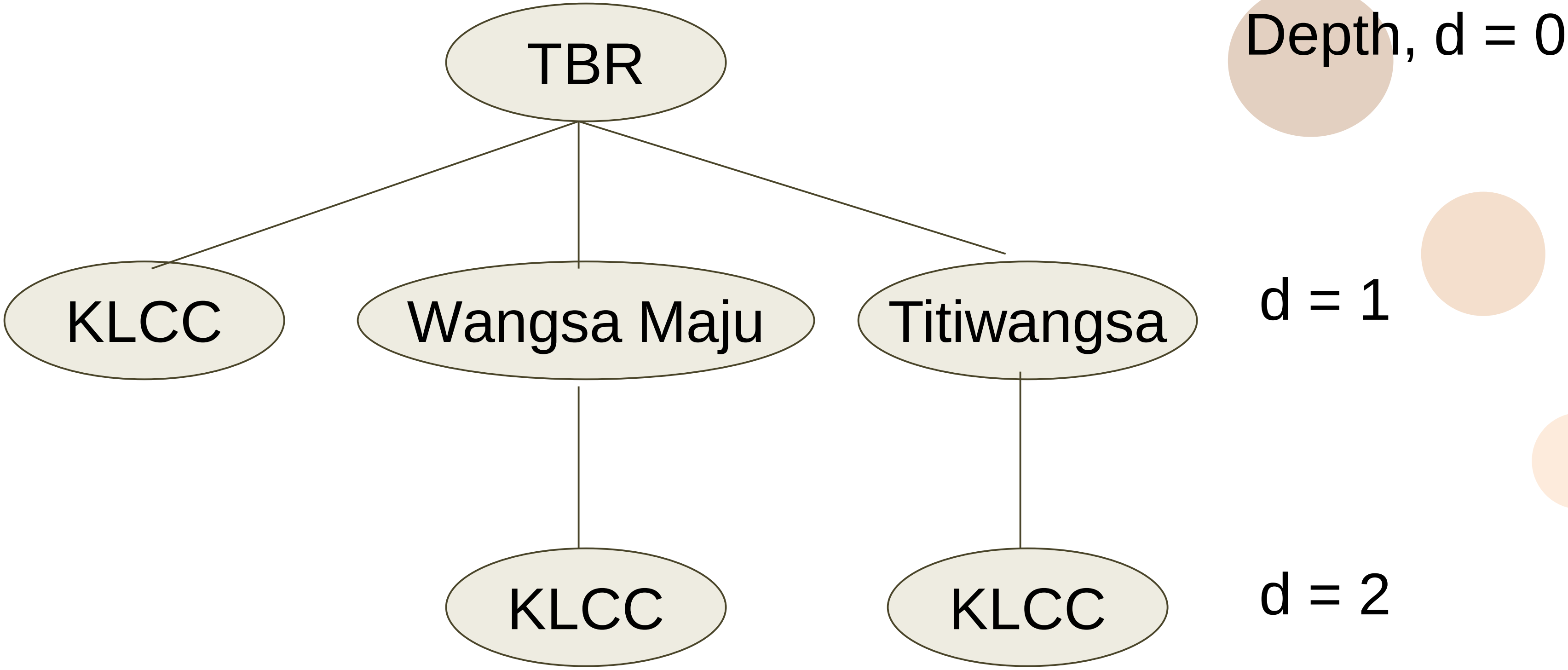
Theories of Search Tree



Avoiding Repeated States

- Wasting time
- Can cause a solvable problem become unsolvable if the algorithm does not detect them
- How to solve?

Search Tree without Repeated State



Measuring problem-solving performance

Completeness

- Is the algorithm guaranteed to find a solution when there is one?

Optimality

- Does the strategy find the optimal solution?

Time complexity

- How long does it take to find a solution?

Space Complexity

- How much memory is needed to perform the search?

Example of searching solution

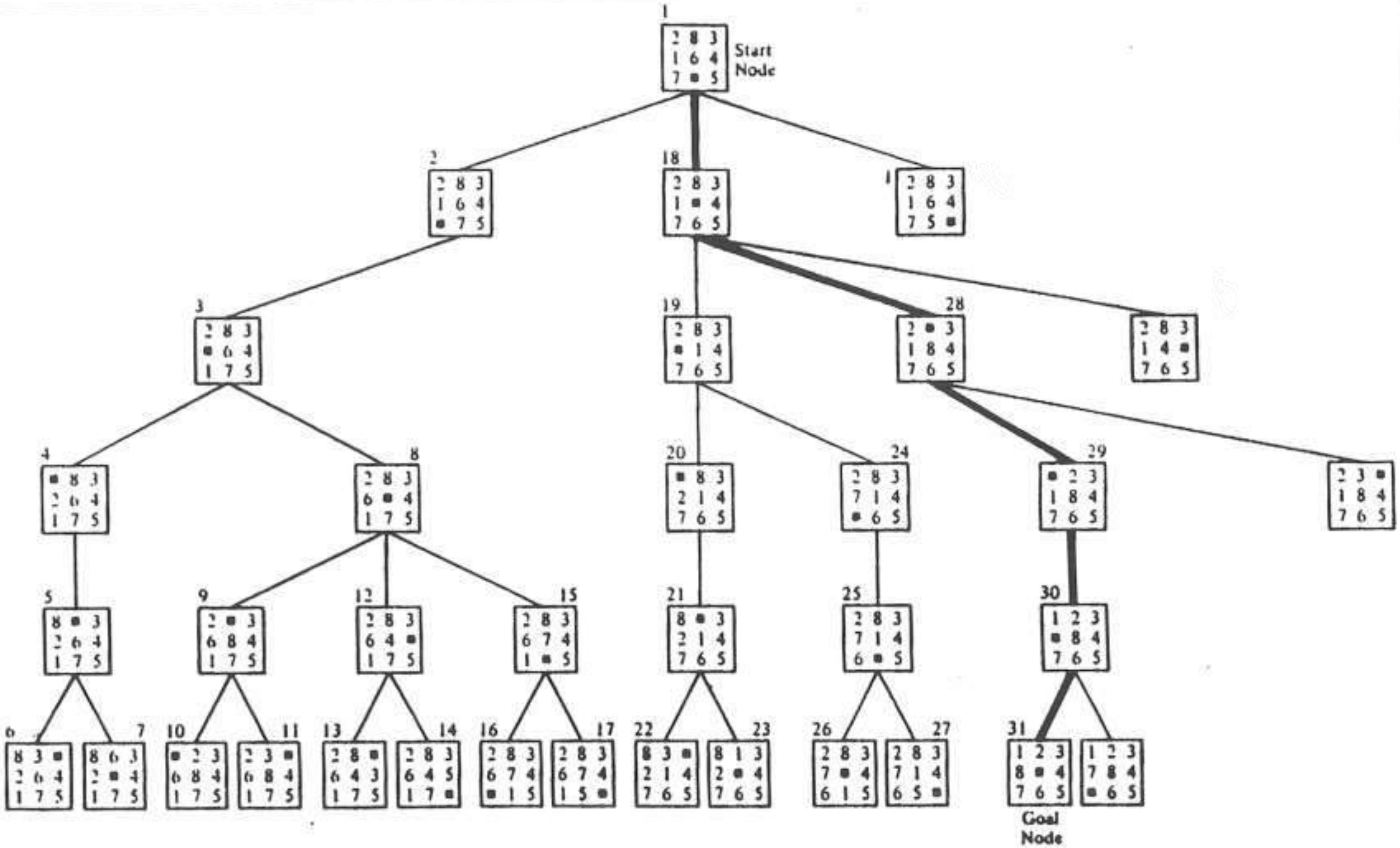


Fig. 2.6 A search tree produced by a depth-first search.

TARC
TUNKU ABDUL RAHMAN UNIVERSITY COLLEGE
BEYOND EDUCATION

Uninform Search

Next Lecture