# IT – historie og utvikling

**Rundt 1975:**

* Skriver kode på papir («puncheskjema»).
* Leverer inn papiret til «punchedame»
* Punchedamen lager hullkortene
* Kortstokken kjøres. Kompileringsfeil? Kjørefeil?
* Som studenter i 1974 fikk vi én kjøring pr dag
* Programmeringsspråket var Algol og FORTRAN

**Skjermer (terminaler), ca 1978**

* Kommandobasert grensesnitt
* Så trege at et godt brukergrensesnitt lot brukeren velge mellom korte (en bokstav) og lange kommandoer

# Software engineering

**Eksempler på feil i programvare**

* NASAs Mars Climate Orbider
  + Værsatellitt brant opp i atmosfæren
    - En del av programmet regnet med SI-enheter (meter, cm, osv.), mens en annen del regnet med ikke-SI-enheter (fot, tommer, osv.)
* Ariane 5 (1996, 7,5 milliarder $)
  + Eksploderte etter 40 sekunder
    - 64 bits floating point konvertert til 16 bits integer.

**Programmeringens begynnelse**

* Charles Babbage (1791-1871)
  + Matematiker med interesse for å bygge regnemaskiner
  + Ønsket å lage en ikke-spesialisert regnemaskin
* Ada Byron Lovelace (1815-1852)
  + Beskrev hvordan maskinen kunne mates med ulike instruksjoner for å gjøre beregningene som var ønsket
  + Hun regnes som historiens første dataprogrammerer
  + Programmeringsspråket Ada (1975) er oppkalt etter henne

**Tidlig historie**

* I begynnelsen var det mest fokus på maskinvaren.
* John Von Neumann: «Draft Report on the EDVAC (1945).
* Herman Goldstine
  + En av de først personene som ikke oppdaget at det ikke var så enkelt.
  + Meget begrenset minne.
  + De bygde på matematisk grafteori og flow charts.

**SAGE 50 årene (U.S Air Force)**

* 1000 programmere “fra gata” fikk noen ukers kurs.
* Feilfri kode ble belønnet med $25-$75 pr instruksjon.
* Etter hvert fant de ut at programvarebiblioteker var lurt, hvis det var noe man gjorde ofte, så kunne dette lagres i biblioteker og enkelt hentes når det trengs.

**De fem fasene**

1. Spesifisere systemet
   1. Finne ut hva programmet skal gjøre.
2. Designe systemet
3. Programmere systemet
   1. Koden skal være billig å utvikle, effektiv å kjøre og lett å forstå.
4. Teste systemet
5. Vedlikeholde systemet
   1. Vedlikeholdskostnadene overstiger ofte utviklingskostnadene.

**Programvarekrise**

1965-1968

* Feil produkt leveres for sent til langt høyere kostnad enn planlagt
* Eksempler:
* 1995: Statens Vegvesen taper 152 mill kr ved utvikling av et nytt system son viser seg å være ubrukelig.
* 1995: NSB taper 200 mill kr på et nytt billettsalgsystem som ikke fungerer
* 2010: Nytt billettsystem i Bergen skulle koste 90 millioner, prisen ble 187 millioner.

1995:

* Halvparten av prosjektene gav løsninger som fungerte, men som likevel ikke kunne betraktes som vellykkede.
* Gjennomsnittsprosjektet overskred budsjett med 50%

**Objektorientert programmering – begynnelsen**

* Simula 67
  + Utviklet av: Kristen Nygård (1926-2002) og Ole-Johan Dahl (1931-2002)
  + De eneste nordmennene som har fått «Nobel-prisen i IT», Turing-prisen (2001)
  + Kristen Nygård var også veldig imot EU, så han var aktiv på flere fronter.

**1990-1999: Internettet kommer!**

* En helt ny måte å jobbe på for programutviklere (og alle andre).
* Nye krav til programvareproduktene
  + Internasjonalisering
  + Sikkerhet
  + Antall potensielle brukere øker formidabelt
* Nye produkter
  + Distribuerte systemer, nettverksprogramvare
  + Søkemotorer
  + Maskinoversettelse av naturlige språk
  + Web-applikasjoner med alle typer data (bilde, lyd, video mm.)

**Programvareutvikler: Ingeniør eller matematiker?**

* Ingeniør
  + kommer av latin *ingenium* som betyr «begavelse, oppfinningsevne»
* Mye er lån av andre tradisjonelle ingeniørdisipliner: utbyttbare deler, standarder, konstruksjon av kompliserte systemer, kvalitetskontroll.
* Det som er spesielt: Programvare er ikke fysisk, bygger ikke på naturvitenskap (fysikk, kjemi og biologi), vedlikehold er ikke det samme i programvare som til en bro.

Internett – «Nett mellom nett»

* ARPA (Advanced Research Projects Agency)
  + «Visjonen om et "intergalactic network" – et verdensomspennende desentralisert nettverk av datamaskiner hvor alle raskt kunne aksessere data og programmer fra er hvilket som helst sted – ble skapt av J.C.R. Licklider, leder av Information Processing Techniques Office ved ARPA. (1962)»
* 1973: TCP/IP funnet opp, 32 bit adresser
* 1983: TCP/IP protokoll for ARPANET
* 1984: DNS Tjeneste som oversetter domenenavn til IP-adresser.
* 1992: Alle norske høgskoler og universitet er nå koplet til Internett (Uninett)
* 1994: De første nettbutikkene og den første nettbanken kommer.
* 1996: Første mobiltelefon med Internett: Nokia 9000 Communicator
* Slutten av 90-tallet: Noe helt nytt: Umiddelbar kontakt med hele verden 🡪 Dot-com-boblen
* (Programmererne fikk i stedet holde på med 2000-årsproblemet.)

**WWW (World Wide Web) – en tjeneste på Internett**

* 1991: Web kommer!
  + Definerte HTML (HyperText Markup Language), HTTP (HyperText Transfer Protocol) og URL (Universal Resource Locator) - 1990
  + HTML definerte strukturen (kapitler, underkapitler og avsnitt). Nettleseren bestemte presentasjonsformen.
  + Tim Berners-Lee, CERN, satt sammen www og skapte mediumet for det 21.århundre.
* 1993: Nettleseren Mosaic
  + WWW-trafikken øker med 341 000% på ett år.
* 1994: Første søkemotor: Yahoo! (Yet Another Hierarchical Officious Oracle)
* 1995: Sun lanserer Java
  + helt maskinuavhengig
  + programmer («applets») kan lastes ned fra Internett og kjøres på en hvilken som helst maskin uansett operativsystem – kjøres fra en nettleser
  + (Java er også et generelt programmeringsspråk)
* 1995: Microsoft lanserer Internet Explorer
  + Leveres som en del av Windows.
  + (Nesten) alle PCer får lett tilgang til Web
* Web 2.0
  + Samarbeid på nettet
  + **You** **– Time Person of the Year 2006**