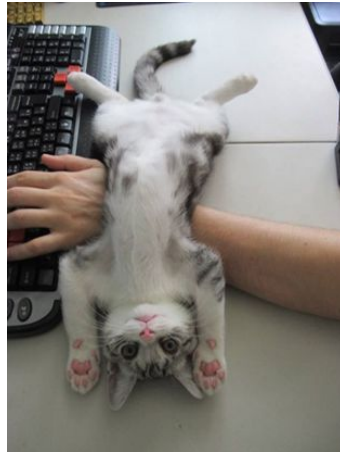


# 1. Programmering

- hvad er det?

Informatik C  
November-december 2023



# Informatik C - kort introduktion

## Informatik på C niveau:

- Informatik er et fag der beskæftiger sig med informationsteknologi.
- *“Faget beskæftiger sig med digitale data i et samspil mellem teori/model på den ene side og afprøvning/eksperiment på den anden.”*
- *“Faget øger elevernes evne til at forholde sig til den enkeltes, uddannelsens og samfundets brug af it gennem teoretisk indsigt i og praktisk arbejde med at skabe forskellige former for it-systemer. Eleverne får herigennem indsigt i faget i forhold til egne styrker og interesser med henblik på uddannelses- og karrierevalg.”*

Hvilket IT bruger I? - snak sammen med  
sidedemanden



# Informatik C – de faglige mål

- Konstruktion af et it-system som løsning til en problemstilling
  - Lave et mindre it-system helt eller delvist med lidt kode og/eller pseudokode
  - Et it-system består af tre lag (tre-lagsarkitektur): Præsentationslaget, datalaget, logiklaget
- It-systemers og menneskelig aktivitets gensidige påvirkning
  - Hvordan påvirker it-systemer menneskers gøren og laden. Lever vi på en anden måde pga. it.
- It-sikkerhed, netværk, arkitektur
  - Hvordan et it-system er bygget op (arkitektur)
  - Hvordan internettet er bygget op (netværk)
  - It-sikkerhed (din og andres privatliv på nettet f.eks. ect.)
- Repræsentation og manipulation af data
  - Hvordan gemmer man data og finder det igen
  - Databaseopbygning
- Programmering
  - Simple programmeringsforståelse, såsom for-løkker, if-else-sætninger osv.
- Interaktionsdesign
  - Design af præsentationslaget: hensigtsmæssig placering af knapper, tekst, valg af farve, nem navigering osv.
- Innovation
  - Hvad er et innovativt it-system?
  - Modeller til at kunne identificere innovation

# Informatik C – de faglige mål

- Konstruktion af et it-system som løsning til en problemstilling
  - Lave et mindre it-system helt eller delvist med lidt kode og/eller pseudokode
  - Et it-system består af tre lag (tre-lagsarkitektur): Præsentationslaget, datalaget, logiklaget
- It-systemers og menneskelig aktivitets gensidige påvirkning
  - Hvordan påvirker it-systemer menneskers gøren og laden. Lever vi på en anden måde pga. it.
- It-sikkerhed, netværk, arkitektur
  - Hvordan et it-system er bygget op (arkitektur)
  - Hvordan internettet er bygget op (netværk)
  - It-sikkerhed (din og andres privatliv på nettet f.eks. ect.)

# Informatik C - de faglige mål

- Repræsentation og manipulation af data
  - Hvordan gemmer man data og finder det igen
  - Databaseopbygning
- Programmering
  - Simple programmeringsforståelse, såsom for-løkker, if-else-sætninger osv.
- Interaktionsdesign
  - Design af præsentationslaget: hensigtsmæssig placering af knapper, tekst, valg af farve, nem navigering osv.
- Innovation
  - Hvad er et innovativt it-system?
  - Modeller til at kunne identificere innovation

# Faglige mål vi arbejder med i dette forløb:

- Konstruktion af et it-system som løsning til en problemstilling
  - Lave et mindre it-system helt eller delvist med lidt kode og/eller pseudokode
  - Et it-system består af tre lag (tre-lagsarkitekturen): Præsentationslaget, datalaget, logiklaget
- It-systemers og menneskelig aktivitets gensidige påvirkning
  - Hvordan påvirker it-systemer menneskers gøremål og læring. Lever vi på en anden måde pga. it.
- It-sikkerhed, netværk, arkitektur
  - Hvordan et it-system er bygget op (arkitektur)
  - Hvordan internettet er bygget op (netværk)
  - It-sikkerhed (din og andres privatliv på nettet f.eks. ect.)
- Repræsentation og manipulation af data
  - Hvordan gemmer man data og finder det igen
  - Databaseopbygning
- Programmering
  - Simple programmeringsforståelse, såsom for-løkker, if-else-sætninger osv.
- Interaktionsdesign
  - Design af præsentationslaget: hensigtsmæssig placering af knapper, tekst, valg af farve, nem navigering osv.
- Innovation
  - Hvad er et innovativt it-system?
  - Modeller til at kunne identificere innovation

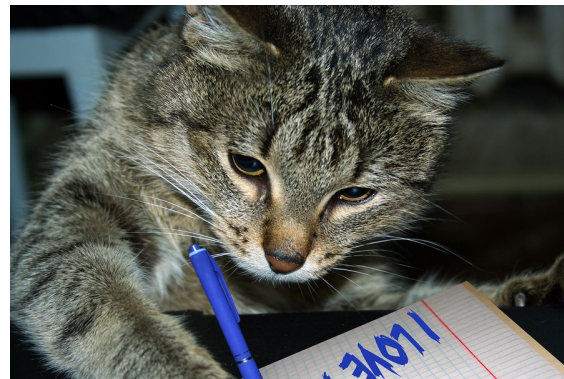
Video: "You should learn to program"





# Introduktion til logbog

- I Informatik er det et krav, at man fører en logbog.
- Læreren skal have adgang til logbogen.
- Logbogen er et meget nyttigt værktøj på mange forskellige måder.
- I slutning eller nogle gange undervejs i timerne, vil I blive bedt om, at skrive en logbog.



# LOGBOG

- **Øverst: Fulde navn, fag (informatik)**
- **Skriv i Word, Google Docs ect. - så længe dokumentet kan laves om til en pdf.**
- **Start et nyt afsnit med dags dato**
- **Logbog kan indeholde alt: noter, refleksionskrivning, billeder, MEN ISÆR: egenproduktion: kodning, App Lab-projekter, lydfile/podcast/videoproduktioner (YouTube-links) osv.**
- **Op til standpunkt: send en pdf-kopi af din logbog til mig - jeg beder om det, når det er tid.**
- **OBS: du er selv ansvarlig for at gemme og back'e din logbog op!**

LOGBOG informatik 2018-2019

Navn Navnsen|

---

D. 7. maj 2019

Placering: Ændrer opfattelse af IT-produktets position - nye kunder kan være med. Ud fra de 4 p'er:

Produkt: Hvad kan vi tilbyde?

Proces: Hvordan skaber vi det?

Position: Hvem skaber vi det til?

Paradigme: Hvordan vinkler vi det vi gør?

Vi snakkede om bekendtgørelsen og de faglige mål indenfor informatik  
Konstruktion af it-system som løsning til en problemstilling: Vi lavede vores eget it-system (slikland)

# Robotøvelsen!

Øvelsen går ud på, at I skal lave en model af en virkelig handling som i dette tilfælde er at børste tænder.

I skal skrive ned, hvad der sker trin for trin, når man børster tænder. De første trin skal være:

1. Stå foran badeværelsesskabet.
2. Tag venstre arm frem
3. Brug venstre arm og venstre hånd til at åbne badeværelsesskabet
4. Osv.
5. Osv.

Bagefter afprøver vi, om modellen kan få en robot til at børste tænder. Brug 5 minutter på at skrive jeres trin ned.

# Refleksionskrivning

Skriv i logbog:

- Hvad tænker du om forskellen mellem den menneskelige måde at tænke på (kognition) og computerens måde at tænke på (computational thinking). Fik robotøvelsen sat nogle tanker i gang om dette hos dig?
- Tror du, at der vil være forskel på den menneskelige kognition og computerens kognition i fremtiden?
- Tror du menneskeheden kan opnå mere, hvis vi alle får en forståelse for computerens tankegang og evt. lærer programmering?

Skriv i 5 minutter.

My army is ready, we attack at  
nightfall



# Forløbets programmeringsdilemma/-opgave:

Din lokale restaurant/take-awaysted/kantine har brug for hjælp:

Tiden er knap, kunderne er mange og personalet er i undertal. I skal derfor hjælpe firmaet med et stk. IT-system.

- Man skal kunne bestille mad gennem IT-systemet

Brainstorm individuelt eller med din sidemakker om, hvilket IT-system I gerne vil bygge med App Lab, som kan hjælpe.

# De vigtigste begreber inden for programmering i Informatik C:

- Variabler
- Løkker
- Forgreninger
- Funktioner
- Sekvenser





# Variabler

En variabel er *et navngivet sted*, hvor du kan gemme noget.

Man kan sammenligne en variabel med *en kasse med en etiket på*. I kassen kan du lægge forskellige ting, fx tal, bogstaver m.m. Du kan ændre tallene eller bogstaverne, fjerne dem eller udskifte dem med nye.

En variabel skal have et navn, som du bestemmer - uden mellemrum.

<https://informatikforalle.ibog.forlagetcolumbus.dk/?id=285#c385>



Jetpack Cat

# Variabler

I tekstmodus kan en variabel se sådan ud:

```
var a = 100;
```

Dette kan læses: Definer en ny variabel a og sæt den lig 100.

Man behøver ikke give en variabel en værdi. Man kan nøjes med at definere den.

NB! En variabel skal erklæres, inden du kan bruge den.

Den gør man ved at skrive VAR foran.



# Variables

The screenshot shows a web development IDE interface. At the top, there's a teal header with "Untitled Project" and "Saved a few seconds ago". Below this are buttons for "Rename", "Share", "Remix", and "Create". The main workspace is divided into three sections: "Code", "Design", and "Data". The "Code" section is active, showing a list of code blocks in the "Toolbox" and a "Workspace" area. The "Toolbox" lists categories: UI controls, Data, Control, Variables, Canvas, Turtle, Math, and Functions. The "Variables" category is selected, showing several code blocks: `var x = 0;`, `var x;`, `x = 0;`, `var x = prompt("Enter a va");`, `var x = promptNum("Enter a");`, `console.log(message);`, `console.clear();`, `var str = "Hello World";`, `str.substring(start, end);`, and `str.indexOf(searchValue);`. The "Workspace" area is empty. At the bottom, there's a "Run" button and a "Debug Console" section with "Show Debug Commands" and "Debug Console" options.

Untitled Project  
Saved a few seconds ago

Rename Share Remix Create

Code Design Data

screen1

Run

Toolbox

- UI controls
- Data
- Control
- Variables
- Canvas
- Turtle
- Math
- Functions

```
var x = 0;
var x;
x = 0;
var x = prompt("Enter a va");
var x = promptNum("Enter a");
console.log(message);
console.clear();
var str = "Hello World";
str.substring(start, end);
str.indexOf(searchValue);
```

Workspace

Show Debug Commands Debug Console

# Variabler - deklarerer/opret

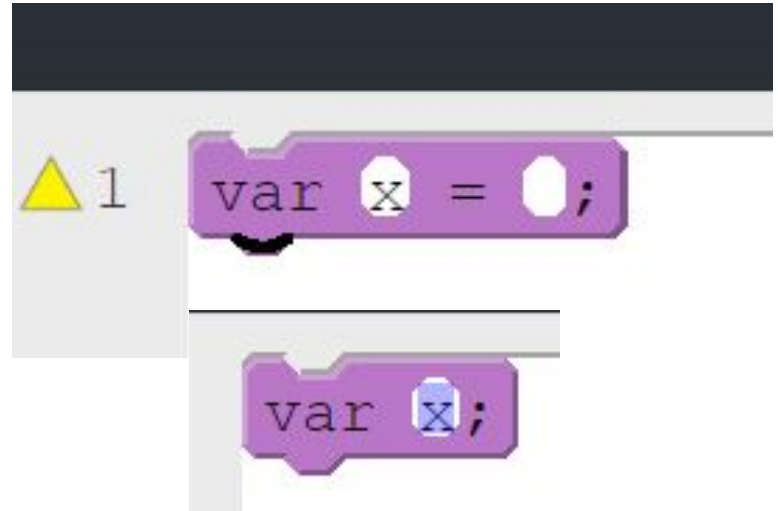
I App Lab (blokprogrammering)

Deklarerer først: VAR

Kun VAR første gang! (pakker kassen ud)

X = erstat med logisk navn (skriver navn på kassen)

Tildel værdi eller vent (Put noget i kassen eller vent - =)



# Variabler - efter oprettelse

## I App Lab (blokprogrammering)

1. linje: opretter variabel og kalder den for: **navn**
2. linje: putter en værdi ind i min variabel: en streng (gåseøjne"""). (Streng er ord, sætninger)
3. Ændrer værdien

KUN ÉN TING AD GANGEN

```
1 var navn;  
2 navn = "Michala Dahl Mensing";  
3 navn = "Julemanden";
```

# App Lab

Untitled Project  
Saved a few seconds ago

Rename

Share

Remix

Create ▼

Terminatormic ▼



Code Design Data

screen1 ▼

▶ Run

Toolbox



UI controls	Canvas
Data	Turtle
Control	Math
Variables	Functions

Workspace

🕒 Version History

</> Show Text

```
onEvent(id, type, callback)  
button(id, text)  
textInput(id, text)  
textLabel(id, text)  
dropdown(id, option1, etc)  
getText(id)  
setText(id, text)  
getNumber(id)  
setNumber(id, number)  
checkbox(id, checked)  
radioButton(id, checked)
```

English ▼



Show Debug Commands

Debug Console

🗑️ Clear

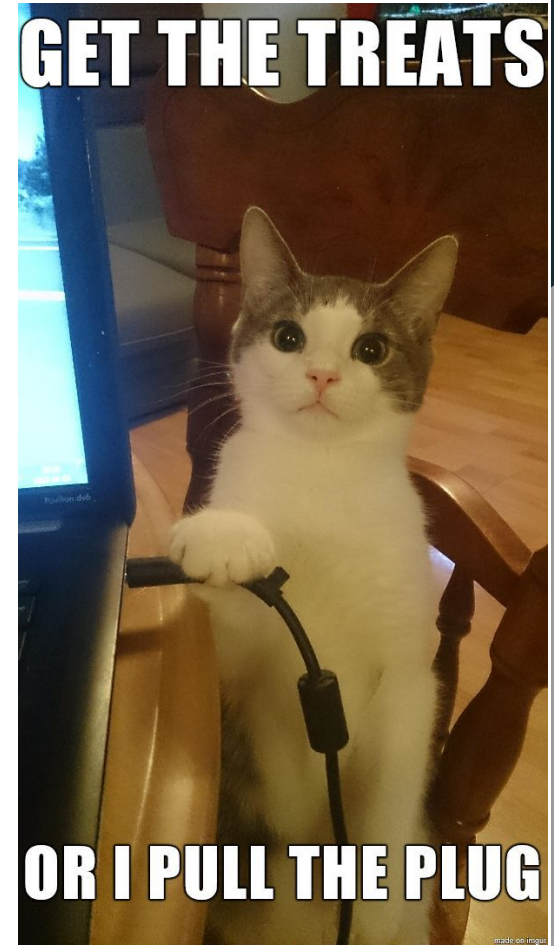
Watchers



# App Lab



App Lab - tutorial  
[https://studio.code.org/  
s/applab-intro/lessons/1  
/levels/1](https://studio.code.org/s/applab-intro/lessons/1/levels/1)





# App Lab - design af sprogapp



# Løkker



# Løkker

- Løkker er gentagelser af kode
- Smart hvis vi skal bruge den samme kode mange gange.

```
1 for ( var i = 0 ; i < 4 ; i++ ) {  
2  
3 }
```

opretter variabel med navn "i" - skal bruges til at tælle med

definerer hvor mange gange vi skal gentage - her 4 gange. Der står: så længe variabelen "i" er mindre end 4, så skal du blive ved med at udføre koden nedenunder.

Variabelen "i" skal plusse sig selv med 1.

(Dvs.: nu står der 0 i variabelen, men vi plusser med 1 og så står der 1 i variabelen "i".)

# Forgreninger



The image features a woman with dark hair, wearing a white and black striped sweater, speaking against a purple background. To her right, there are two educational graphics. The top graphic shows a row of ten stylized human figures: four orange, two pink, and three teal. A pink bracket underneath the first six figures is labeled "PG 13". The bottom graphic is a screenshot of a Scratch script editor, showing a sequence of code blocks: a "say" block, a "wait" block, a "say" block, a "wait" block, a "say" block, and a "wait" block, all with speech bubble icons.

# Forgreninger

- Også kaldet if/else.
- Bruges til at vælge imellem nogle valg i programmeringen - enten sker der dét eller også sker der dét her/enten går vi den ene vej ellers går vi den anden vej.
- Man kan tilføje uendeligt antal else if
- Man kan også bare nøjes med en if, uden else eller else if.

```
1 if ( ) {  
2  
3 } else {  
4  
5 }
```

```
1 var vejret = "solskin";  
2 if ( vejret == "solskin" ) {  
3     write("Tag solbriller på");  
4 } else {  
5     write("Lad solbrillerne blive hjemme.");  
6 }
```

# Funktioner



# Funktioner

- En måde at gentage kode på.
- Man skriver et lille program inde i koden.
- Man opretter/skriver sit lille program og giver det et navn.
- Når man skal bruge programmet, kalder man på det.

```
7 function tjekVejret() {  
8  
9 }  
10 myFunction ();
```

Skriver programmet og giver det et navn. (Her mangler jeg at skrive programmet.)

Her kalder jeg programmet, så det bliver udført.

# Funktioner

[https://studio.code.org/projects/applab/77HwRi7\\_Pv46Nu\\_ciWr7IT5iYp\\_uoFfnzk5utfadomM](https://studio.code.org/projects/applab/77HwRi7_Pv46Nu_ciWr7IT5iYp_uoFfnzk5utfadomM)

```
2 function tjekVejret () {  
3   if (vejret == "solskin") {  
4     write("Tag solbriller på");  
5   } else {  
6     write("Lad solbrillerne blive hjemme.");  
7   }  
8 }  
9 tjekVejret();
```



# Sekvenser (sequences)

- Robotlegen - tandbørsteøvelsen
- Kan sammenlignes med en opskrift man følger minutiøst
- F.eks.:
- `var navn = "Michala";`
- `var besked = "Hej ";`
- `var personligBesked = besked + navn;`

# Tutorials i App Lab

<https://studio.code.org/s/csp4-2022/lessons/1/levels/1>

<https://studio.code.org/s/csp4-2022/lessons/3/levels/1>

<https://www.youtube.com/watch?v=GDV4f2brxfE>

# “Informatik for alle”: “Guide til App Lab”

<https://studio.code.org/projects/applab/U4jhmmFLLwdDf7n-sChKkpH-CmfjuDYTrV6FgJoSw6I>

## First Semester

me in october:

me in january:

```
34 write("Du vandt en sekser!");
35 } else {
36 write("Desværre, der er kun gevinster på seksere. Prøv igen.");
37 }
38 //Funktion:
39 function login() {
40 var navn = prompt("Brugernavn: ");
41 var kode = prompt("Kodeord: ");
42 if (navn=="admin" && kode=="1234") {
43 write("Du har skrevet det rigtige!");
44 setScreen(▼"login");
45 } else {
46 write("Dit brugernavn og kodeord er ikke korrekt.");
47 }
48 }
49 login();
50 function udskrivNavn (navn) {
51 setText(▼"labell", "Du er nu logget ind " + navn);
52 }
53 udskrivNavn (mitnavn);
54
```



## Rutediagram og pseudokode

- Læs kapitel 3 fra bogen “Informatik C” t.o.m. s. 37 og besvar arbejdsspørgsmål. (Ligger på modulet).

## Skriv i logbog

- I skal nu brainstorme jeres app idé.
- Bestillingsapp - hvilket produkt skal man kunne bestille hos jer.

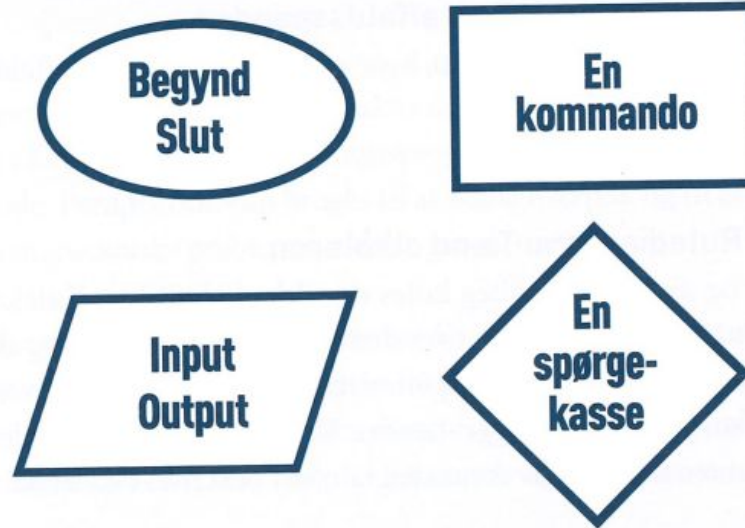


# Rutediagram

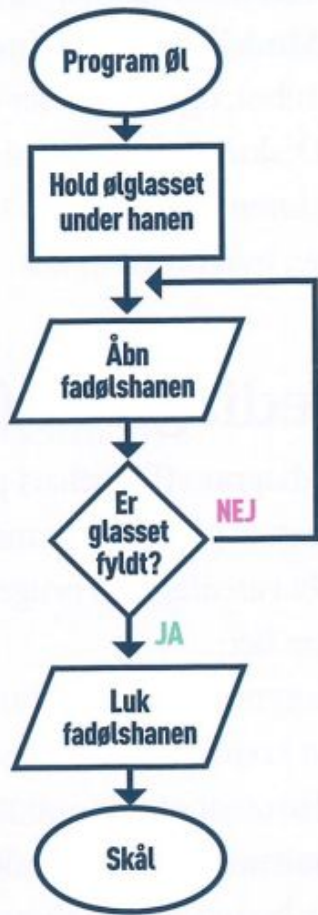




# Rutediagram



Figur 3.4: De vigtigste figurer i et rutediagram.



Figur 3.5: Rutediagram for øl-program.

## Rutediagram

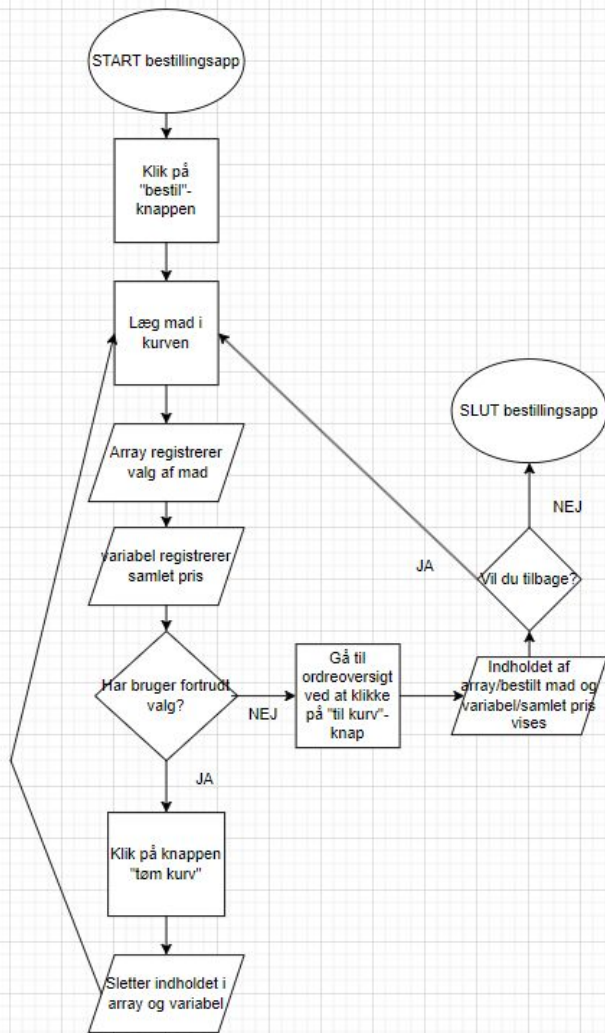
Kommando: en funktion eller programmeringshandling

Input/output: data der går ind og ud efter en kommando.

Input repræsenterer de oplysninger eller værdier, der er nødvendige for at udføre en bestemt kommando eller proces.

Eksempel: For en kommando som "Tilføj vare til kurv" kan input være oplysningerne om den valgte vare, og output kan være opdateringen af indkøbskurvens status.





Nu skal I lave eget rutediagram  
over egen app.



Speeddating!



# Prototyper

**Hvad er det:** tidlig skitse af et it-produkt.

**Formål:** Hurtigt danne sig overblik over it-produktet og hurtig måde at arbejde med idéudvikling af app. Og vise brugerne og/eller medudviklere, hvad man forestiller sig at ville bygge.

**Forskellige slags:**

**“Kvaliteten”/omfanget:**

- Low fidelity
- High fidelity

**Til hvilket formål:**

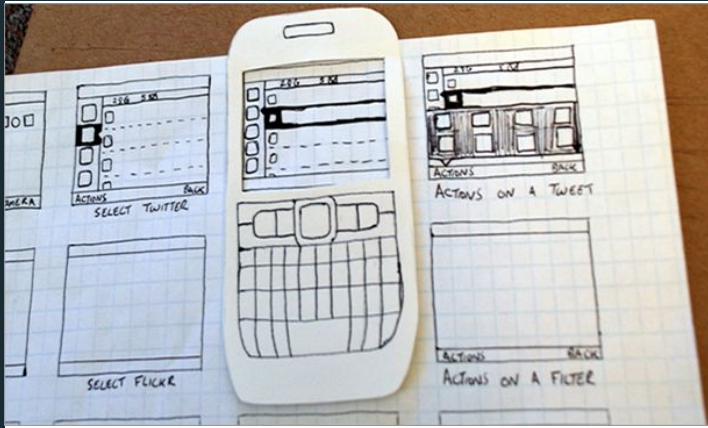
- Evolutionær prototyping
- Throw away prototyping
- Rapid prototyping



# Prototyper – Lofi vs. hifi

- **Lofi / Low fidelity prototyper:** papir, træ metal, simple, billige materialer, hurtig at lave – fremstilles tidligt i forløbet.
- **Hifi / High fidelity prototyper:** ligner det endelige produkt, mere funktionalitet, tidskrævende og kræver flere ressourcer, men kan bruges til at teste mange flere aspekter af produktet, f.eks. det følelsesmæssige aspekt – fremstilles som regel sent i forløbet.





iPhone Prototype - 1983

It lacked the portability of a mobile phone, but showed that Apple was looking to take the initiative in touchscreen technology long before the iPhone was even a glimmer in Steve Jobs' imagination.

Evolutionær,

throw away,

rapid

- **Evolutionær:**

I evolutionær prototyping bliver man ved med at udvikle prototyper, der forfølger samme spor som, fx hvordan skal brugergrænsefladen virke.

- **Throw away:**

Man laver en mockup/skitse/rapid prototype. Så prøver man den af med målgruppen, får noget feedback og bliver klogere. Derefter skaber man en ny type prototype og får mere viden. Hver gang man gennemfører en prototypesession, smider man den gamle version af prototypen væk (men man beholder naturligvis den viden, som man har fået).

- **Rapid:**

Man fremstiller hurtigt nogle mock-ups eller prototyper. Det er illustrationer af, hvordan produkter eller løsninger kan komme til at se ud. Disse skabes og vises til brugerne, så man hurtigt kan få noget feedback fra brugerne. Det kan bruges til at teste et koncept. Hvis konceptet ikke virker på papir, vil det ikke virke digitalt.

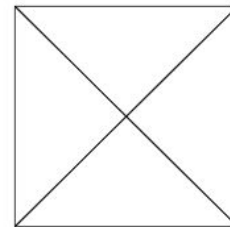
# Lofi, rapid prototyping

I skal nu skitsere jeres app-idé på whiteboards.

I skitserer alle skærme: fx

1. skærm: forside med billede
2. skærm: oversigt over varer og mulighed for at klikke dem i kurv
3. skærm: ordreoversigt/kurv
4. skærm: frivilligt om hvor mange ekstra skærme I ønsker.

Kattens kantineservice

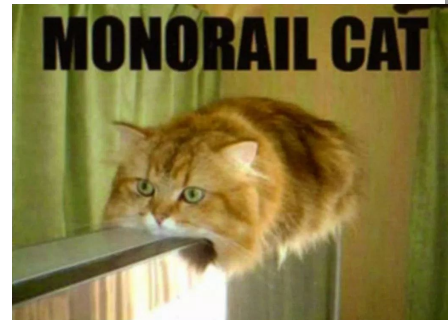


Klar til at bestille



# Hurtig intro til Tænke-højt-test

- **Tænke-højt-metoden er et stærkt værktøj til at afdække problemer på en hjemmeside eller på skærbillederne i en app. Metoden giver et indblik i, hvorledes brugerne navigerer på en side, hvordan ikoner og menupunkter opfattes.**
1. Man lader testbrugeren prøve produktet efter anvisning og siger ikke noget! Hvis testbrugeren går i stå og ikke kan finde vej, så viser det en fejl i systemet.
  2. Man laver en række opgaver. I jeres tilfælde fx:
    - A) klik på startknappen på forsiden.
    - B) Vælg mindst ét produkt, du skal putte i kurven
    - C) Gå til ordreoversigten.
    - D) Find selv på flere...





Fortsæt med min bestillingsapp i  
App Lab

<https://www.youtube.com/watch?v=GDV4f2brxFE>



# Walk and talk - 15 minutter

- Hvad er dit yndlingsprogram/-app?
- Kan du huske og beskrive dit første møde med en computer?
- Hvor meget bruger du din smartphone og til hvad?
- Tror du robotterne kommer til at overtage verdensherredømmet?





DRAMATIC PAUSE

Now what....

## Skriv i logbog

- Hvad har jeg lavet/lært i dag?
- Skriv i jeres logbog i 5 minutter - indsæt gerne skærmpoint, hvis I har.

