

# CSc 365 Coursework 2002

This coursework is intended to help you develop an understanding of safety-critical systems. As well as helping to develop your technical understanding of safety-critical systems development, the work will also help you develop transferable skills in written communication, Java programming and will illustrate the practical application and use of formal system specification in Z. The mark for this work will constitute 20% of the final mark for this option.

## Objective

The objective of this coursework is to develop a software simulation of an insulin pump. In your simulation, you have to devise a means of simulating the input from the blood sugar sensor and simulating the exceptional conditions that can arise. Your simulator should work faster than real-time (i.e. you should not wait 10 minutes between sensor readings) and the simulator displays should replicate the displays on the insulin pump system. The design of the simulator user interface is your responsibility.

The specification of the insulin pump control software is given below and your simulator should be developed according to this specification. The system must be developed in Java and must include a graphical simulation of the displays and buttons on the insulin pump as described here. For further background see the notes distributed in class and 'Software Engineering, 5<sup>th</sup> or 6<sup>th</sup> edition'.

## Deliverables

The deliverables from this coursework are:

1. A short user manual for the insulin pump simulation system. This **must** include screen shots showing the simulator user interface and the simulated interface of the insulin pump (6 marks). The manual must also describe how all possible inputs, including exceptional inputs such as battery low are input to the system (2 marks). I would not normally expect this to be any more than 4 pages long.
2. A design description of the insulin pump simulation system. This **must** include:
  - a description of the overall approach that you have adopted to simulate the system (2 marks)
  - a diagram showing the objects in your software system and the interactions between these objects. This may be expressed in either the UML if you are familiar with that notation or some other notation that shows objects and object relationships (5 marks).
  - a description of the algorithm which you use to compute the dose of insulin to be delivered along with a safety argument (not a proof) that demonstrates that your algorithm is safe. Demonstrating safety means that you must show that the system can never deliver a dose of insulin which exceeds the maximum single dose and that the maximum daily dose of insulin cannot be exceeded. This must make explicit reference to Java statements that are part of your code. (5 marks)

Both the user manual and the design description must include an appropriate introduction which describes the purpose of the document and the intended readership. Up to 20% of the total marks may be deducted for poor presentation or layout.

**Written deliverables should be handed in to the Department by 5pm, Friday 3<sup>rd</sup> May 2002. Penalties may be applied to work that is submitted late.**