

INF100

Prinsipper i

informasjonsbehandling

Hans Ekkehard Plesser

Institutt for matematisk realfag og teknologi
Universitetet for miljø- og biovitenskap

Forelesning nr 9 (del 1) — 7. november 2013



Dagens program

- ▶ Regneark
- ▶ Databehandling: Historie (se plansjer Del 2)
- ▶ Oppbygging av datamaskiner (se plansjer Del 2)

Regneark

Regneark

Regneark

- ▶ Repetisjon: Ark, celler, referanser, formler
- ▶ Råd: Dokumentasjon, validering, beskyttelse
- ▶ Eksempel: Banksparing med regneark og Python

Referanser

- ▶ Gir tilgang til verdier i andre celler
- ▶ består av `kolonne rad`, f eks A2, D12, BB1
- ▶ Relative referanser
 - ▶ tilpasses ved kopiering
 - ▶ `=B2` kopieres fra C2 til D3 → blir `=C3`
- ▶ Absolutte referanser
 - ▶ merkes med `$` foran kolonne, rad eller begge
 - ▶ `=$B2` kopieres fra C2 til D3 → blir `=$B3`
 - ▶ `=B$2` kopieres fra C2 til D3 → blir `=C$2`
- ▶ Områder
 - ▶ `A3:D6` viser til et rektangulært område
 - ▶ typisk bruk: `=Sum(A3:A10)`
- ▶ Navngitte felt
 - ▶ navn til celle defineres gjennom
`Insert>Names>Define`

Formler

- ▶ begynner alltid med =
- ▶ enkle regneopersasjoner = $0,25 * \text{Sum}(A2:A5)$
- ▶ omfattende analyser
$$= \text{IF}(\text{AND}(\text{H2} < \text{I2}; \text{H2} < \text{J2}); (\text{I2} + \text{J2}) / 2;$$
$$\text{IF}(\text{AND}(\text{I2} < \text{H2}; \text{I2} < \text{J2}); (\text{H2} + \text{J2}) / 2; (\text{I2} + \text{H2}) / 2))$$
- ▶ aktualiseres når hvilken som helst celle forandres
- ▶ mange matematiske og logiske funksjoner tilgjengelig
 - Sum() sum av verdier
 - Average() middelverdi
 - Sin() sinus
 - Today() aktuell dato
 - If(Test ; A ; B) gjør A eller B avhengig av Test
- ▶ bruk assistenten & les hjelp til å bygge formuler!
- ▶ ikke-tall verdier “ignorerers” i regnefunksjoner
- ▶ funksjonsnavn er språkavhengig

Dokumentasjon av regneark

- ▶ regnearket skal være selvforklarende for brukeren
- ▶ “revisor” og “utvikler” skal få informasjon om oppsett av utregninger og begrunnelser for utforming av formler
- ▶ dokumentasjon for brukeren
 - ▶ tekstfelt: bruk `Merge` for å slå celler sammen, under `Format>Cells>Alignment` **velg** `Wrap text automatically`, evt også plassering
 - ▶ hjelp til enkelte celler: gi beskrivelse under `Data>Validity>Input Help`
- ▶ dokumentasjon for utvikler/revisor
 - ▶ bruk notater til celler
 - ▶ opprettes med `Insert>Note`
 - ▶ celler med notat har en liten rød merke
 - ▶ notat vises når pekere er over cellen
 - ▶ slett notat med høyreklikk `>Delete Note`
 - ▶ rediger notet med høyreklikk `>Show Note` og dobbelklikk i noten

Kontroll av inngaver

- ▶ GIGO: Garbage in, garbage out
- ▶ kontroller inngaver for å beskytte bruker mot feil
- ▶ eksempler på feilkilder
 - ▶ manglende verdier
 - ▶ desimalpunkt vs desimalkomma
 - ▶ negative tall eller null, der regningen forutsetter positive tall
 - ▶ tekst istedenfor tall
- ▶ Løsning: datavalidering med `Data>Validity`
- ▶ Defineres for hver inngavecelle
- ▶ `Criteria` definerer hva som er tillatt
- ▶ bruk `Data: greater than` og `Value: -infinity` for tall av vilkårlig størrelse
- ▶ `Error Alert` definerer hva som skjer ved feil
- ▶ bruk `Action: Stop` for å være på sikker side

Feilsøking og validering

- ▶ kontroller hver enkelt regneskritt, ta kontrollregning i hode/på papir/med lommekalkulator
- ▶ formuler dine forventninger om resultatet for å ha et grunnlag for å kunne kontrollere resultat fra regneark
- ▶ hvis mulig, finn formel for sluttresultatet for å kunne sammenlikne med resultatet fra regnearket
- ▶ bruk “enkle” data til de første testene, slik at hoderegning blir lettere
- ▶ gjennomfør mange “test cases”, som også tester spesielle situasjoner/grensetilfeller
- ▶ får du overraskende resultater, ikke tro på dem før du ha funnet en logisk forklaring
- ▶ **La en annen kontrollere ditt regneark!**

Feilsøking—Verktøy

- ▶ Små grønne trekanter i “mistenkelige” celler (bare Excel)
- ▶ View>Value Highlighting viser tall i blått, formler i grønt
- ▶ Tools>Detective>Trace Precedents viser alle felt som inngår i formelen i en celle
- ▶ Tools>Detective>Trace Dependents viser alle felt som benytter verdien i en celle

Beskyttelse av innhold

- ▶ Formål med beskyttelse
 - unngå utilsiktede endringer** beskyttelse uten
passord
 - hindre endringer uten autorisasjon** beskyttelse med
passord
 - skjule formler fra bruker** bruker må stole blind på oss
- ▶ Fremgangsmåte
 - ▶ beskyttelse defineres individuelt for hver celle som del
av celleformat: `Format>Cells>Cell Protection`
 - ▶ i utgangspunkt er beskyttelse aktivert for alle celler
 - ▶ cellebeskyttelse blir **virksom bare hvis Sheet
beskyttelse er aktivert** via `Tools>Protect
Document>Sheet`
 - ▶ la passord stå tomt for beskyttelse uten passord
 - ▶ beskytt hele dokumentet med `Tools>Protect
Document>Document`
- ▶ NB: “Protect Document” passord er veldig lett å knekke
i Excel

Eksempel: Banksparing med Calc og Python

Oppgave

Beregn verdiutviklingen over 10 år for kr 100.000 som er satt inn på høyrentekonto. Ta hensyn til inntektsskatt. Verdien, renter og skatter skal beregnes og tabuleres årlig. Oppgaven skal løses både med OO Calc og Python.

| | |
|------------------------|-----|
| Rente på høyrentekonto | 3% |
| Inntektsskatt | 28% |

Løsning

- ▶ Diskusjon på tavle
- ▶ Calc-implementering som regneark i timen
- ▶ Viktige aspekter ved Calc-løsningen
 - ▶ Brukerinformasjon i toppen
 - ▶ Alle celler låst, unntatt rentesats, skattesats, startbeløp
 - ▶ Navngitte celler for rentesats, skattesats, startbeløp
 - ▶ Celler for rentesats og skattesats formatert som prosenttall
- ▶ Python-implementering i notebook