Available at the COMP2111 website.

# Assignment 2 COMP2111 13s1
# Traffic Around the Hotel

Kai Engelhardt

Revision: 1.2 of Date: 2013/04/29 23:59:49

This assignment is worth 14 marks and due **Sunday May 12th, 23:59:59** local time Sydney.

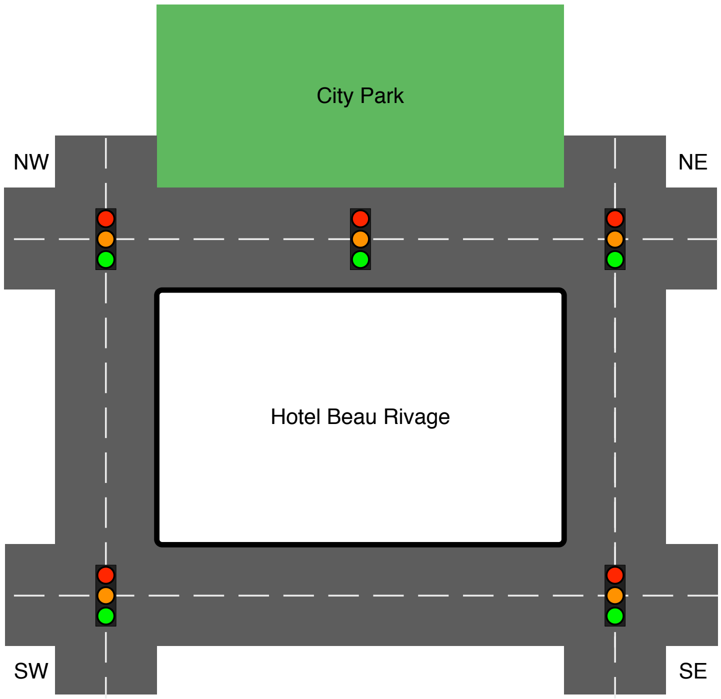


Figure 1: A sketch of the hotel traffic situation. Traffic lights indicate the position of traffic light installations rather than individual lights.

## Introduction

Your local council has been bitten by the "user-pays" bug. The traffic light control system for the streets around the hotel under your management is falling apart and you've been approached to organise a replacement system.

Informed by a short presentation of a traffic light development in COMP2111, you feel confident enough to accept this challenge. You also recall that Ken's book draft has a chapter on traffic lights.

So what is known about the traffic situation around the hotel? The hotel is rectangular and there is a 4-way intersection on every corner. Opposite the main entrance there's the city park. To facilitate access for hotel guests, there's currently a zebra crossing but no traffic lights. Because traffic volume has increased over the years, you have always hoped to be able to replace that crossing with a pedestrian crossing with traffic lights. So the plan is to install five new sets of lights, four at the corner intersections (with road traffic lights and pedestrian traffic lights for all four directions) and one at the pedestrian crossing (with road traffic lights for two directions and pedestrian traffic lights for the other two). See Fig. 1 for a sketch.

Lights are not allowed to be necessarily stuck at one colour. Regardless of what happened in the past, it must always be possible for a light to change colour later. That does not mean that every light must change colour again and again, just that it must be possible, say, in order to allow a car that has pulled up to progress.

Lights on the same crossing are coordinated such that there are always two physical sets of lights that behave exactly the same. For instance, the two pedestrian lights at the park crossing are such a pair and so are the two road lights on that crossing.

Every pedestrian light will be equipped with a push button. Every road traffic light is associated with an induction coil under the tarmac at the white line where cars stop if this light is red.

Additional requirements are based on the notion of *conflicting* lights. Two lights are conflicting if they are located at the same intersection/crossing intersection but not opposite to each other.

**Safety:** Traffic is *safe* around the hotel, that is, no two conflicting lights are green at the same time. In fact, a stronger form is required: before one of set of conflicting lights can turn green, all the others need to be red.

**Fairness:** Traffic management aspires to be *fair* in the sense that, if a light $\ell$ has been requested to become green (by a pedestrian pushing a button or by a car stopping on the induction coil), then no conflicting light can become green twice without $\ell$ becoming green.[1]

**Flow:** If a light $\ell$ has been requested to become green (by a pedestrian pushing a button or by a car stopping on the induction coil), then it will eventually become green.

## Tasks

### Task 1: Requirements Gathering

Gather the requirements in list form in your LaTeX document. Reference these requirements from within Event-B comments in your model.

### Task 2: Abstract Model

Create an Event-B project `HotelTraffic`. Add one or more contexts and machines for an abstract model of the traffic management system.

Do not include the induction coils and push buttons yet. Ignore the fairness and flow requirements. Capture the conflict relation on traffic lights. Ensuring safety is your major concern for now. Reference your requirements from within Event-B comments.

[1] The only exception to this rule is if there is an emergency, but that part of the system is only for bonus marks.

### Task 3: Add Sensors

Refine the abstract model by introducing the induction coils and push buttons to the system. You still can't guarantee fairness and flow.

### Task 4: Add Logic

Refine your model once again to hook up the sensors to the traffic lights. You may require an additional data structure to keep track of recent sensor readings and light changes. Can you capture the fairness and flow requirement as invariants? Can you enforce these requirements?

### Task 5: Documentation

Describe and justify your modelling decisions in a LaTeX document named `HotelTraffic.tex`. Make it concise, convincing, and fun to read for your tutor. Don't include lengthy Event-B listings. Discuss and prove POs you couldn't discharge in Rodin.

### Bonus Task: Emergencies

The city has requested that the emergency response coordinators are given overriding access to control lights in case emergency vehicles need to be routed along one of the 4 roads (and in one of two possible directions). Add such an emergency switch for changing the lights for one of eight possible lanes to green as quickly as possible. In case of an emergency, fairness need no longer be maintained.

## Deliverables

`HotelTraffic.zip` is your Event-B project exported from `Rodin`.

`HotelTraffic.tex` is a LaTeX document with your name or student number mentioned in the `\author` command. It contains your report.

## Submission Instructions

The `give` command to be run is:

```
% 2111
% give cs2111 ass2 HotelTraffic.zip HotelTraffic.tex
```

The command above submits the bare minimum. Should you feel the need to include more files, e.g., for vector diagrams, just list them as well.