NUS National University of Singapore

School of Computing

The Tourist Problem:

(Problem Solving the CS Way)

Video 5.1

Hon Wai Leong

Department of Computer Science
National University of Singapore

Email, FB: leonghw@comp.nus.edu.sg

Experience the fun of problem solving

In this segment sequence…

You experience how to solve problems the computer science way. You will see how we

- Analyze a problem,
- formulate, model, and solve a problem, and
- analyze our solution.

We do these *before* we develop programs/sw

Often iterate this process for better solutions.

We constantly "Ask Questions"!

The *Setting…*

You are interning
in **Q-Tour.com**

Your task:
To solve the
Tourist Bus Scheduling Problem
or Tourist Problem, in short.

The Tourist Problem…

GIVEN: A list of tourists; each tourist has a list of places to visit.

TO DO: Schedule bus trips for them so that each tourist visits all the places in his list.

An Instance of Tourist Problem

Tourist	Places of Interest
Aaron	SZG, BG, JB
Betty	CG, JG, BG
Cathy	VC, SI, OR
David	JG, CG, OR
Evans	CG, JG, SZG
Frances	BG, SZG, JB
Gary	CG, OR
Harry	JG, CG

8 tourists,
8 places

Note:

- **Tourist Bus Scheduling Problem**
  - Tourist Problem, TP
- **An example problem of TP**
  - Called an *instance* of the TP
- **Solution of TP**
  - Must solve *all possible instances* of TP
  - Not just some special cases;

Tourist Problem (Entities)

- **Good to know the entities we are dealing with…**

  The Tourists:

  $T = \{ A, B, C, D, E, F, G, H \}$

  The Attractions (Places):

  $P = \{ BG, CG, JB, JG, OR, SI, VC, SZG \}$

… Places of Attraction …

Place	Common Name	Place	Common Name
BG	Botanical Gardens	CG	Chinese Gardens
JB	Jurong Birdpark	JG	Japanese Gardens
OR	Orchard Road	SI	Sentosa Island
SZG	Spore Zoological Gardens	VC	VivoCity

Tourist Problem (Analysis… 1)

Q: Can we derive any useful insights?

Consider

- Aaron { SZG, BG, JB }
- ~~Frances { SZG, BG, JB }~~

An Instance of Tourist Problem

Tourist	Places of Interest
Aaron	SZG, BG, JB
Betty	CG, JG, BG
Cathy	VC, SI, OR
David	JG, CG, OR
Evans	CG, JG, SZG
~~Frances~~	~~BG, SZG, JB~~
Gary	CG, OR
Harry	JG, CG

Useful Insight:

For our purpose of scheduling bus trips to the 8 places, we can "*safely remove*" Frances.

Why? (in a nutshell)

Frances can just follow Aaron and take all the trips that Aaron takes;

Tourist Problem (Analysis… 2)

Some Simplifications: Consider

- Aaron { SZG, BG, JB }
- ~~Frances { SZG, BG, JB }~~

Also consider

- David { JG, CG, OR }
- ~~Gary { CG, OR }~~

An Instance of Tourist Problem

Tourist	Places of Interest
Aaron	SZG, BG, JB
Betty	CG, JG, BG
Cathy	VC, SI, OR
David	JG, CG, OR
Evans	CG, JG, SZG
~~Frances~~	~~BG, SZG, JB~~
~~Gary~~	~~CG, OR~~
Harry	JG, CG

Cannot safely remove David.

Why?

Can safely remove Gary.

Gary "follow" David, but only for the places in Gary's list.

Tourist Problem (Analysis… 3)

Some Simplifications: Consider

- Aaron { SZG, BG, JB }
- ~~Frances { SZG, BG, JB }~~

Also consider

- David { JG, CG, OR }
- ~~Gary { CG, OR }~~

An Instance of Tourist Problem

Tourist	Places of Interest
Aaron	SZG, BG, JB
Betty	CG, JG, BG
Cathy	VC, SI, OR
David	JG, CG, OR
Evans	CG, JG, SZG
~~Frances~~	~~BG, SZG, JB~~
~~Gary~~	~~CG, OR~~
~~Harry~~	~~JG, CG~~

Simplification Rule:

If $P(T_1) \subseteq P(T_2)$, then tourist T_1 can just "follows" tourist T_2. Thus, we can omit T_1 from consideration.

Oh, can also omit Harry

- Betty { CG, JG, BG }
- ~~Harry { CG, JG }~~

The (Reduced) Tourist Problem…

GIVEN: A list of tourists; each tourist has a list of places to visit.

TO DO: Schedule bus trips for them so that each tourist visits all the places in his list.

An Instance of Tourist Problem

Tourist	Places of Interest
Aaron	SZG, BG, JB
Betty	CG, JG, BG
Cathy	VC, SI, OR
David	JG, CG, OR
Evans	CG, JG, SZG

Every solution to this reduced instance…

… is a solution to the original instance!

$T = \{ A, B, C, D, E \}$

$P = \{ BG, CG, JB, JG, OR, SI, VC, SZG \}$

Quick Summary:

- So far, we defined the Tourist Problem,
- Seen an *instance* of TP and
  defined a *solution* for TP,
- We *analyzed* the instance, and
- Used *Simplification Rule* to reduce it
  to a smaller equivalent problem
- Along the way, we asked lots of question.

(End of video 5.1)

If you want to contact me,

Email: leonghw@comp.nus.edu.sg

School of Computing