# Traffic Light Program Evaluation

Overall we are very pleased with our implementation of the project. We are particularly pleased with:

* Traffic lights respond to 10 cars being queued – turning green to allow them to pass and red after a set amount of time which allows 10 cars to pass through the junction.
* Cars are randomly distributed amongst possible exits of the junction which they are crossing.
* The server can handle any number of traffic lights being connected at any given time.
* Each junction uses a semaphore that forces three lights to remain red at any given point in time. The others lights are queued to turn green based on which notified the server of the 10th car arriving first.
* We used a thread to control each light change so that we could use Sleep() commands to cause delays between light colour changes instead of using our own timer.
* We implemented 2 junctions using semaphores and threads.

Obviously there are going to be parts of the implementation that we are unsatisfied with. Things that we would like to improve are:

* We were unable to use encapsulation to expand to an unlimited number of junctions. This was caused by errors thrown when trying to pass variables to a thread when it is called. This also impacted efficiency as we had two threads constantly instead of creating a single thread for each one required.
* We didn’t implement car animations or a city map as part of the submission. We feel that this was less critical to the assessment criteria than the parts that we did implement.
* We also didn’t implement for the lights to change after a certain amount of time regardless of how few cars are queued. This is something that was brought to our attention as a bit of an afterthought – it would be useful in the middle of the night when there is limited traffic on the road, but obviously during rush hour it would not likely be required.

We think that our solution covers the majority or the specification and are therefore overall very pleased with it.

As far as contribution is concerned, we used the practice of paired programming so we feel that we equally contributed the ideas and implementation of the project.