

Question 1

- อธิบาย ความหมายของคำต่อไปนี้ โดยใช้ตัวอย่างของเครื่องดูดฝุ่นมาอธิบาย
 - State, state space, search tree, search node, goal, action, successor function, and branching factor
- อธิบายองค์ประกอบของ **Problem** และ **State Space Selection** โดยใช้ตัวอย่างของเครื่องดูดฝุ่น (slide 7-8)

Question 2

- ให้อธิบายความหมายของ **Problem solving** ใน slide ที่ 3 และ **formulate problem** และเลือก **state space** ของตัวอย่าง 8 puzzle

function Simple-Problem-Solving(*percept*) returns an action

static: *seq*, an action sequence, initially empty

state, some description of the current world state

goal, a goal, initially null

problem, a problem formulation

state $\leftarrow$ Update-State(*state*, *percept*)

if *seq* is empty then

goal $\leftarrow$ Formulate-Goal(*state*)

problem $\leftarrow$ Formulate-Problem(*state*, *goal*)

seq $\leftarrow$ Search(*problem*)

action $\leftarrow$ Recommendation(*seq*, *state*)

seq $\leftarrow$ Remainder(*seq*, *state*)

return *action*

Question 3

- ให้อธิบายความหมายของ General Search algorithm โดยใช้ตัวอย่างของ Romania

function Search(*problem*, *fringe*) returns a solution, or failure

fringe $\leftarrow$ Insert(Make-Node(Initial-State[*problem*]), *fringe*)

 loop do

 if *fringe* is empty then return failure

node $\leftarrow$ Remove-Front(*fringe*)

 if Goal-Test(*problem*, State(*node*)) then return *node*

fringe $\leftarrow$ InsertAll(Expand(*node*, *problem*), *fringe*)

function Expand(*node*, *problem*) returns a set of nodes

successors $\leftarrow$ the empty set

 for each *action*, *result* in SuccessorFn(*problem*, State[*node*]) do

s $\leftarrow$ a new Node

 Parent-Node[*s*] $\leftarrow$ *node*; Action[*s*] $\leftarrow$ *action*; State[*s*] $\leftarrow$ *result*

 Path-Cost[*s*] $\leftarrow$ Path-Cost[*node*] + Step-Cost(*node*, *action*, *s*)

 Depth[*s*] $\leftarrow$ Depth[*node*] + 1

 add *s* to *successors*

 return *successors*

Question 4

- ให้อธิบายคุณสมบัติของ **BFS** โดยใช้ ตัวอย่างของ **Romania**
 - Completeness หรือไม่อย่างไร
 - Time
 - Space
 - Optimal (what is the cost?)
- อธิบายข้อดีและข้อเสียของ **BFS** เปรียบเทียบกับ **DFS**

Question 5

- ให้อธิบายคุณสมบัติของ DFS โดยใช้ ตัวอย่างของ Romania
 - Completeness
 - Time
 - Space
 - Optimal (what is the cost?)
- อธิบายข้อดีและข้อเสียของ BFS เปรียบเทียบกับ DFS

Question 6

- ให้อธิบายคุณสมบัติของ **Uniform-cost search** โดยใช้ตัวอย่างของ **Romania**
 - Completeness
 - Time
 - Space
 - Optimal (what is the cost?)
- อธิบายว่า **search algorithm** นี้มีความคล้ายหรือแตกต่างจาก **BFS** อย่างไร