Xday, XX XXX 2012.

9.30 am - 11.15am (check this!)

University of Glasgow

DEGREES OF BEng, BSc, MA, MA (SOCIAL SCIENCES).

COMPUTING SCIENCE - SINGLE AND COMBINED HONOURS
ELECTRONIC AND SOFTWARE ENGINEERING - HONOURS
SOFTWARE ENGINEERING - HONOURS

SAFETY-CRITICAL SYSTEMS DEVELOPMENT

Answer 3 of the 4 questions.

1.

a) Several major car manufacturers have recently been forced to recall their vehicles following software related failures involving Anti-lock braking systems (ABS) and cruise control applications.

Briefly explain why it is so hard for companies to guarantee the safety of their software before it is released onto the market.

[5 marks]

[seen/unseen problem]

In the course we have argued that it is impossible to achieve absolute safety – one reason for this is that safety is an emergent property – the ways in which we use safety devices (including software systems) can change once they are deployed for longer periods of time (up to 2 marks). For instance, risk homeostasis theories have suggested that drivers will trade speed for safety when in vehicles equipped with ABS (1 mark). There are also pragmatic issues associated with ensuring that millions of lines of code do not contain unexpected bugs or that problems may not have occurred in requirements prior to the deployment of software in mass market applications like cars (2 marks). The problems of software testing (Dijkstra's maxim that testing establishes the presence but not the absence of bugs) also make it impossible to guarantee that no bugs exist (1 mark). Finally, some students may talk about process based certification and regulation rather than produce based testing – and the consequent limitations that this can impose (1 mark).

b) Earlier this year, a warning was issued in the United States about a complex software problem that led to a vehicle recall:

> XXXX IS RECALLING CERTAIN TRUCKS. THESE VEHICLES WERE INSPECTED USING AN INTEGRATED DIAGNOSTIC SYSTEM (IDS) THAT HAD A CUSTOM SOFTWARE ROUTINE TO READ THE SUSPECT BODY CONTROL MODULE (BCM) SERIAL NUMBER. BASED ON THE SERIAL NUMBER THE BCM WAS EITHER NOT AFFECTED OR REPLACED. THE CUSTOM SOFTWARE ROUTINE WAS NOT READING THE CORRECT SET OF CHARACTERS, AND WAS NOT ABLE TO IDENTIFY A BCM THAT REQUIRED REPLACEMENT. THE AFFECTED BCMS MAY HAVE THE POTENTIAL FOR AN INTERNAL SHORT (THAT COULD RESULT IN A VEHICLE FIRE).

Briefly explain how a risk-based approach to safety-critical software development might have been used to identify the hazards associated with the failure of the IDS.

[5 marks]

[unseen problem]

This question is intended to determine whether or not students can apply concepts from the course to an unseen problem. An initial problem with the body control module was identified and then the IDS was used by engineers to determine whether or not the BCM had to be replaced. However, a fault in the IDS meant that engineers were not always correctly warned that a BCM should be replaced (1 mark for restating the problem from the software engineering report). A risk based approach might have identified the hazard that an IDS bug would fail to accurately identify/diagnose a faulty module in a car (1 mark). The consequences of this would be those associated with the failure mode that the IDS was being used to identify, which could be severe (2 marks). The probability of the IDS containing a bug is difficult to assess – however, as a secondary system (ie not a direct control application) it may not have been developed to the most rigorous standards – hence with the benefit of hindsight it is a potential area of concern for the future development of vehicle diagnosis systems (2 marks).

c) Last year, the media raised concerns that a major vehicle manufacturer had a software problem with their ABS. Groups within NASA were commissioned to determine whether or not such a problem existed. They were unable to replicate the reported bugs and instead chose to focus on driver error and problems with the floor mats inside the vehicle.

Why is it so hard to identify the causes of intermittent software failures? Your answer should refer both to the likelihood of failure and also the risk exposure for mass-market applications.

[10 marks]

[seen/unseen problem]

This is similar to the previous question by providing an unseen problem scenario that students will then have to apply the concepts from the course. Many software failures are intermittent. In other words, the complexity of run-time execution patterns means that a bug may only rarely reveal itself through the combination of many different input parameters within a particular state of the system (2 marks). Small differences in those input parameters may mean that the erroneous code will not be executed even though the state of the system looks to be identical to the one in which the bug was observed (2 marks). Intermittent faults are particularly hard to diagnose because there is no guarantee that they will not recur even though it has proven to be impossible to identify the cause amongst many hundreds of thousands of lines of code (2 marks). These comments apply to application software but also to operating systems and network protocols (such as Flexray and CAN in automotive applications) (1 mark). In this case study, NASA could not recreate the errors reported by drivers. However, this may have been due to the complexity of the code and the task of recreating intermittent failures. A key issue here is that mass market applications have an extremely high risk exposure – bugs that are rarely executed WILL occur because of the many millions of hours that the code will be executed in these vehicles (2 marks). External auditors cannot easily run the exhaustive tests that would be needed to replicate this level of risk exposure in order to determine whether or not a bug exists (2 marks). None of this need undermine the arguments about driver error or about the mat problems – however, these may have occurred in addition to undiagnosed software bugs – hence great care must be taken in the audit techniques used by external agencies (2 marks).

2.

a) The IEC 61508 standard focuses on the hazards that are associated with equipment under control (EUC). Briefly explain how this assessment is used to derive the Safety Integrity Level (SIL) of software within a safety-critical application.

[5 marks]

[seen problem]

This has been explained several times in the lecture. It is difficult to assess the probability of failure in software modules – if code contains a bug then the probability of it containing a bug is 1 – in other words the stochastic assessment of hardware reliability cannot be directly used to inform safety-critical software development (2 marks). Instead, the likelihood and consequences of hazards associated with the Equipment Under Control are assessed (1 mark). Risk matrices can then be used to assess whether the product of this calculation is broadly acceptable (see the answer the 2c) (2 marks). If a risk is unacceptable then software can be used as a means of mitigation (1 mark). The SIL can be thought of as a measure of the necessary risk reduction, if the code provides protection/mitigation against a significant risk then it should be developed with greater care than if it only protects against a marginal risk (2 marks). The higher the SIL then the greater will be the costs of the associated development processes (1 mark).

b) A number of criticisms have been raised about the application of SILs within international standards. These include the following:

- It is difficult to harmonize the use of SILs within and between industries;
- It is difficult to ensure the consistent use of SILs across different international standards;
- The use of SILs often drives the application of process rather than product based metrics;
- SILs are typically derived from expert judgments and reliability estimates that are difficult to validate.

Write a brief technical report for a software development company explaining what steps they might take to address some of these criticisms within their own engineering practices.

[10 marks]

[unseen problem]

This provides students with the opportunity to critically assess leading standards for software development. I have provided the items in the list as a starting point so everyone should be able to attempt an answer. However, more able students should have the scope to propose more innovative counter-measures.

From the perspective of a company, there may be less concern to ensure the harmonization of SILs within and between industries. Even so, they must demonstrate that they are following 'leading' practices and it is usual to recruit external auditors to benchmark their application of the standards (2 marks). Participation in industry for a and even in standards development activities can help – also close communication with technical and domain specialists within the Health and Safety Executive can lend additional confidence (2 marks). Again, the criticism that it is difficult to ensure the consistent use of SILs across different international standards may not be so relevant for some companies. However, in the course we have described how some organizations have responded to this challenge by writing code that is reused in different market places and certified against different standards (2 marks) – including several VOIP companies where the same code is used for military command and control, ATM, Fire and rescue services etc. In this case, internal company procedures are often used to cross-check SIL determination between projects to ensure that comparable approaches are used between standards and domains (2 marks).

The criticism that the use of SILs often drives the application of process rather than product based metrics is more germane to the company. In particular, it is important that senior management understand the application of this approach does not directly use testing to demonstrate the absence of bugs (in other words safety does not simply rst on the testing of the product, the code), it is also ensured through the application of appropriate development processes (2 marks). Hence attempts to cut costs by limiting those processes will have a direct consequence both on the likelihood of certification and on safety (1 mark). The final caveat is that SILs are typically derived from expert judgments and reliability estimates that are difficult to validate. Again external auditors can be used to confirm the SIL assessments and the risk analysis conducted on EUC. There are also more rigorous empirical techniques such as those developed by Ben Ale's group at the University of Delft, that can be applied to probe expert judgments of technical risks – however, as yet these have not been widely applied in industry (2 marks).

c) In the UK, Health and Safety legislation is drafted to ensure risks are As Low As Reasonably Practicable (ALARP). The US equivalent is known as the As Low As Reasonably Achievable (ALARA) principle. Other countries use Minimum Endogenous Mortality (MEM) to guide risk-based decision making. MEM requires that organizations do not introduce hazards, which will 'significantly increase' the death rate beyond that expected from disease, congenital mortality etc.

Summarize the problems that arise when these approaches are used to guide the engineering of safety-critical software.

[5 marks]

[unseen problem]

Previous sections have described how we must often reduce the risks associated with Equipment Under Control to an 'acceptable' level – for instance, using software control systems. The degree of risk reduction to achieve this acceptable level is used to determine the SIL and this in turn determines the development processes used to design, implement, validate and verify the code. However, this relies on largely subjective judgments about acceptable levels of risk (2 marks). ALARP and ALARA provide a framework for this judgment using the legal concept of reasonable care (1 mark). Numeric approaches can support this assessment – for instance by specifying ranges for the probability of catastrophic hazards – however, the very low numbers used here make them difficult to validate (2 marks). In contrast, MEM attempts to quantify acceptability in terms of epidemiology. One problem with this is that it is a retrospective measure – the increase in morbidity from introducing a new system might only be seen after it has been put into operation (2 marks).

3.

a) In November 2010, Space Exploration Technologies (Space X) was granted the first Federal Aviation Administration license allowing the reentry to Earth of a privately developed spacecraft. The FAA conducted a review of 1) the public safety issues; 2) the environmental impact; 3) the potential payloads; 4) national security and foreign policy concerns, and 5) insurance.

Identify the technical and managerial problems that can arise when assessing the adequacy of a commercial organization's ability to meet each of the five requirements considered in the FAA's review.

[5 marks]

[unseen problem]
This is an open-ended question asking about general safety concepts before focusing in part b) on software issues. In general terms, it is difficult for external regulators to understand the detailed technical and engineering processes used by many 'leading edge' organizations. It is difficult for state-funded regulators to recruit and pay the best engineers who might otherwise be recruited into high technology firms such as Space X (2 marks). Further problems stem from the need to understand the detailed techniques proposed by companies while at the same time respecting commercial confidentiality and a need for national security (2 marks). This is a particularly sensitive area given the lack of previous commercial space operations of this nature. Clearly, information about the levels of insurance and premiums paid by a company would be of significant interest to competitors as would details about the potential payloads that might be supported by these operations (1 mark). In other areas, such as the assessment of the environmental impact, there are problems associated with the identification of 'all' potential hazard scenarios (1 mark). It can be difficult in this specific case for the FAA to consider all aspects of public safety given that the risks could affect people in other countries during a Space X flight for which the FAA arguably does not have appropriate jurisdiction (1 mark).

b) The FAA certification described in part a) was required before tests could be conducted to determine whether or not it was safe for Space X's Dragon capsule to carry crew to the International Space Station (ISS).

Summarize the problems that arise when government agencies must assess the adequacy of tests that are conducted by commercial agencies meeting service oriented safety software requirements rather than delivering a product, such as a capsule or launch vehicle.

[5 marks]

[seen problem]
In the past, many government agencies used a product acquisition cycle to buy critical equipment that they then operated (1 mark). For example, NASA used to issue contracts for shuttle components that were then directly assembled under the supervision of NASA engineers. In terms of software engineering, this led to companies such as Lockhead Martin being tasked to develop code for detailed requirements that were identified by NASA. However, many governments now prefer a service oriented model where fewer constraints are imposed on systems components and architectures (1 mark). Instead, companies are paid to deliver a service with greater freedom on the systems that they will use to do this (2 marks). Hence, the customers may not be able to specify detailed requirements for the software that is used to deliver such a service. For regulators this new way of working creates a host of problems. They must often rely of 'black box' tests to assess the safety and reliability of the systems that must work together to implement the desired service (2 marks). If white box testing is allowed then this creates considerable concerns that government agencies are interfering with commercial rather than technical decisions. At present, there are few (no?) generally accepted techniques for resolving these tensions in the software engineering of complex, safety-critical systems (2 marks).

c) The first flight of the Dragon to the ISS was scheduled for January 2012. This was delayed because the control software had to be delivered to NASA so that the algorithms could be validated. NASA's Commercial Orbital Transportation Services teams also needed copies of the code so that they could use Monte Carlo techniques in software verification.

Explain how program delays impose particular pressures for 'white box' verification and the validation of complex, safety-critical software systems. Identify techniques that can be used to reduce the impact of these pressures.

[10 marks]

[unseen problem]

In many commercial, safety-related projects there are considerable pressures to begin operations – usually because of the significant costs of developing such systems (1 mark). It is important that these pressures do not undermine safe and successful operation. Delays increase the pressures on technical teams and can undermine the arguments for conducting necessary tests (1 mark). 'White box' verification assumes that the code is available for inspection (1 mark). This can lead to delays because it can be important to complete the installed version of the software prior to verification (1 mark) – otherwise, partial verification raises significant technical concerns for integration and for configuration management prior to launch (2 marks) – to insure that the tests accurately reflect the code as it is installed (1 mark). Time pressures can lead to code being used that does not meet all appropriate tests (1 mark). Validation can also be undermined by program delays because there is a temptation to state that the delivered code is "good enough" even if some significant requirements are not satisfied or are deliberately overlooked (2 marks).

External, independent review boards can help to ensure that verification and validation are not undermined by project delays and the pressure to begin operation (1 mark). Their work can be supported by adequate contingency plans that consider the potential impact of delays early in the development cycle (1 mark). Other techniques include the use of configuration management tools and compositional testing techniques so that elements of the code can be verified prior to full integration testing (2 marks).

4. Brazil, the Russian Federation, India and China have become known as the 'BRIC' nations. They are characterized by emerging and fast growing economies. However, their safety record arguably lags behind other nations'.

   You have been asked to write a brief technical report presenting a strategy for the introduction of software systems safety concerns into the development practices of a company that develops safety-critical software in a BRIC nation.

   (Hint: you should illustrate your answer by referring to previous accidents. Reference should also be made to the application of risk assessment techniques and to the study of human factors in safety critical systems).

   [20 marks]

   [essay]

   This is an open-ended essay – I will tell the class that there is a question on safety-critical software development in 'emerging' or rapidly developing economies. Many different answers are possible. One line of argument might be to extend Perrow's work on Normal Accidents – this suggests that the complexity and integration of new technologies will increase the likelihood of accidents (2 marks). This argument might be applied to nations and not simply to individual industries. The pressures for economic progress and innovation exceed the more cautious constraints imposed in more established industries (2 marks). . The counter-argument provided by Sagan's work on High Reliability Organizations has also been mentioned in lectures. This suggests that it is still possible for 'leading practices' to be introduced in innovative organizations so that safety is maintained (2 marks). .

   I would, however, expect most of the answer to focus on the needs of the software development company rather than more general observations about the BRIC nations (2 marks). . In particular, I would focus on the need for training and competence requirements. This is a particular problem in emerging economies were rapid development means that skilled engineers are scarce and demand is high. One approach might be to develop safety-critical software courses such as this one within local Universities – this is already happening in China (2 marks). .

   Links could be made to many of the previous questions – for instance, the use of in-house development techniques, for verification and validation processes that consider non-functional as well as functional requirements (2 marks). Answers might also focus on the possible adoption of development standards such as 61508 even though their own national regulator may not demand their use at present (2 marks). Some comments might also be made about the need to train the regulators or even identify potential regulatory organizations as the market place develops – this may be necessary to maintain the economic competitiveness of the company when their safety improvements might lead to them being undercut by competitor organizations (2 marks).

   Many different solutions are possible and marks will reflect both the ability to use technical material from the course (61508, D)-178B, safety cases etc) as well as the ability to write a coherent technical report (up to 5 marks for style and argument).