

Introduktion till Examensarbeten

på kandidatnivå (kurs: 2DV50E)

Vad är ett examensarbete?

- Ett projekt där du arbetar med ett väl definierat problem.
- Du kan arbeta ensam eller i grupp om två studenter.
- Projektet kan göras på universitetet eller på ett företag.
- Du kommer att ha en handledare från universitetet,
- ... och en handledare från företaget om du gör ditt exjobb på ett företag.

Vad är ett examensarbete?

- Omfattningen av arbetet är 15 hp.
- Det går på halvfart över två terminer.
- Totalt cirka 20 veckor.
- Förkunskapskrav:
 - Minst 75 hp i datavetenskap
 - Minst 15 av dessa måste vara i 2DV... kurser

Förkunskapskrav

- I vissa fall kan du bli “villkorligt antagen” till kursen.
- Detta innebär att du inte uppfyller alla förkunskapskrav, och du måste diskutera med mig om du får läsa kursen eller ej.
- Det kan till exempel vara att du klarat av poäng som inte har dykt upp i Ladok ännu.
- Varje fall hanteras individuellt.

Var ska du börja?

- Du måste hitta ett problem att arbeta med i ditt projekt.
- Du kan antingen:
 - Välja ett tillgängligt förslag
 - Fråga runt efter förslag
 - Själv hitta ett ämne
- Det är generellt lättare att välja ett färdigt förslag än att definiera ett eget ämne.

Färdiga förslag

- En lista över tillgängliga förslag finns på:
 - <https://coursepress.lnu.se/subject/thesis-projects/proposals/>
- Listan innehåller förslag från företag och från forskare på universitetet.

Topic	Company	Contact Person	More Info
A Big Data Java Corpus		Jonas Lundberg	Link
N-Grams as a Measure of Naturalness and Complexity		Jonas Lundberg	Link
Testing	Combitech	Toresson Albin	Link
Information Systems	Combitech	Toresson Albin	Link
Domain Driven Security	Omegapoint	Narges Khakpour	
Projects at IST (several topics)	IST	Tor Olsson	Contact Tor for more info
Projects at Sigma (several topics)	Sigma	Fredrik Alserin	Contact Fredrik for more info
Topics in Information Visualization (list of several topics)		Andreas Kaman	Link



Välja ett färdigt förslag

- Om du är intresserad av ett förslag:
 - Läs hela beskrivning (följ länken).
 - Kontrollera att du uppfyller kraven (om det finns några).
 - Skicka ett email till kontaktpersonen och säg att du är intresserad av förslaget.
 - Om kontaktpersonen accepterar att du startar projektet, meddela kursansvarig via email.
 - Kontaktpersonen kan säga nej om förslaget redan har blivit tilldelat en annan student.

Fråga runt efter förslag

- Hitta ett område du är intresserad av.
- Fråga en lärare som är erfaren inom området om hen har något förslag till dig.
- Är det något företag som sysslar med saker du är intresserad av?
- Fråga om dom har några förslag till dig.

Hitta ditt eget ämne

- Börja med att hitta ett område du är intresserad av:
 - Programvaruarkitekturer
 - IT säkerhet
 - Internet of Things
 - ...
- Läs på om området i böcker, bloggar, artiklar, etc.
- Finns det ett intressant problem du kan arbeta med?
- I så fall, diskutera problemet med en lärare eller kursansvarig för att se om det är lämpligt för ett exjobb.

Vad är ett lämpligt problem?

- Ingen given lösning (det har inte redan lösts i samma kontext)
 - Exempel: "Hur kan man sortera en lista med ord" är redan löst
- Inte för stort eller för komplext
 - Exempel: "Göra bättre användarinterface för mobila enheter" är för stort och otydligt

Vad är ett lämpligt problem?

- Inte för litet eller för trivialt
 - Exempel: jämföra användarinterface i Word och OpenOffice
- Inte bara implementera någonting som redan finns
 - Inget problem, bara arbete
- Det måste finnas något att undersöka!
- Mer om detta senare.

Exjobb på företag

- Du kommer att ha en handledare på universitetet och en på företaget.
- Företag har ofta andra mål med arbetet än universitetet, och du måste se till att båda parternas krav uppfylls.
- Det är upp till företaget om de kan erbjuda dig en arbetsplats på kontoret.
- Universitetet kan inte stå för resor till/från företaget.
- Ofta behöver du skriva på ett Non-Disclosure Agreement (NDA). Det är oftast inga konstigheter, men läs igenom vad du skriver på.
- Eftersom du inte är anställd av företaget har Du äganderätt till saker som du producerar. Du kan dock överlåta äganderätten till företaget i ett skriftligt avtal.

Informationsresurser





Degree Projects in Computer Science

Home

[General Information](#)[Project Proposals](#)

Introduction to Degree Projects

[Requirements](#)[Projects at external companies](#)

Useful Resources

[Tools and links](#)

Submissions

[Time plan](#)[Project Description](#)[Project Plan](#)[Report](#)

Problem Formulation

[Problem](#)[Objectives](#)[Literature Review](#)

Methods

[Method overview](#)

General Information

Senast ändrad 2017-04-20 02:14 av [Johan Hagelbäck](#) | [Edit](#)

What is a degree project?

A degree project is the last thing you do in your studies. In the project you work alone, or in a group of two students, on a well-defined problem. The purpose is to make a more detailed and in-depth investigation of a topic that interests you. The project can be carried out at the university or at an external company, and spans over a whole study period on half-time (bachelor level). You work on your project for 20 weeks so it is important that you choose a topic that you find interesting!

Where do I start?

The first step is to find a topic for your project. You can either:

- Select a topic from the [list of open project proposals](#).
- Ask a teacher or research that is experienced in an area that you are interested in if they have a topic for you.
- Propose your own topic.

It is generally easier to select an open project proposal or ask around for a topic, than defining your own.

What is a suitable topic?

For a topic to be suitable it must be well-defined, of reasonable size for a degree project, something that can be investigated, and within computer science. Simply re-implementing something that already exists is not allowed. There must be something new in the project! In general, all ideas that fulfill these basic criteria are allowed, but some ideas can be denied if we cannot find a supervisor that has knowledge about the area or if the topic is too small or too large.

<https://coursepress.lnu.se/subject/thesis-projects/>

Projektbeskrivning och Projektplan

Första inlämningen

- Du skall skriva en en-sidas projektbeskrivning innehållande:
 - Bakgrund. Beskriv området ditt projekt är inom.
 - Problembeskrivning. Beskriv vilket specifikt problem du ska försöka lösa?
 - Förväntat resultat. Vilka/vilka resultat förväntar du dig från projektet?
- Mallar och exempel på projektbeskrivning finns på:
 - <https://coursepress.lnu.se/subject/thesis-projects/project-description/>
- Deadline och inlämning: **Se Mymoodle**

Student:	Sarah Kerrigan	sake98@student.lnu.se
----------	----------------	-----------------------

Supervisor:	-
External company:	-
External supervisor:	-

Background

Real-Time Strategy (RTS) games are a sub-genre of strategy games where the game runs in real-time, in contrast to turn-based strategy games such as Chess or Civilization. Since the game is running in real-time, the player has to make often complex decisions in very short time and also be able to quickly react if something unexpected happens in the game. Players can play against other human players or against computer-controlled opponents. The complex and quick decision making is a very difficult task for a computer-controlled opponent, and developing such a software system is not a trivial task.

Problem formulation

There exist several open-source bots for the popular RTS game Starcraft. A majority of these, according to their descriptions, does not explicitly react to what the opponent does and only uses a static strategy for building bases and creating armies. I want to investigate how an adaptive, reactive strategy can be incorporated in an existing open-source bot. I plan to use some form of fuzzy logic for the adaptive strategy selection since it is capable of selecting among several options that fit the problem to a certain degree.

Expected result

I expect the adaptive, reactive strategy selection to be better (i.e. win more games) than a static strategy selection, but that the problem is very complex and not everything can be solved within the scope of a thesis project.



Diskussionsseminarier

- Kursen startar med tre diskussionsseminarier där era förslag diskuteras och förbättras.
- Ni kommer att delas in i grupper
- 1 timme per grupp och vecka
- Grupper och schema kommer att läggas ut på Mymoodle efter inlämningen av projektbeskrivningen
- Distansstudenter diskuterar sina projektidéer med kursansvarig via email eller Skype (boka möte via email)

Diskussionsseminarier

- På seminarier ska du i grupp och med en lärare diskutera:
 - Problemformulering
 - Mål
 - Metod
 - Omfattning och begränsningar
 - Förväntat resultat
- På varje seminarie får du feedback på saker du ska arbeta på till nästa seminarie, där din idé förbättras ytterligare.
- Se till att utnyttja dessa seminarier!

Projektplan

- Efter seminarierna ska du skriva och lämna in en projektplan.
- Projektplanen är en utökad och mer formell beskrivning av ditt projekt, och hur du tänker angripa problemet.
- Den ska också innehålla en tidsplan som hjälper dig att strukturera tiden under projektets gång.
- Projektplanen beskriver vad du ska göra, och hur du ska göra det.
- Du kan börja arbeta med projektplanen under seminarieveckorna.

Projektplan

- Projektplanen måste godkännas av kursansvarig innan du formellt kan börja arbeta på projektet.
- Projektplanen hjälper dig att planera tiden och att se till att du är på rätt spår.
- Se till att tidsplanen har specifika och rimliga delmål!
- Mallar och exempel på projektplan finns på:
 - <https://coursepress.lnu.se/subject/thesis-projects/project-plan/>
- Deadline och inlämning: **se Mymoodle**

General Information

Title:	Reactive strategy selection in a Real-Time Strategy game
External company:	-

Persons involved

Student 1:	Sarah Kerrigan	sake98@student.lnu.se
Student 2:		

Supervisor:	Jim Raynor
External supervisor:	-

Background

Real-Time Strategy (RTS) games are a sub-genre of strategy games where the game runs in real-time, in contrast to turn-based strategy games such as the board game Chess or the video game Civilization[1]. Example of popular RTS games are Command & Conquer and Starcraft 1 and 2. Since the game is running in real-time, the player has to make often complex decisions in very short time and also be able to quickly react if something unexpected happens in the game. The vast amount of possible actions and strategies and the fast pace of the game is probably one reason why RTS games are very popular in e-Sports. Players can play against other human players or against computer-controlled opponents, usually called bots.

Problem formulation

The goal of this thesis project is to expand an existing bot for the RTS game Starcraft[2] with an adaptive, reactive strategy selection. The bot currently only has a static strategy that never changes regardless of what its opponent does. The reactive strategy selection will analyze what the opponent does using fuzzy logic and, if needed, modify the current selected strategy. To limit the scope of the thesis project only one of the three fractions in Starcraft, known as Protoss, will be used. We will also only consider how effective the implemented system is in terms of winning games, and not take player enjoyment into consideration. We expect the adaptive, reactive strategy selection system to outperform the same bot using a static, non-adaptive strategy.

Motivation

Developing computer-controlled opponents in RTS games is a very challenging task. It can broadly be divided into two main parts: micro- and macro-management. Micro-management involves controlling single units in the game, for example moving, shooting and using special abilities. Macro-management involves taking decisions on a higher level, for example which units to use in the attacking army and when and where to attack the opponent. Of these two macro-management is the most complex for a bot. Ontanon et al. states in their extensive survey of the field that reacting to opponents is today one of the most important challenges for RTS bot developers[3].

Fuzzy logic is a method that is well suited for reactive systems[4]. Instead of giving a boolean answer, yes or no, fuzzy logic uses real numbers between 0 and 1 and can therefore be true or false to some degree. This is useful in strategy selection since it is very common that more than one strategy suits a particular situation.

Objectives

O1	Implement an adaptive, reactive strategy selection into an existing Starcraft bot
O2	Select a number of suitable maps to use in the experiment
O3	Run an experiment where the modified bot is compared to the original, static bot on the selected maps

Method

The problem will be answered by conducting a series of experiments that provide quantitative data. The modified bot will play a number of games against the original bot and the number of wins, losses and draws will be measured. A pool of different maps will be used to ensure the result is reliable.

Time plan

Week	Milestone
3	Literature review finished
7	First version of fuzzy controller
8	Draft of introduction and background chapters
9	Testing session 1 finished
12	Second version of fuzzy controller
13	Draft of methodology and implementation chapters
13	Testing session 2 finished
15	Implementation finished
16	Experiments finished
17	Draft of results, analyzis, discussion and conclusion chapters
19	Report finished

Rekommendationer för skrivande

Rekommendationer för skrivande

- Börja skriva tidigt, skriv regelbundet
- Vänta inte tills projektet är klart eller tills du “känner dig redo att skriva”.
- Risken är stor att du inte kommer att skriva alls!
- Att skriva om ditt projekt hjälper dig att utveckla dina idéer och identifiera vad som saknas.
- Genom att skriva regelbundet bygger du upp ditt självförtroende i att skriva.
- Genom revidering plockar du ut de delar som passar i slutrapporten, och kastar det som inte passar in.

Rekommendationer för skrivande

- Ibland är det väldigt svårt att komma igång med att skriva.
- Du vet att du måste skriva, men du sitter och stirrar på den tomma sidan framför dig.
- Det finns några strategier du kan testa och se om de fungerar för dig:
 - Skriva till uppmaningar
 - Fritt skrivande
 - Generativt skrivande
- Mer information finns här:
 - <https://coursepress.lnu.se/subject/thesis-projects/how-to-write/>

Grammatik och stavning

- Använd verktyg för grammatik och stavningskontroll.
- Jag rekommenderar starkt att använda Grammarly (om ni skriver på engelska):
<http://grammarly.com>
- Gratisversionen är väldigt bra, men överväg att betala för Premium under tiden ni skriver er rapport.

Roller och Ansvar



Ansvar: Student

- Studenten ansvarar för att:
 - Skriva och lämna in projektbeskrivning och projektplan i tid.
 - Tillsammans med handledare bestämma hur ofta och var handledningsmöten ska hållas. Möten kan till exempel hållas varje vecka, varannan vecka, eller endast när studenten kontakter handledaren och begär ett möte.
 - Tillsammans med handledaren planera och diskutera leveranser (när delar av ditt projekt ska vara klara och lämnas in till handledaren för feedback).
 - Lämna in leveranser och utkast på rapporten till handledaren innan möten.
 - Kontinuerligt (åtminstone varje vecka) informera handledaren om hur arbetet fortlöper via email.
 - Se till att arbetet fortlöper enligt tidsplanen i projektplanen.
 - Driva projektet framåt och initiera diskussioner och möten.
 - Hantera och besvara kritik, vägledning och förslag som din handledare ger dig..
 - Informera din handledare om svårigheter, tekniska så väl som icke tekniska och personliga, som påverkar arbetet inträffar.
 - Korrekturläsa utkast innan de skickas till handledaren.

Ansvar:Handledare

- Handledaren ansvarar för att:
 - Hjälpa dig att planera ditt arbete så att du hinner bli klar i tid.
 - Se till att ditt projekt och din rapport är av tillräcklig standard innan du presenterar ditt arbete och om inte, informera dig om vad som behöver göras mer för att nå tillräcklig standard.
 - Träffa dig regelbundet och diskutera arbetet samt ge feedback på det du gjort.
 - Besvara email inom rimlig tidsram.
 - Läs och kommentera utkast på rapporten. Notera att handledare huvudsakligen ska kommentera på innehållet. Det är studentens ansvar att språkbruket håller tillräckligt hög kvalitet, så att den begränsade handledningstiden utnyttjas till det som är viktigt.
 - Meddela kursansvarig om du hinner bli klar i tid för att presentera ditt arbete på ett givet datum.

Ansvar: Kursansvarig

- Kursansvarig ansvarar för att:
 - Informera studenterna om hur examensarbeten ska genomföras på universitetet.
 - Informera studenterna om var de kan hitta viktig information (betygskrav, schema, etc.).
 - Informera studenter och handledare om deadlines (när projektplan ska lämnas in, när slutrapporten ska vara klar, etc.).

Ansvar: Examinator

- Examinatorn bedömer ditt arbete för att säkerställa att det uppfyller kraven för ett examensarbete, och är ansvarig för att sätta betyg på ditt arbete.
- Bedömning görs vid tre tillfällen:
 - När du lämnar in projektbeskrivning. Bedömning görs för att säkerställa att ditt projekt är rimligt och kan genomföras inom tidsramen för exjobb, och att det inte är för komplext.
 - När du lämnar in projektplanen. Bedömning görs för att säkerställa att forskningsfrågor och metodval är lämpliga.
 - Efter presentationen när du lämnar in rapporten. Bedömning görs för att säkerställa att arbetet är av tillräcklig kvalitet och att innehåll och språkbruk håller tillräcklig standard.

Betygskrav



Bedömningsmatris för Examensarbeten på Kandidatnivå

	Process	Innehåll	Presentation
A	- Planera och genomföra arbetet inom överenskomna tidsramar, visa god initiativförmåga samt vara öppen för handledning och kritik. - Kunna identifiera egna behov av ny kunskap, samt visa god förmåga att inhämta dessa kunskaper. - Visa god förmåga att sätta sig in i ett annat arbete och formulera viss kritik.	- Utifrån problemställning, visa god förmåga att tillämpa färdigheter som problemformulering, metodval, utveckling, analys och utvärdering. - Där så är relevant för uppgiften, visa god medvetenhet om samhällsliga aspekter, inklusive ekonomiskt, socialt och ekologiskt hållbar utveckling, och etiska aspekter på forskningsarbete.	- Uppvisa väl disponerad rapport, med tydlig redovisning av arbete och resultat, viss analys och argumentation, samt god språkbehandling och formalia. - Visa god förmåga att muntligt redovisa, diskutera och argumentera för arbetet.
C	- Planera och genomföra arbetet inom överenskomna tidsramar, visa initiativförmåga samt vara öppen för handledning och kritik. - Kunna identifiera egna behov av ny kunskap, samt visa förmåga att inhämta dessa kunskaper. - Visa förmåga att sätta sig in i ett annat arbete och formulera viss kritik.	- Utifrån problemställning, visa förmåga att tillämpa färdigheter som problemformulering, metodval, utveckling, analys och utvärdering. - Där så är relevant för uppgiften, visa medvetenhet om samhällsliga aspekter, inklusive ekonomiskt, socialt och ekologiskt hållbar utveckling, och etiska aspekter på forskningsarbete.	- Uppvisa skriftlig rapport, med tydlig redovisning av arbete och resultat, viss analys och argumentation, samt god språkbehandling och formalia. - Visa förmåga att muntligt redovisa, diskutera och argumentera för arbetet.
E Godkänt	- Planera och genomföra arbetet inom överenskomna tidsramar, visa viss initiativförmåga samt vara öppen för handledning och kritik. - Visa förmåga att inhämta nya kunskaper. - Visa viss förmåga att sätta sig in i ett annat arbete och formulera viss kritik.	- Utifrån problemställning, visa viss förmåga att tillämpa färdigheter som problemformulering, metodval, utveckling, analys och utvärdering. - Där så är relevant för uppgiften, visa viss medvetenhet om samhällsliga aspekter, inklusive ekonomiskt, socialt och ekologiskt hållbar utveckling, och etiska aspekter på forskningsarbete.	- Uppvisa skriftlig rapport, med redovisning av arbete och resultat, viss analys och argumentation, samt godtagbar språkbehandling och formalia. - Visa förmåga att muntligt redovisa och diskutera arbetet.
F Underkänt	Bristande respekt för överenskommelser, betydande osjälvständighet, eller ohörsamhet för handledning. Oförmåga eller ovilja att inhämta nya kunskaper.	Stora brister i färdigheter, eller betydande kvarvarande brister i metodik trots anmodan, eller stora brister i den vetenskapliga motiveringen av problemställningen.	Kvarvarande brister i den skriftliga rapporten trots anmodan, eller betydande oförmåga att muntligt presentera eller diskutera arbetet.



Några frågor om kursupplägget?



DEL 2 - METODER



Metod översikt

- I ditt projekt har du definierat ett problem att undersöka, och du behöver någon problemlösningsaktivitet för att besvara problemet.
- Det är detta som menas med metod.
- En metod är ett beprövat och strukturerat sätt att angripa och besvara ett problem.
- Det finns ett flertal metoder du kan använda, och vi vill titta närmare på några av de vanligaste.

Kvalitativ och Kvantitativ

- Metoder delas ofta upp i två kategorier: kvalitativa och kvantitative
- Det har att göra med vilka typ av data som samlas in:
 - Kvantitativ data: siffror, till exempel exekveringstid eller temperatur
 - Kvalitativ data: icke-numerisk, till exempel fritextsvar i ett frågeformulär

Vanliga metoder

- Kontrollerade experiment
- Undersökning med frågeformulär
- Intervju
- Fallstudie
- Systematisk litteraturundersökning
- Verifiering och Validering

Kontrollerade experiment

- Systematiskt testa någonting (system, algoritm, ...) i en kontrollerad miljö.
- Samla in mätdata för att bevisa eller förkasta en hypotes/proposition.
- Vi måste definiera:
 - Beroende variabler: vad som ska mätas
 - Oberoende variabler: indata som påverkar den beroende variabeln och som ändras i experimentet

Kontrollerade experiment

- Exempel:
 - Mäta körtiden (exekveringstid) för två olika parallella algoritmer som körs på olika antal processorkärnor.
 - Beroende variabel: körtid
 - Oberoende variabler: antal processorkärnor, algoritm 1 eller 2

Kontrollerade experiment

- Ett experiment ska, om möjligt, utföras i en kontrollerad miljö.
- Vad som bör beskrivas:
 - Hårdvarukonfiguration (CPU typ, minne, ...)
 - Mjukvarukonfiguration (operativsystem, viktig programvara som finns installerad, ...)
 - Beroende och oberoende variabler
 - Vilka verktyg som används för mätningarna
- Detta gör det möjligt att upprepa experimentet.

Undersökning

- En undersökning används för att få reda på personers uppfattning om någonting.
- Den består av:
 - Deltagare (ett urval av eller en hela population, till exempel 20 studenter på LNU)
 - En metod för att samla in data (ett frågeformulär är vanligt)
 - Individuella frågor som ger data som kan analyseras

Undersökning

- Det finns två vanliga skalor för svar på frågor.
- En *bedömningsskala* har ett intervall, till exempel mellan 1 och 5.
- Exempel:
 - på en skala från 1 till 5, hur bra tyckte du filmen var?
- Bedömningsskalor kan också vara förhållanden:
 - dubbelt så mycket, tre gånger så mycket, ...
- Exempel:
 - hur mycket extra skulle du kunna tänka dig att betala om programvaran hade en specifik funktion?

Undersökning

- I en *Likert-skala* bedömer deltagarna hur väl de håller med om någonting.
- En Likert-skala kan ha olika nivåer, men fem är vanligast:
 1. Starkt emot
 2. Emot
 3. Varken för eller emot
 4. För
 5. Starkt för

Undersökning

- I en undersökning är det viktigt att urvalet (deltagarna) är representativa för populationen som helhet.
- Exempel:
 - om du vill göra en undersökning på LNU studenter, är det inte representativt om ditt urval är fem manliga förstaårsstudenter på IT säkerhets programmet.
- Att ha ett för litet eller för homogent urval är ett vanligt fel i undersökningar.

Undersökning

- En undersökning kan även vara kvalitativ.
- I detta fall är svaren på frågorna flervalssalternativ eller fritext.
- Exempel:
 - vilka funktioner uppskattade du mest i programvaran?
- En undersökning kan också ha en blandning av kvantitativa och kvalitativa frågor (kallat en blandad metod).

Fallstudie

- En nära, grundlig och detaljerad undersökning av ett specifikt fall.
- Ett fall kan vara:
 - En individ (en expert inom området)
 - En organisation eller ett företag
 - En händelse (ett intrång i ett IT system)
 - En handling (användare testar nya funktioner i en programvara)
- Data samlas in genom observationer, intervjuer, från dokument och manualer, etc.

Fallstudie

- Den insamlade data analyseras och slutsatser dras om fallet.
- Exempel:
 - Ett företag vill testa två olika användargränssnitt för en databashanterare.
 - Användare testar båda gränssnitten.
 - Data samlas in genom observationer (ser användarna stressade eller frustrerade ut?) och genom ett frågeformulär (vad tycker användarna om respektive gränssnitt?)

Intervju

- Används för att få reda på personers uppfattning om någonting.
- Likheter med en undersökning, men:
 - Undersökning: många deltagare, korta och enkla svar
 - Intervju: en eller ett fåtal deltagare, väldigt detaljerade svar
- Intervjuer kan vara *öppna* eller *stängda*.

Intervju

- En öppen intervju är inte planerad i förväg.
- Inga förberedda frågor.
- Diskussion mellan intervjuaren och deltagaren.
- Intervjuaren leder diskussionen mot det som hen vill prata om.
- Deltagaren har kontroll över intervjun, och pratar om det hen tycker är viktigt.

Intervju

- En stängd intervju är planerad i förväg.
- Intervjuaren har förberett en uppsättning frågor att ställa till deltagaren.
- Intervjuaren har kontroll över intervjun.
- Fördel: lättare att upprepa på flera deltagare.
- Nackdel: deltagaren kan tycka att saker som det inte frågas om är viktigare än det som forskaren vill ta upp.

Intervju

- Oavsett typ av intervju är det viktigt att inte enbart lite på intervjuarens minne.
- Det är omöjligt att komma ihåg allt som har sagts under en intervju.
- Det finns olika sätt att komma ihåg vad som sagts:
 - Ta anteckningar: tidskrävande, och missar ofta en hel del som sägs.
 - Ljudinspelning: rekommenderat. Kan enkelt göras med en smartphone idag.
 - Videoinspelning: fångar kropps- och ansiktsuttryck som ljudinspelning missar. Kräver lite mer jobb, men det bästa alternativet.

Systematisk Litteraturstudie

- SLS kallas en andra-gradens studie.
- Ingen egen forskning görs.
- I stället samlas andras forskning inom området in och sammanställs.
- Forskningen summeras och egna slutsatser av andras forskning dras.

Systematisk Litteraturstudie

- I SLS samlas data in på ett systematiskt sätt (vad som ska tas med och inte tas med), och oftast behandlas en stor mängd artiklar.
- Först måste vi definiera hur vi ska söka efter artiklar:
 - Vilka databaser och vilka söktermer
 - Vilka konferenser eller tidskrifter som är intressanta
- Artiklar tas med eller ej baserat på fördefinierade *inclusion* och *exclusion* kriterier.
 - Kontrolleras oftast genom att läsa Abstract

Systematisk Litteraturstudie

- SLS är vanlig inom medicinsk- och hälsoforskning.
- Exempel är för att utvärdera effekt och biverkningar av en medicinsk behandling eller tillsatser i mat.
- SLS är också användbart i datavetenskap, till exempel för att hitta för- och nackdelar med olika utvecklingsprocesser från ett stort antal företag.

Verifiering och Validering

- Verifiering och validering används för att kontrollera om en programvara uppföljer krav och att den gör det den är ämnad att göra.
- Du startar med ett antal krav (både funktionella och icke-funktionella) som beskriver vad systemet ska göra.
- Exempel:
 - **R1:** Systemet ska kunna skapa arrayer med slumpmässiga heltal (mellan ett anggett min och max värde) av valfri längd (max 10 miljoner värden)
 - **R2:** Systemet ska kunna sortera en array med heltal med bubblersortering och mäta exekveringstiden
 - ...

Verifiering och Validering

- I verifieringsfasen testar du om systemet möter alla krav genom manuella eller automatiska tester.
- Varje krav måste ha minst ett test.
- För varje test ska du ange:
 - Test id, till exempel T1
 - Vilket krav som testas, till exempel R1
 - Hur testet körs (automatiskt i en testsvit, eller hur det körs manuellt)
 - Förväntat resultat, vilket används för att se om testet klarades eller ej

Verifiering och Validering

- I valideringsfasen kontrolleras att systemet uppföljer förväntningarna hos slutanvändare eller kunder.
- Detta involverar oftast att låta slutanvändare eller kunder testa systemets funktionalitet och ge feedback till utvecklarna.
- I examensarbetet görs validering typiskt av det företag du gör ditt projekt på.

Verifiering och Validering

- Skillnaden mellan verifiering och validering beskrevs kortfattat av Barry Boehm 1979:
 - Verifiering: “Are we building the product right?”
 - Validering: “Are we building the right product?”

Data



Data

- Beroende på valet av metod, samlas olika typer av data in.
- Datan ska presenteras i din rapport.
- Du ska också dra slutsatser från datan, vilket oftast kräver att du gör någon form av analys.
- Olika typer av data kräver olika metoder för analys och presentation.

Numeriska Data

- Exempel:
 - Två algoritmer, A och B, som löser samma problem jämförs.
 - Vilken algoritm som används, A eller B, är den oberoende variabeln
 - Körzeit (exekveringstid) och minnesanvändning är de beroende variablerna.
 - För att minimera hot mot validiteten upprepas experimentet 5 gånger och medelvärden räknas ut.

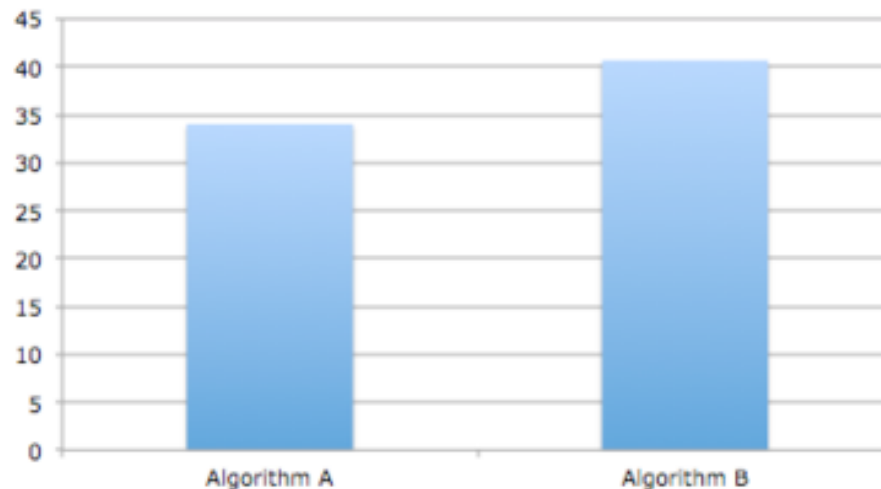
Numeriska Data

- Mätdata presenteras i en tabell:

Run	Algorithm A		Algorithm B	
	Exec. Time (ms)	Memory (MB)	Exec. Time (ms)	Memory (MB)
1	34.5	0.17	42.3	0.21
2	37.3	0.18	41.9	0.23
3	31.2	0.17	38.6	0.22
4	32.0	0.17	39.7	0.22
5	35.1	0.18	40.8	0.23
Average	34.02	0.17	40.66	0.22

Numeriska Data

- För att göra det lättare att jämföra resultaten, kompletteras ofta tabeller med diagram:



Har frågan besvarats?

Run	Algorithm A		Algorithm B	
	Exec. Time (ms)	Memory (MB)	Exec. Time (ms)	Memory (MB)
1	34.5	0.17	42.3	0.21
2	37.3	0.18	41.9	0.23
3	31.2	0.17	38.6	0.22
4	32.0	0.17	39.7	0.22
5	35.1	0.18	40.8	0.23
Average	34.02	0.17	40.66	0.22

- Kan vi besvara frågan:
 - Är algoritmen A snabbare än B för ...?
- Ja, mätningarna från ett experiment i en kontrollerad miljö verifierar att A är snabbare än B.
- Det finns dock en hake:
 - Ibland är medelvärdena så lika att skillnaden kan bero på slumpmässiga faktorer i stället för på den oberoende variabeln.



Statistisk Signifikans

- För att verkligen vara säkra kan vi använda ett statistiskt test för att kontrollera om två medelvärden skiljer sig åt.
- Detta kallas ett T-test, och kan göras i Excel eller andra verktyg:
 - <https://sourceforge.net/projects/javastats/>
- Testet resulterar i ett P-värde.
- Om P-värdet är lika med eller lägre än 0.05 kan vi säkert säga att det finns en skillnad.
- Resultatet är då *statistiskt signifikant*.

Statistisk Signifikans

- T-test på vårt exempel:

	A	B	C
1		A	B
2	Run	Exec. Time (ms)	Exec Time (ms)
3	1	34,5	42,3
4	2	37,3	41,9
5	3	31,2	38,6
6	4	32	39,7
7	5	35,1	40,8
8	Average	34,02	40,66
9	T-test	0,0009	
10			

- P är mindre än 0.05. Resultatet är statistiskt signifikant.

Statistisk Signifikans

- Samma exempel men med andra exekveringstider:

	A	B	C
1		A	B
2	Run	Exec. Time (ms)	Exec Time (ms)
3	1	34,5	32,3
4	2	37,3	32,8
5	3	31,2	38,6
6	4	32	39,7
7	5	35,1	33,1
8	Average	34,02	35,3
9	T-test	0,5259	

- P värdet är större än 0.05.
- Resultatet är inte statistiskt signifikant, även om det är en skillnad mellan de beräknade värdena.
- Vi kan därmed säga att det inte finns en skillnad mellan medelvärdena.

Statistisk Signifikans

- Det finns ett antal olika test som används vid olika tillfällen:
 - Två medelvärden: T-test
 - Tre eller fler medelvärden: ANOVA
 - Två medelvärden, ej normalfördelade: Wilcoxon Signed-Ranks
 - Tre eller fler medelvärden, ej normalfördelade: Kruskal-Wallis
- Likert- och bedömningsskalor är oftast inte normalfördelade.
- Om du behöver använda statistiska tester på dina resultat, be din handledare om hjälp.
- De flesta tester finns i verktyget JStats:
 - <https://sourceforge.net/projects/javastats/>

Icke-numeriska Data

- Data från kvalitativa metoder är lite svårare att analysera och presentera.
- Eftersom icke-numeriska data kan se väldigt olika ut, bör du diskutera med din handledare hur du bäst kan analysera och presentera din data.

Icke-numeriska Data

- Exempel:
 - Data från ett frågeformulär där företag svarar på vilka utvecklingsmetoder de använder:

Method	Company
Waterfall	C
UML	A, B, C
Scrum	B
Agile	A
XP	A, C

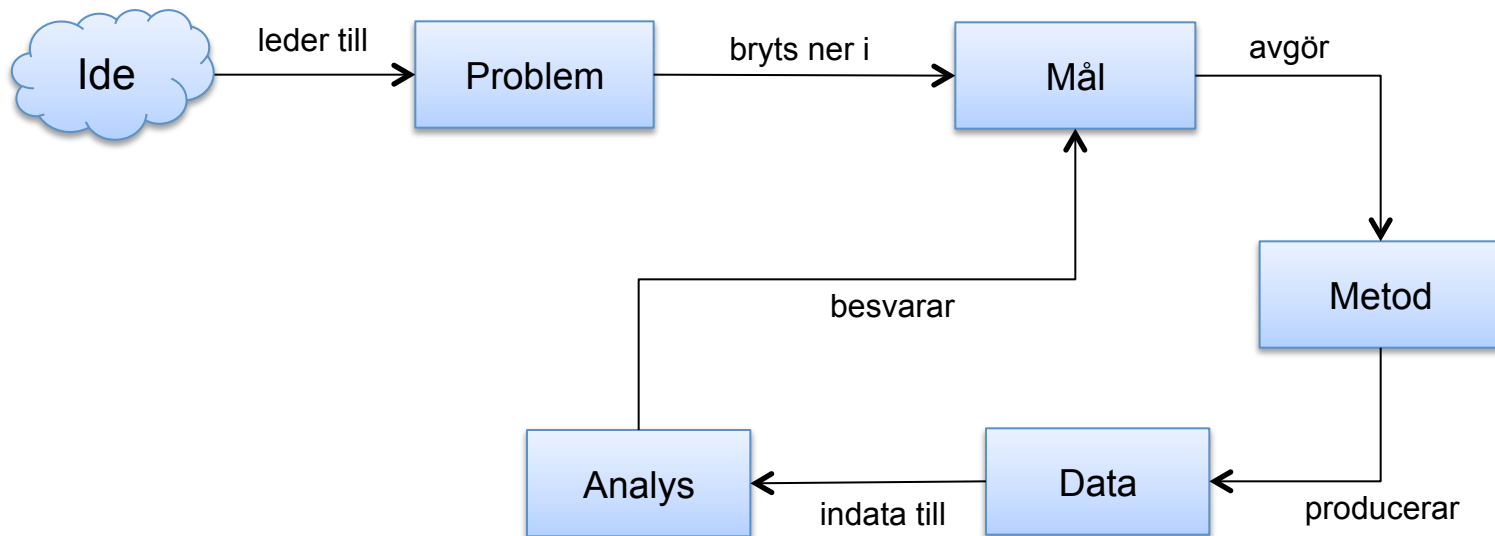
Validitet och Tillförlitlighet



Validitet och Tillförlitlighet

- För att besvara ditt problem använder du en metod, samlar in (och eventuellt analyserar) data, och drar slutsatser från datan.
- Det är viktigt att dina slutsatser är korrekta, det vill säga att de stöds av hur du genomfört ditt projekt och den data du samlat in (validitet).
- Med tillförlitlighet menas om andra kommer att få samma resultat som du om de replikerar ditt arbete.
- Problem med tillförlitligheten kan uppstå om du använt en felaktig metod för att samla in data.

Sammanfattning



Examensarbetscykeln

Frågor?

