This is an open book exam – you may use all material in the form of paper, such as books, lecture slides, and own (manual written) notes.

Electronic devices are not permitted! Success!

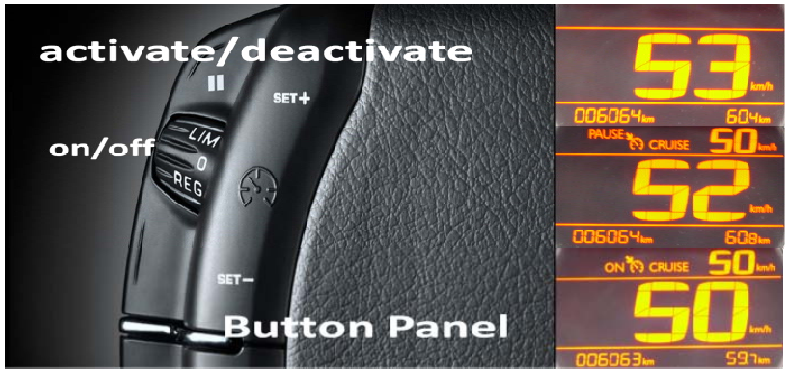
Dit is een open boek tentamen – je mag alle materiaal in de vorm van papier gebruiken, zoals boeken, college dia's, en eigen (handmatig gemaakte) notities.

Elektronische gereedschappen zijn niet toegestaan! Succes!

PART I – Scenario [25/50 pts]

Imagine you work as software engineer in the automotive industry. Your task is the design of a cruise control system (CCS). Once set to a specific set speed, the CCS is supposed to control the throttle of the vehicle such that it maintains the set speed as current speed. The CCS has a user interface with a button panel and a display panel. The button panel is supposed to provide

- An ON/OFF switch for switching the CCS on or off, respectively.
- A PAUSE button for enabling/disabling the speed control if a speed is set.
- A SET+ button for setting the speed to be controlled and increasing it.
- A SET- button for setting the speed to be controlled and decreasing it.



The display panel is supposed to show

- The current speed of the vehicle, if the CCS is switched off.
- The current speed, the set speed, and "CC Pause", if the CCS is switched on but inactive (ON, but not controlling the speed).
- The current speed, the set speed, and "CC ON", if the CCS is switched on and active (ON, and controlling the speed).

The user of the CCS (the driver) can switch the system ON in order to have the current speed of the vehicle controlled through the throttle. Once switched ON, the CCS should display "*CC Pause*", and the system should be

Exam/Tentamen IN2705-T Software Engineering 27.06.11; 14.00-17.00
DTC-Tentamenzaal 2 Gebouw 65 Kluyverweg 4 + 6

idle and waiting for a speed to be set. The requested speed can be set through pressing either of the buttons *SET+* or *SET-*. Pressing these buttons should set the speed of the CCS to the current speed of the vehicle and start controlling the throttle. The CCS should then continuously monitor the current speed of the vehicle through accessing *getCurrentSpeed()* of the motor control system. It should maintain the throttle through continuously calling an internal control function *controlThrottle*() that should adjust the current throttle of the motor control system through *setThrottle(throttle)* to match the set speed stored in the CCS. Once the CCS is active and controlling the throttle, pressing SET+ will increase and pressing SET- (for less than 2 seconds) will decrease the set speed of the CCS by 1 Km/h, respectively. Pressing SET+ or SET- for more than 2s should lead to increase or decrease of the set speed of 5 Km/h, respectively. For safety reasons, the CCS should be deactivated if the brake or clutch are pressed, or if the gear is shifted. It should not be possible to activate the CCS if the current speed of the vehicle is less than 40 Km/h or to set a speed below this threshold.

Question/Assignment 1 [3 pts]

Prepare a domain glossary for the scenario described above. Classify the domain concepts into concepts for structure and concepts for behaviour.

Maak een domein glossarium voor de bovenstaande scenario. Deel de domein concepten in naar concepten voor structuur en voor gedraag.

Question/Assignment 2 [5 pts]

Draw a use case diagram for the scenario described above.

Teken een use case diagram voor het bovenstaande scenario.

Question/Assignment 3 [7 pts]

Draw a UML structural diagram for the scenario, representing the CCS in its context.

Teken een UML structuur diagram die het CCS in zijn context weer geeft.

Question/Assignment 4 [8 pts]

Draw a suitable UML behaviour diagram that represents the behaviour described in the scenario.

Teken een geschikte UML gedraag diagram die het gedraag in het scenario goed weer geeft.

Question/Assignment 5 [2 pts]

Write down 2 questions (maximum) of missing information, missing functionality, missing or wrong requirement, etc., that you would like to present to the "product owner."

Schrijf 2 vragen op (maximum) van ontbrekend informatie, ontbrekend functionaliteit, ontbrekend of foute eisen, etc., die je tegen de klant (product owner) zou willen stellen.

Part II – Open Questions [25/50 pts]

Question 6 [1 pt]

Why is *manufacturing* cost negligible in software development? Provide 1 argument!

Waarom zijn de *productie*kosten verwaarloosbaar in software ontwikkeling? Geef 1 argument!

Question 7 [1 pt]

The IEEE defines Software Engineering as application of quantifiable approaches. What is meant by quantifiable? Provide 1 argument!

De IEEE geeft als definitie voor Software Engineering: applicatie van kwantificeerbare aanpak. Wat betekend kwantificeerbaar? Geef 1 argument!

Question 8 [2 pts]

How does a Software Lifecycle Model help in software development? Give 2 Examples!

Hoe helpt een Software Lifecycle Model in software ontwikkeling? Geef 2 voorbelden!

Question 9 [1 pt]

Why can't we use test cases in verification? Provide 1 argument!

Waarom kunnen wij test cases in verificatie niet gebruiken? Geef 1 argument!

Question 10 [1 pt]

Why should you not discuss the class hierarchy in requirements elicitation? Provide 1 argument!

Waarom zou je klassen hiërarchien niet in requirements elicitatie bespreken? Geef 1 argument!

Question 11 [1 pt]

Why is a measure on nominal scale not useful? Provide 1 argument!

Waarom is een meeteenheid op nominale schaal niet bruikbaar? Geef 1 argument!

Question 12 [1 pt]

How does the Single Responsibility Principle (SRP) facilitate cohesion? Provide 1 argument!

Hoe vergemakkelijkt het Single Responsibility Principle (SRP) cohesion? Geef 1 argument!

Question 13 [3 pt]

Discuss the tradeoffs Natural Language Requirements (NLR) vs. Formal Requirements (FR). Provide 3 arguments!

Bespreek de trade-offs (wissel-werkingen) Natuurlijke Taal Eisen (NTE) vs. Formele Eisen (FE). Geef 3 argumenten!

Exam/Tentamen **IN2705-T Software Engineering** **27.06.11; 14.00-17.00**
DTC-Tentamenzaal 2 **Gebouw 65** **Kluyverweg 4 + 6**

Question 14 [3 pts]

Which key performance indicators would you install in your organization in order to measure productivity of people? Give 3 example indicators, and motivate your choice!

Welke key performance meeteenheden zou je in jouw organisatie willen installeren om productiviteit van mensen te meten? Geef 3 argumenten, en motiveer je keuze!

Question 15 [3 pts]

How can usability, a subjective concept, be measured objectively? Provide 3 direct counting metrics!

Hoe kan usability (gebruikbaar-heid), een subjectief concept, objectief worden gemeten? Geef 3 directe telling meeteenheden!

Question 16 [1 pt]

Why can structural code metrics not be used to express cognitive complexity? Provide 1 argument!

Waarom kunnen structurele code meet-eenheden niet gebruikt worden om cognitieve complexiteit uit te drukken? Geef 1 argument!

Question 17 [1 pt]

How/why can coupling be used to express reusability? Provide 1 argument!

Hoe/waarom kan coupling gebruikt worden om reusability uit te drukken? Geef 1 argument!

Question 18 [1 pt]

Why are classes inherently cohesive? Provide 1 argument!

Waarom zijn klassen inherent cohesive? Geef 1 argument!

Question 19 [2 pt]

Scrum and Extreme Programming are both agile methods, but what are their differences? Provide 2 arguments.

Scrum en Extreme Programming zijn allebij agile methoden, maar wat zijn hun verschillen? Geef 2 argumenten!

Question 20 [3 pts]

Cyclomatic Complexity is an indicator for testability. How would you validate empirically that this mapping holds? Give a list of steps (with concrete examples for cyclomatic complexity, following an axiomatic approach) that you would follow, in order to validate this hypothesis!

Cyclomatic Complexity is een indicator voor testability. Hoe zou je empirisch valideren dat deze mapping waar is? Geef een stappenlijst (met concreet voorbelden voor cyclomatic complexity, volgens een axiomatic benadering) die je zou volgen, om deze hypothese te valideren!