

THIS EXAM IS NOT FOR KIE -STUDENTS!!!

**T-76.601, Introduction to Software Engineering
Final Exam (Augmented version), 21.12.2004**

*T-76.601, Ohjelmistotuotannon perusteet
Loppuentti (Laajennettu versio), 21.12.2004*

*T-76.601, Grunderna i Programvaruproduktion
Sluttentamen (Utökad version), 21.12.2004*

THIS EXAM IS NOT FOR KIE -STUDENTS!!!



21.12.2004

Instructions:

- Write your name, student number, degree program and signature in the reserved space below
- Write your name and student number at the bottom of *each sheet*
- Answer the questions in the spaces provided on the question sheets. If you run out of room for an answer, continue on the back of the page.
- You can answer in English, Finnish or Swedish.

Ohjeita:

- Kirjoita nimesi, opiskelijanumerosi, koulutusohjelmasi sekä allekirjoituksesi alla olevaan tilaan
- Kirjoita nimesi ja opiskelijanumerosi jokaisen paperin alareunaan
- Vastaa kysymyksiin koepaperissa varattuun tilaan. Jos tarvitset lisätilaa, kirjoita paperin takapuolelle.
- Voit vastata englanniksi, suomeksi tai ruotsiksi.

Instruktioner:

- Skriv ditt namn, studentnummer, utbildningsprogram samt underskrift i det reserverade utrymmet nedan
- Skriv ditt namn och studentnummer nere på varje ark
- Besvara frågorna i utrymmet på provpappren. Om du behöver mera utrymme kan du skriva på arkets baksida.
- Du kan svara på engelska, svenska eller finska.

Question:	1	2	3	4	5	Total
Points:	10	6	12	6	26	60
Score:						

Name/Nimi/Namn: _____

Student id/Opiskelijanumero/Studentnummer: _____

Degree program/Koulutusohjelma/Utbildningsprogram: _____

Signature/Allekirjoitus/Underskrift: _____

1. Define (or explain briefly) the following terms. *Määrittele (tai selitä lyhesti) seuraavat termit. Definiera (eller förklara kort) följande termer.*

(a) Software architecture. *Ohjelmistoarkkitehtuuri. Programarkitektur.*

(1p)

(b) Time-pacing. *Ajallinen rytmittäminen. Time pacing.*

(1p)

(c) Test oracle. *Testioraakkeli. Testorakel.*

(1p)

(d) Test level. *Testitaso. Testnivå.*

(1p)

(e) Conceptual design. *Konseptuaalinen suunnittelu. Konceptuell planering.*

(1p)

(f) Refactoring. *Refaktorointi. Refaktorering.*

(1p)

(g) Software engineering. *Ohjelmistotuotanto. Programvaruproduktion.*

(1p)

(h) Inspection. *Tarkastus. Inspektion.*

(1p)

Name/Nimi/Namn: _____

Student number/Opiskelijanumero/Studentnummer: _____

(i) SW-CMM.

(1p)

(j) Black-box testing. *Mustalaatikkotestaus. Svartlådetestning.*

(1p)

Name/Nimi/Namn: _____

Student number/Opiskelijanumero/Studentnummer: _____

2. Tomas Nyström presented a breakdown of project management into leadership, management, and administration. Present the activities related to each of these subareas. What was the main point he tried to make with this breakdown? Do you think he was right? Why?

Tomas Nyström esitteli projektinhallinnan jaottelun johtamiseen (leadership), ohjaamiseen (management) ja hallinnointiin (administration). Esitele näihin osa-alueisiin liittyvät aktiviteetit. Mikä oli Tomaksen jaottelun keskeisin ajatus? Oliko hän mielestäsi oikeassa? Miksi?

Tomas Nyström presenterade en uppdelning av projektstyrning i ledarskap, styrning (management) och administration. Presentera aktiviteterna relaterade till dessa delområden. Vad var det Tomas ville säga med indelningen? Tror du att han har rätt? Varför?

(6p)

Name/Nimi/Namn: _____

Student number/Opiskelijanumero/Studentnummer: _____

5. Are the following statements true or false? Mark a **T** for true statements, and an **F** for false ones. If you don't know, leave the row empty. You will get +1 p for a correct answer, 0p for an empty answer, and -1 p for an incorrect answer. The maximum score for this question is 12, and the minimum 0, i.e., you will not get a negative score even if you have more incorrect than correct answers.

Ovatko seuraavat väittämät oikeita vai vääriä? Kirjoita O oikean väittämän ja V väärän väittämän jälkeen. Jos et tiedä, jätä vastaamatta. Oikeasta vastauksesta saat +1 p, tyhjästä 0p ja väärästä -1 p. Tehtävän maksimipistemäärä on 12 ja minimi 0, eli et saa negatiivista pistemäärää vaikka sinulla olisi enemmän vääriä kuin oikeita vastauksia.

Är följande påståenden sanna eller falska? Skriv S efter de som är sanna och F efter de som är falska. Om du inte vet, lämna tomt. Du får +1 p för korrekt svar, -1 p för fel svar, och 0p för tom lucka. Maximipöängen för denna uppgift är 12, och minimi 0. Du kan alltså inte få negativa poäng fastän du skulle ha flera inkorrekta än korrekta svar.

- (a) You cannot release software if it has known bugs.

Et voi julkaista ohjelmistoa, jossa on tunnettuja virheitä (bugeja).

Du kan inte släppa ut ett program på marknaden om det har kända fel (buggar).

(a) _____

(1p)

- (b) One of the main roles of software architecture is to provide an overview of the system — to give the big picture.

Ohjelmistoarkkitehtuurin tärkeimpiä rooleja on antaa yleiskuva järjestelmästä — kokonaiskuvan antaminen.

En av mjukvaruarkitekturens främsta roller är att ge en överblick av systemet — att ge en helhetsbild.

(b) _____

(1p)

- (c) Design patterns are generic solutions to particular common problems in program design and implementation.

Suunnittelumallit (design patterns) ovat yleisiä ratkaisuja tiettyihin yleisiin ongelmiin ohjelmiston suunnittelussa ja toteutuksessa.

Designmodeller (design patterns) är generella lösningar på specifika och vanliga problem vid planering och implementering av program.

(c) _____

(1p)

- (d) "Code smells" can be used to objectively determine the quality of a piece of code.

"Koodin hajuja" voi käyttää ohjelmiston osan laadun objektiiviseen arviointiin.

"Kodlukter" kan användas för att objektivt bestämma kvaliteten av ett stycke programkod.

(d) _____

(1p)

- (e) In pair programming, there are two roles: the driver and the navigator. These roles are assigned at the beginning of the project and not changed for the duration of the project.

Pariohjelmoinnissa on kaksi roolia: kuljettaja ja navigaattori. Roolit jaetaan projektin alussa eikä niitä sen jälkeen muuteta.

Vid parprogrammering finns det två roller: föraren och navigeraren. Rollerna delas ut i början av projektet och ändras inte under dess lopp.

(e) _____

(1p)

Name/Nimi/Namn: _____

Student number /Opiskelijanumero /Studentnummer: _____

- (f) The *severity* of a bug refers to its position in the fixing order of the found faults, and is therefore most appropriately determined by the tester who discovers the problem.

Virheen vakavuudella (severity) viitataan siihen järjestykseen, jossa virheitä korjataan virhelistasta. Näin ollen paras henkilö määrittelemään virheen vakavuuden on testaja, joka löytää virheen.

Ett fels (bug) allvarlighet (severity) refererar till dess ordning i korrigeringslistan av funna fel, och är därför läpligast bestämd av testaren som hittar felet.

(f) _____

(1p)

- (g) According to Tuomo Kähkönen, the concept of a *feature* can be used to simplify the requirements management process.

Tuomo Kähkösen mukaan käsitettä ominaisuus (feature) voidaan käyttää yksinkertaistamaan vaatimusten hallintaprosessia.

Enligt Tuomo Kähkönen kan användandet av konceptet egenskap användas till att underlätta kravhanteringsprocessen.

(g) _____

(1p)

- (h) Agile software development methodologies are not applicable to all kinds of projects.

Ketteriä ohjelmistokehitysmenetelmiä ei voi käyttää kaikenlaisissa projekteissa.

Viga (agila) mjukvaruutvecklingsmetodologier är inte tillämpbara på alla slags projekt.

(h) _____

(1p)

- (i) The use of frequent deliveries can be beneficial in distributed software development projects.

Tiheiden toimitusvälien käyttäminen voi olla hyödyllistä hajautetuissa ohjelmistokehitysprojekteissa.

Användning av frekventa leveranser kan vara nyttigt i distribuerade mjukvaruutvecklingsprojekt.

(i) _____

(1p)

- (j) The term "Iterative and Incremental Development" refers to an approach to building software in which the software is grown in small increments until it is ready.

Käsite "Iteratiivinen ja Inkrementaalinen Kehitys" viittaa ohjelmistokehityksen malliin, jossa ohjelmistoa kasvatetaan pienissä inkrementeissä, kunnes se on valmis.

Termen "Iterativ och Inkrementell Utveckling" hänvisar till en modell för att bygga mjukvara, i vilken mjukvaran växer i små inkrement tills den är färdig.

(j) _____

(1p)

- (k) U.S. software patents are valid in Europe.

Yhdysvalloissa myönnetyt ohjelmistopatentit ovat voimassa myös Euroopassa.

Amerikanska mjukvarupatent beviljade i USA är giltiga i Europa.

(k) _____

(1p)

- (l) Testing in agile software development is typically considered a separate phase to be carried out after the actual development has finished.

Ketterässä ohjelmistokehityksessä testauksen katsotaan tyypillisesti olevan erillinen vaihe, joka suoritetaan kun ohjelmistokehitys on tehty valmiiksi.

Testning i viga mjukvaruutveckling anses typiskt vara en separat fas som utförs efter att utvecklingen är klar.

(l) _____

(1p)

Name/Nimi/Namn: _____

Student number/Opiskelijanumero/Studentnummer: _____

- (m) Class diagrams are drawn during object-oriented design.
Luokkakaavioita piirretään oliopohjaisessa suunnittelussa.
Klassdiagram ritas under objektorienterad design.
- (m) _____
- (1p)
- (n) It is vital that the specification document is thoroughly verified, e.g., by using a document walkthrough.
On keskeistä, että määrittelydokumentin laatu varmistetaan tarkasti, esim. katselmoimalla.
Det är viktigt att specifikationsdokumentet verifieras noggrant, t.ex. genom en inspektion.
- (n) _____
- (1p)
- (o) The achievement of milestones is based upon deliverables, not dates.
Virstanpylväät saavutetaan tuotosten, ei päivämäärien mukaan.
"Milestones" uppnås på basis av resultat, inte tidtabell.
- (o) _____
- (1p)
- (p) The productivity of programmers is basically the same, which makes software project scheduling easier.
Ohjelmoijien tuottavuus on pohjimmiltaan sama, mikä helpottaa ohjelmistoprojektien aikataulutusta.
Programmerares produktivitet är i princip den samma, vilket gör det lätt att planera tidtabeller för programutvecklingsprojekt.
- (p) _____
- (1p)
- (q) TSP is a process improvement methodology targeting individual software engineers.
TSP on prosessinparannusmenetelmä, joka keskittyy yksittäisiin ohjelmistoinsinööreihin.
TSP är en processförbättringsметод som koncentrerar sig på den enskilda programingenjören.
- (q) _____
- (1p)
- (r) COCOMO is a model for software project effort estimation.
COCOMO on ohjelmistoprojektien työmäärien arviointiin tarkoitettu malli.
COCOMO är en modell för estimering av arbetsmängden i programutvecklingsprojekt.
- (r) _____
- (1p)
- (s) Dynamic modeling is done during object-oriented analysis.
Dynaamista mallinnusta tehdään oliopohjaisen analyysin aikana.
Dynamisk modellering görs under objektorienterad analys.
- (s) _____
- (1p)
- (t) When using Cleanroom software engineering, code artifacts are not compiled until they have passed inspection.
Cleanroom-menetelmässä koodia ei käännetä ennen kuin se on läpäissyt katselmuksen.
I Cleanroom-utveckling kompileras inte kod innan den godkänts vid en inspektion.
- (t) _____
- (1p)

Name/Nimi/Namn: _____

Student number /Opiskelijanumero /Studentnummer: _____

- (u) Adding people is the preferred way to catch up a late software project.
Henkilöstön lisääminen on paras tapa kuroa aikataulu kiinni myöhästyneessä ohjelmistoprojektissa.
Att öka på personalen är det bästa sättet att ta igen tidtabellen i ett försenat mjukvaruprojekt.
- (u) _____
- (1p)
- (v) A central theme in the spiral model is risk management.
Riskien hallinta on keskeinen teema spiraalimallissa.
Riskhantering är ett centralt tema i spiralmodellen.
- (v) _____
- (1p)
- (w) When using the Personal Software Process, you log your time spent on various software development activities with a one-minute granularity.
Käyttäessäsi PSP:tä kirjaat eri aktiviteetteihin kuluttamasi ajan minuutin tarkkuudella.
När du använder den personliga mjukvaruprocessen (PSP), loggar du den tid du använder till olika aktiviteter med en minuts noggrannhet.
- (w) _____
- (1p)
- (x) Maintenance incorporates aspects of all other phases of a software process.
Ylläpito sisältää piirteitä kaikista muista ohjelmistoprosessin vaiheista.
Underhåll innehåller aspekt av alla andra faser i mjukvaruprocessen.
- (x) _____
- (1p)
- (y) Software development life-cycle models define a sequence of activities (or 'steps') in which a piece of software is built.
Ohjelmistokehityksen elinkaarimallit määrittelevät sarjan aktiviteettejä (tai 'askeleita'), joissa ohjelmisto rakennetaan.
Livscykelmodeller för mjukvaruutveckling definierar en sekvens av aktiviteter (eller 'steg') enligt vilka mjukvaran byggs.
- (y) _____
- (1p)
- (z) Each phase of the RUP corresponds to an activity (e.g. requirements specification, design, implementation).
RUP:n jokainen vaihe vastaa aktiviteettia (esim. vaatimusten hallinta, määrittely, toteutus).
Varje fas i RUP motsvarar en aktivitet (t.ex. kravspecifikation, design, implementation).
- (z) _____
- (1p)

Name/Nimi/Namn: _____

Student number/Opiskelijanumero/Studentnummer: _____