

## Studiebrev om eksamensopgaven

### Det formelle

Eksamensformen for faget Grundlæggende Programmering kaldes "Mundtlig prøve kombineret med et praktisk produkt og en skriftlig opgave", og er i studieordningens tillæg "Prøveformer og bedømmelsesgrundlag" beskrevet således:

Det er en mundtlig prøve, hvor den studerende får en karakter på baggrund af den mundtlige eksamination. Den godkendte skriftlige opgave er sammen med den praktiske opgave oplæg til og udgangspunkt for den mundtlige prøve.

En skriftlig opgave må samlet set være på maksimum 6 normalsider svarende til 6 x 2400 tegn inkl. mellemrum. Figurer, billeder og grafik tæller hver som ét tegn. En opgave kan således godt fylde mere end 6 fysiske A4-sider.

Den mundtlige eksamen er på 30 minutter i alt, hvori både det praktiske produkt og den skriftlige opgave indgår som en del af bedømmelsen. Eksaminationen har følgende fordeling, og der oplyses ikke delkarakterer:

- 1/4 til et mundtligt oplæg
- 2/4 til eksaminationssamtale
- 1/4 til eksaminators og censors votering samt en tilbagemelding til den studerende.

Det betyder på almindeligt dansk, at der gives én karakter, hvori indgår både det praktiske produkt, den skriftlige opgave og præstationen ved eksaminationen. De nævnte "1/4 til et mundtligt oplæg", betyder at du har de første 7-8 minutter af eksamenstiden til at præsentere dit produkt og din opgave.

Formelle krav til opgavers forside (iflg. studieordningens tillæg):

Til alle opgaver skal benyttes et titelblad, som skal indeholde følgende oplysninger:

1. Retning og fagets navn (Akademiuddannelse i Informationsteknologi og Grundlæggende Programmering)
2. Titel og evt. undertitel (Titel kan tages fra opgavebeskrivelsen)
3. Den studerendes fulde navn (det er dig 😊)
4. Vejleders navn (det er Kristian og mig 😊)
5. Uddannelsesinstitutionens navn (SmartLearning)
6. Måned og år (maj 2016)
7. Evt. påtegning hvis materialet skal håndteres fortroligt
8. Antal tegn incl. mellemrum (figurer tæller for ét tegn)

Opgaven skal have en læsevenlig struktur og være skrevet på korrekt dansk. Opgaven kan dog i særlige tilfælde, ifølge aftale med vejleder/uddannelsesinstitution, skrives på engelsk.

### **Det var de formelle beskrivelser - nu til hvordan det udmøntes i Grundlæggende Programmering:**

Iflg. de formelle beskrivelser kan et praktisk produkt defineres på mange forskellige måder. I Grundlæggende Programmering må det logisk være en applikation programmeret i C#.

Rammerne er, at applikationen forventes at indeholde følgende:

- Brugerinput, typisk via et eller flere skærbilleder
- Validering af brugerinput
- Behandling af data, f.eks. beregninger
- Anvendelse af en form for datastruktur (array, arraylist eller andet)
- Respons til brugeren
- At der vises en listning / oversigt (f.eks. deltagerliste, vareliste, reservationsliste)
- Navigation mellem mindst 3 skærbilleder, evt. ved anvendelse af menu
  - applikationen skal foruden hovedformen have en About-form
  - de øvrige skærbilleder kan evt. være i form af tabs på hovedformen
- Det forventes at anvendelse af klasser og objekter indgår i opgaven

Den skriftlige opgave skal være i form af en mini-rapport der indeholder følgende

- Indledning, beskrivelse af hovedemne, problemstilling og evt. afgrænsninger
- Overvejelser om hvordan opgaven tænkes løst
- Overvejelser om programmeringsmæssige udfordringer
- Overvejelser mht. sikring af kvaliteten, herunder tests
- Beskrivelse af anvendte klasser
- Perspektivering af opgaven. F.eks. hvad du ville ændre og/eller tilføje
  - hvis du havde mere tid
  - hvis du havde mere viden
  - evt. andre forhold
- Konklusion på opgaven
  - i hvilket omfang fik du løst problemet / opgaven.

Det er vigtigt at der i forbindelse med overvejelserne argumenteres for de beslutninger og valg der træffes.

Det er naturligvis også vigtigt at der er sammenhæng mellem det du beskriver i opgaven, og den måde applikationen implementeres på.

Som nævnt i starten må den skriftlige opgave være på max. 6 normalsider

Applikationen og den skriftlige opgave kan laves individuelt eller af flere studerende sammen; men eksaminationen og karaktergivningene er individuel.

Applikationen skal kunne afvikles i Visual Studio Community 2015.

### Hvad skal du gøre nu?

Du skal selv definere din opgave. Det kan være en opgave relateret til en virksomhed, til en organisation, til en forening, eller til en interesse du har.

Lav en overordnet beskrivelse af opgaven og send den til mig pr. mail ([per@perkjeldsen.dk](mailto:per@perkjeldsen.dk)). Jeg skal formelt godkende, at opgaven opfylder de formelle krav og er inden for rammerne nævnt ovenfor.

Der vil blive uploadet et eksempel på en opgavebeskrivelse, som du kan bruge til inspiration og ide. Der vil ligeledes blive uploadet to eksempler på rapport/skriftlig opgave, som du er velkommen til at bruge som inspiration.

Skulle du være helt tom for ideer til en opgave, kan du vælge at bruge den uploadede opgavebeskrivelse som grundlag for din eksamensopgave.

Opgaven skal afleveres på [Wiseflow](#) senest den 30.05.2016 kl. 12:00. Som sædvanelig med alle filer - også den skriftlige opgave - pakket sammen i én zip-fil, så censor kun skal downloade én fil pr. opgave.

Censorer hader at skulle bøvle med teknik og downloade mange filer, og det er jo i alles interesse at holde censor i godt humør 😊

Der kommer et link på SmartLearning Online til en vejledning i hvordan Wiseflow anvendes

### Tidsplan for eksamensopgaven

11.05.2016	Opgaveformulering skal være godkendt
Uge 20-21	Du arbejder med opgaven.
30.05.2016	Opgaveløsninger afleveres på <a href="#">Wiseflow</a> senest kl. 12:00
07.06.2016	Besked om godkendelse af opgaven
Uge 24	Eksamen - der vil senere komme besked om hvor eksamen vil finde sted, samt om muligheden for eksamen via Adobe Connect