

Studieordning for masteruddannelsen i informationsteknologi linjen i softwarekonstruktion

1. Introduktion	2
2. Formål	2
3. Indhold	2
4. Adgangskrav	3
5. Eksaminer	3
6. Rammer for sammensætning af studieplan	4
Samlet oversigt over masteruddannelsens fag/prøver:	4
7. Beskrivelse af de enkelte discipliner og prøver	4
7.1 Introducerende objektorienteret programmering.....	4
7.2 Arkitektur og Operativsystemer	5
7.3 Algoritmer og datastrukturer.....	6
7.4 Projektledelse	7
7.5 Programmering af Store Systemer	8
7.6 Distribuerede systemer og sikkerhed.....	8
7.7 Objektorienteret systemudvikling.....	9
7.8 Hovedopgave	11
8. Merit og dispensation	11

1. Introduktion

Ved Aarhus Universitet udbydes en masteruddannelse i Informationsteknologi, linjen i Softwarekonstruktion.

Uddannelsen udbydes i henhold til Undervisningsministeriets bekendtgørelse om masteruddannelsen i informationsteknologi under IT-Vest-samarbejdet (Bekendtgørelse nr. 793 af 21. august 2000).

Uddannelsen hører under Studienævnet for 2-fagsuddannelser, Det Naturvidenskabelige Fakultet, Aarhus Universitet.

Denne studieordning er godkendt på studienævnsmøde den 7. maj 2002.

2. Formål

Formålet med uddannelsen er at give studerende med praktisk erhvervserfaring videregående uddannelse i informationsteknologi. Uddannelsen skal give de studerende en teoretisk, analytisk, metodisk og praktisk indsigt i informationsteknologi.

Uddannelsen skal på videnskabeligt grundlag sætte de studerende i stand til at medvirke ved udvikling og implementering af ny informationsteknologi i private og offentlige virksomheder, herunder:

- 1) at analysere og behandle information med udgangspunkt i teorier og metoder inden for de informationsteknologiske discipliner,
- 2) at kunne indgå aktivt i forbindelse med planlægning, udvikling og ibrugtagning af informationssystemer samt tilrettelæggelse af arbejdsgange i forbindelse med anvendelse af disse systemer og
- 3) kritisk at kunne vurdere, hvilke muligheder og begrænsninger der er for anvendelsen af informationsteknologien.

3. Indhold

Linjen i Softwarekonstruktion kvalificerer til at løse krævende tekniske og designmæssige opgaver inden for IT-området. Uddannelsen sætter de studerende i stand til at

- 1) forstå og anvende de teorier og arbejdsmetoder, der knytter sig til anvendelse, udvikling og vurdering af software,
- 2) forstå udviklingen af og deltage i udvikling af software,
- 3) forstå og udnytte informationsteknologiens muligheder og begrænsninger inden for forskellige anvendelsesområder og

- 4) analysere, evaluere og kritisk vurdere software og anvendelsen af software.

Linjen i Softwarekonstruktion dækker følgende fagområder, der udgør uddannelsens kerneområder:

- 1) Programmering, herunder specielt objektorienteret programmering.
- 2) Algoritmer og datastrukturer.
- 3) Arkitekturer for datamater og datamatnetværk.
- 4) Distribuerede systemer med fokus på deres egenskaber og programmering.
- 5) Softwarearkitektur omfattende såvel klassiske arkitekturer som frameworks og design patterns.
- 6) Strukturering af information og data herunder databaser og database management.
- 7) Metoder og arbejdsformer i softwareudvikling.
- 8) Ledelse og organisering af softwareudviklingsprojekter.

Linjen i Softwarekonstruktion kan tillige dække bl.a. følgende fagområder:

- 1) Menneske-maskin-grænseflader, herunder brugbarhed af software.
- 2) Sikkerhed og kryptering.
- 3) Aftestning og verifikation.
- 4) Dokumentation.
- 5) Konfigurationsstyring og versionsstyring
- 6) Projektøkonomi.
- 7) Kvalitetsstyring og kvalitetsstyringssystemer.

4. Adgangskrav

Adgang til uddannelsen forudsætter at ansøgeren har bestået en relevant videregående uddannelse på mindst bachelorniveau og har mindst to års relevant erhvervserfaring. Endvidere skal ansøgere have matematikkundskaber svarende til gymnasialt A-niveau. Studienævnet kan på grundlag af en individuel vurdering dispensere fra adgangskravet, hvis ansøgeren skønnes at have tilsvarende forudsætninger for at kunne gennemføre uddannelsen. Kravet om erhvervserfaring kan dog ikke fraviges.

5. Eksaminer

Uddannelsen består af et antal prøver. Hver prøve skal bestås for sig. Beståede prøver kan ikke tages om.

Prøverne bedømmes enten med intern eller ekstern censur. Interne prøver bedømmes af eksaminator samt ingen eller flere interne censorer. Eksterne prøver bedømmes af eksaminator og en eller flere ministerielt beskikkede censorer.

Ved prøverne anvendes enten karakterskala efter gældende regler (13-skalaen) eller bedømmelsen består/ikke-består.

Bedømmelsen af en prøve skal ved sommereksamen foreligge senest 1.juli, ved vintereksamen senest 1.februar.

En studerende kan højst tre gange indstille sig til den samme prøve. Studienævnet for 2-fagsuddannelser kan ved dispensation tillade indstilling for fjerde gang, hvis særlige forhold gør sig gældende.

Aarhus Universitet udsteder bevis for gennemført uddannelse. Til studerende, der forlader uddannelsen uden at have gennemført den, udsteder Aarhus Universitet bevis for beståede dele af uddannelsen.

6. Rammer for sammensætning af studieplan

Samlet oversigt over masteruddannelsens fag/prøver:

Fagets/prøvens navn	censur	vægtning	bedømmelse	Eksamensform
Introducerende objektorienteret programmering	Intern	10 ECTS	bestået/ikke bestået	Mundtlig - forberedelse
Arkitektur og operativsystemer	Ekstern	10 ECTS	13-skala	Mundtlig + forberedelse
Algoritmer og datastrukturer	Ekstern	10 ECTS	13-skala	Skriftlig N/A
Projektledelse	Ekstern	10 ECTS	13-skala	Mundtlig + forberedelse
Programmering af store systemer	Ekstern	10 ECTS	13-skala	Mundtlig + forberedelse
Distribuerede systemer og sikkerhed	Ekstern	10 ECTS	13-skala	Mundtlig + forberedelse
Objektorienteret systemudvikling	Ekstern	10 ECTS	13-skala	Mdt./gruppe - forberedelse
Hovedopgave	Ekstern	20 ECTS	13-skala	Mundtlig - forberedelse

7. Beskrivelse af de enkelte discipliner og prøver

7.1 Introducerende objektorienteret programmering

Målbeskrivelse

Målet med kurset er at den studerende lærer principper for og teknikker til systematisk konstruktion af programmer, specielt objektorienterede programmer. Endvidere er det målet at den studerende får erfaring i løsning, beskrivelse og programmering af mindre opgaver samt opnår kendskab til et sædvanligt programmeringssprog og de dertil hørende programmeringssprogsbegreber.

Indhold

Kurset dækker centrale områder inden for grundlæggende objektorienteret programmering, herunder: basale sprogkonstruktioner, objekter og klasser, objektorienteret modellering, algoritmer, modularisering, arv og polymorfi, frameworks, grafiske brugergrænseflader, software engineering i praksis.

Undervisningsmateriale

Endnu ikke fastlagt. Tidligere har følgende materiale været anvendt:

Jaime Niño & Frederick A. Hosch, *An Introduction to Programming and Object-Oriented Design Using Java*, John Wiley & Sons, 2002.

David Gries & Paul Gries, *ProgramLive – Master Java Programming in a Dynamic, Self-Paced Environment*, John Wiley & Sons, 2002.

Undervisningsform

Forelæsninger: 3 timer pr. uge.

Øvelser: 3 timer pr. uge.

Obligatoriske opgaver: Der stilles i løbet af kurset en række ugeopgaver. Aflevering af disse opgaver er obligatorisk og tilfredsstillende besvarelse af opgaverne er en forudsætning for at kunne indstille sig til eksamen.

Eksamensform

En mundtlig prøve af 20 minutters varighed uden forberedelse som bedømmes bestået/ikke-bestået med intern censur.

Vægtning

10 ECTS.

7.2 Arkitektur og Operativsystemer

Måbeskrivelse

Målet med kurset er at give den studerende kendskab til opbygning og virkemåde af det datamatiske system bestående dels af materielle enheder så som ydre enheder og procesenheder, dels af det maskinnære programmel.

Indhold

Datamaskinen opfattet som niveauer af abstrakte maskiner omfattende det digitale niveau, mikroprogramniveauet, det traditionelle maskinniveau og operativsystemniveauet. Eksempler på implementation af et højere liggende niveau ved hjælp af et lavere liggende niveau. Det datamatiske system, dets komponenter (procesenheder, ydre enheder, lagre, osv.) og deres samspil på det traditionelle maskinniveau, herunder synkronisering af parallelt kørende enheder.

Undervisningsmateriale

Endnu ikke fastlagt.

Undervisningsform

Forelæsninger: 3 timer pr. uge

Øvelser: 3 timer pr. uge

Obligatoriske opgaver: Der stilles i løbet af kurset en række ugeopgaver. Aflevering af disse opgaver er obligatorisk og tilfredsstillende besvarelse af opgaverne er en forudsætning for at kunne indstille sig til eksamen.

Eksamensform

En mundtlig prøve med 1/2 times forberedelse som bedømmes efter 13-skalaen med ekstern censur.

Vægtning

10 ECTS

7.3 Algoritmer og datastrukturer

Baggrund

Når et program skal løse en opgave, kræves både en løsningsmetode (en algoritme) og en måde at opbevare data (en datastruktur), og effektiviteten af begge dele har direkte indflydelse på effektiviteten af det pågældende program. Algoritmer og datastrukturer er derfor centrale begreber for al softwareudvikling.

Inden for datalogi er der udviklet et væld af algoritmer og datastrukturer til forskellige problemer. Nogle af disse problemer er specialiserede, mens andre er fundamentale problemer som dukker op i mange sammenhænge. Dette forløb er en gennemgang af algoritmer og datastrukturer for et stort udvalg af disse fundamentale problemer.

Udover en præsentation af de enkelte algoritmer og datastrukturer, vil disse også blive analyseret med hensyn til korrekthed og effektivitet.

Vægten i forløbet vil ligge på principper for konstruktion og analyse, mere end på den egentlige programmering (som deltagerne forudsættes allerede at have kendskab til).

Måbeskrivelse

Formålet er at den studerende bliver rustet til selv at frembringe løsninger på algoritmiske problemer, samt til at analysere egne og andres løsninger og derved foretage mere kvalificerede valg.

Indhold

- Basale algoritmer og datastrukturer: søgning, fletning, sortering, lister, køer, stakke, træer
- Elementær teori: Algoritmers korrekthed og effektivitet
- Fundamentale paradigmer: del-og-kombiner, dynamisk programmering, kombinatorisk søgning
- Dataabstraktioner: Prioritetskøer, ordbøger, ækvivalensrelationer
- Grafalgoritmer: gennemløb, sammenhængsegenskaber, udspændende træer, korteste veje.

Undervisningsmaterialer

Eksempel på mulig lærebog:

Goodrich & Tamassia, *Data Structures and Algorithms in Java*, John Wiley & Sons, 1998.

Anslåede bogudgifter: Ca. 500,- kr.

Opgaver og andet materiale distribueres via World Wide Web.

Undervisningsform

Forelæsning: 3 timer pr. uge

Øvelser: 3 timer pr. uge

Obligatorisk opgave: Tilfredsstillende gennemførelse af den obligatoriske opgave er en forudsætning for at kunne gå til eksamen.

Eksamensform

Fire timers skriftlig eksamen (indstilling til eksamen forudsætter at den obligatoriske opgave er godkendt) som bedømmes efter 13-skalaen med ekstern censur.

Vægtning
10 ECTS

7.4 Projektledelse

Målbeskrivelse

Efter at have gennemført kurset skal den studerende

- have indsigt i projektet som arbejdsform
- have kendskab til en række metoder og værktøjer, som knyttes til planlægning og beskrivelse af et grundlag for en projektgennemførelse
- have kendskab til metoder og værktøjer, der knytter sig til fremdriftsstyring samt kravs- og ændringsstyring
- have kendskab til et konkret projektstyringsværktøj
- have kendskab til forskellige forhold, der kan hæmme hhv. stimulere arbejdet i en projektgruppe.

Indhold

- Projektledelse som fagdisciplin
- Projekttyper og -modeller
- Projektetablering
- Kravspecifikation
- IT-kontrakten som styringsredskab
- Function point-analyse
- Teambuilding
- Projektstyring
- Projektstyringsværktøjer
- Kravstyring
- Projektledelse af udviklingsprojekter
- Projektledelse af anskaffelsesprojekter
- "Brandlukning"
- Projektafslutning.

Undervisningsmateriale

Eksempel på mulige lærebøger:

Steve McConnell, *Rapid Development*, Microsoft Press, 1996.

Andreas Munk-Madsen, *Strategisk Projektledelse – Styr på udviklingen*, Marko 1996.

Undervisningsform

Forelæsninger, øvelser samt præsentation af konkrete projektstyringskoncepter.

Eksamensform:

En mundtlig prøve med 1/2 times forberedelse som bedømmes efter 13-skalaen med ekstern censur.

Vægtning:
10 ECTS

7.5 Programmering af Store Systemer

Målbeskrivelse

Kurset omhandler softwareudvikling (software engineering) med henblik på konstruktion af store programmer. Der lægges vægt på en begrebsmæssig forståelse af programmeringssprogs rolle i modellering af aspekter af anvendelsesområdet, konstruktioner til strukturering af store programmer, samt værktøj til programudvikling.

Indhold

Begrebsmæssigt fundament for softwareudvikling: Begreber (intension, extension, designation), aristoteliske og prototypiske begreber, klassifikation og komposition, informationsprocesser.

Objekt-orienterede sproglige konstruktioner til understøttelse af konstruktion af store programmer, herunder virtuelle klasser og virtuelle procedurer samt klasse og procedure variable.

Arkitektur af store software systemer, eksempelvis design mønstre, frameworks og komponenter.

Introduktion til og konstruktion af grafiske brugergrænseflader.

Værktøjer til håndtering og vedligehold af store systemer, herunder modularisering og fragmentering af kildetekster, samt værktøjer hertil (herunder CASE værktøjer, editorer, debugger GUI editorer og versions- og revisionssystemer).

Teknikker og værktøjer til dokumentation og test af software.

Teknikker og værktøjer til håndtering af store datamængder, herunder database systemer i form af relationelle og objekt-orienterede databasesystemer (SQL og ODMG).

Undervisningsmateriale

Kompendium samt udleverede noter

Undervisningsform

Forelæsning: 3 timer pr. uge

Øvelser: 3 timer pr. uge

Obligatorisk opgave: Der stilles et obligatorisk projekt. Tilfredsstillende besvarelse af projektet er en forudsætning for at kunne indstille sig til eksamen.

Eksamensform

En mundtlig prøve, som kan inddrage de obligatoriske opgaver.

Vægtning

10 ECTS

7.6 Distribuerede systemer og sikkerhed

Målbeskrivelse

Hovedformålet med kurset er at introducere fundamentale begreber og teknikker for distribuerede systemer, dvs. systemer hvor computere er forbundet i et netværk for at samarbejde og dele resurser så som filer, printere og Internet-adgang. Kurset vil diskutere de basale begreber vedrørende computer-netværk, inklusiv netværksprotokoller og internet.

Indhold

- Arkitekturer for distribuerede systems
- Interprocess-kommunikation
- Remote kommunikation
- Sikkerhed
- Transaktioner
- Replikation
- Wide-area-netværk
- Local-area-netværk
- Internet og internetprotokoller

Obligatoriske opgaver

Der stilles en række obligatoriske opgaver i løbet af kurset.

Tilfredsstillende besvarelse af disse opgaver er en forudsætning for at få lov til at gå til eksamen.

Undervisningsmateriale

Distributed Systems, Concepts and Design, 3rd Edition,
by Couloris, George; Jean Dollimore and Tim Kindberg.
Addison-Wesley, 2000.

Data and Computer Communication, 6th Edition,
by Stallings, William.
Prentice-Hall, 2000.

Diverse noter.

Undervisningsform

Forelæsning: 3 timer pr. uge

Øvelser: 3 timer pr. uge

Eksamensform

Mundtlig eksamination i ca. 30 minutter med ca. 30 minutters forberedelse.

Vægtning

10 ECTS

7.7 Objektorienteret systemudvikling

Målbeskrivelse

Igennem de seneste år har måden at udvikle IT-systemer på gennemgået en markant forandring. Det har i mange år været almindeligt accepteret at man laver systemer efter vandfaldsmodellen ved først at formulere alle krav, dernæst analysere problemområdet, derefter designe en løsning og til sidst realisere og implementere denne løsning. Denne udviklingsmodel har vist sig at være utilstrækkelig. Derfor skal udviklingsmodellen og det system der produceres kunne håndtere ændringer af krav på en effektiv måde.

En objektorienteret fremgangsmåde medvirker i høj grad til at lave forandringsberedte systemer, og hvis denne fremgangsmåde suppleres med en inkrementel og iterativ proces med tidlig og hyppig feedback, opnås en udviklingsmodel der kan bidrage til at opfylde de krav der i dag er til udvikling af IT-systemer.

Kurset giver deltagerne en både praktisk og teoretisk indsigt i et systemudviklingsforløb, herunder de enkelte aktiviteter og deres sammenspil i et iterativt og inkrementelt forløb. En stor del af kurset er et projektarbejde hvor en iterativ, objektorienteret udviklingsmodel prøves i praksis.

I en iterativ og inkrementel udviklingsproces er ændringer normen; derfor spiller kvalitetskriterier der fokuserer på at gøre systemerne forandringsberedte en central rolle. Metoder til at opnå kriterierne på, for eksempel brug af mønstre, er en del af kurset.

For at kunne håndtere ændringer på en god måde fokuseres på at få afdækket de største risici tidligt i forløbet; derfor anvendes en eksperimentel tilgang hvor der fokuseres på brugerinddragelse i udviklingsprocessen.

Indhold

Faget består af tre aktiviteter: forelæsninger, øvelsestimer i mindre hold samt et obligatorisk projekt. Der vil være en tæt sammenhæng mellem øvelser og projekt.

Forelæsningerne vil dække områderne:

- Den objektorienterede systemudviklingsproces
- Projektrelaterede emner
- Perspektivering af udviklingsprocessen

Kurset er delt i tre faser:

1. En intensiv teoretisk gennemgang af udviklingsforløbet med mindre hjemmeopgaver, der diskuteres ved øvelsestimerne
 2. Et projektforsløb hvor øvelserne bruges til erfaringsudveksling mellem grupperne og konsultering med vejleder
 3. En perspektiveringsfase hvor både forelæsninger og øvelser trækker på projekterfaringerne.
- Og Obligatoriske opgaver

Undervisningsmateriale

Eksempel på mulig lærebog: Applying UML and Patterns 2nd ed, Craig Larman

Eksamensform

Med udgangspunkt i projektet afholdes en mundtlig gruppeprøve uden forberedelse som bedømmes individuelt efter 13-skalaen med ekstern censur.

Vægtning

10 ECTS

7.8 Hovedopgave

Formål

Ved udarbejdelsen af hovedopgaven skal den studerende demonstrere fortrolighed med almindelige principper for videnskabelig metode og færdighed i at anvende metoder og teorier til selvstændigt at afgrænse og behandle problemstillinger inden for området softwarekonstruktion.

Vejledning

Uddannelsens hovedopgavestudium sker under vejledning. Ved starten på hovedopgavestudiet aftales projektets titel samt tidspunkt for aflevering af hovedopgave. Med mindre andet er fastlagt, er det den studerendes ansvar at finde en vejleder, men IT-Vest ved AU er naturligvis behjælpelig hermed.

På linien i softwarekonstruktion er tilrettelagt et forløb hvor hovedopgaven udarbejdes i tilknytning til faget Objektorienteret systemudvikling med læreren fra dette fag som vejleder; hovedopgaven erstatter det projekt der normalt indgår i faget. I tilfælde hvor det på grund af emnevalg eller af andre årsager er u hensigtsmæssigt at udarbejde hovedopgaven i tilknytning til dette fag, er der muligt at definere en uafhængig hovedopgave; i sådanne tilfælde pålægger det den studerende selv at finde en vejleder og i samråd med denne at fastlægge projektets titel samt tidspunkt for aflevering.

Eksamensform

Prøven er et frit skriftligt arbejde og en mundtlig prøve (et forsvar).

En hovedopgave kan være individuel eller i gruppe med op til tre studerende. Ved gruppebesvarelse skal mindst halvdelen af den enkelte studerendes bidrag kunne gøres til genstand for individuel bedømmelse.

Den mundtlige prøve (er individuel og) finder sted ved at de(n) studerende efter aflevering af det skriftlige arbejde og senest en uge inden selve forsaret modtager en opgave formuleret med udgangspunktet i det skriftlige arbejde. Forsvaret udformer sig som en besvarelse af den udleverede opgave fulgt af en samtale.

Censur og bedømmelse: ekstern censur efter 13-skalaen. Der gives én samlet karakter efter 13-skalaen for det skriftlige arbejde og den mundtlige prøve. Der medvirker ekstern censor ved bedømmelsen; vejleder fungerer som eksaminator.

8. Merit og dispensation

Studienævnet for 2-fagsuddannelser kan godkende, at prøver bestået under andre uddannelser kan erstatte prøver i uddannelserne beskrevet i studieordningen. Studienævnet kan endvidere tillade individuelle ændringer i studieprogrammet, når dette ikke strider med uddannelsens målsætning eller bestemmelser fastsat i bekendtgørelserne for uddannelserne.