

readme

Studentersamfundets
sanne historie

Skaff deg enda flere
bekymringer

Gjerningsmennene bak
dine kalkulus-traumer

Er du på
slemmelisten?

REDAKTØR

Viljan van Raaij

LAYOUTANSVARLIG

Iver Ringheim

MEDVIRKENDE

Elena Kristine Blakstvedt Willmann
Fredrik William Husemoen-Zhang
Eli Fjellbirkeland Johannesen
Henrik Johan Kristiansen
Madelen Hellervik Lothe
Martine Meng Karlsen
Ian McGregor Aksum
Emil Moberg Værnes
Vetle Roos Mangrud
Magnus Hanesand
Lea Aas-Jakobsen
Martine Mansåker
Johanne Norland
Jesper Steinsrud
Øyvind Monsen
Elisabeth Doan
Håkon Sæther
Petter Buseth
Vegard Ervik
Maja Vullum
Juni Bugge

FORSIDE

Ian McGregor Aksum
Viljan van Raaij
Iver Ringheim

KONTAKT

readme@abakus.no

readme, Abakus
Sem Sælands vei 7-9
7491 Trondheim

NETTUTGAVE
readme.abakus.no

netcompany

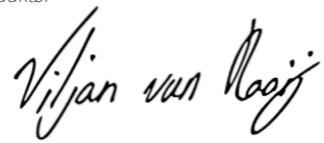
Leder

Nok et år går mot slutten og tiden er kommet for å sette inn siste støtet før alle kan ta seg en velfortjent ferie. Jeg vil anta at det er mange nerver hos dere førsteklassinger, i det som muligens blir deres første eksamensperiode som student. I hvert fall hadde jeg det slik. Frykt derimot ikke, for alle kommer seg gjennom eksamener og studiet handler ikke alltid om å få best karakterer. I eksamensperioden er det spesielt viktig å få noen avbrekk underveis i all jobbingen. Ta noen pauser på kontis med Mario Kart, les denne utgaven av **readme** fra perm til perm eller ta deg noen dager helt fri. Én ting til når jeg først er inne på eksamensperioden: Dere som er ferdig med eksamen 12. desember, vær så snill og la det være igjen litt mat og snacks i SnackOverflow. Det sitter en sulten og tørst gjeng på A3, som ikke kan ta ferie før en god stund senere.

Uansett, ettersom dette blir siste utgave for i år, skulle det bare mangle å se tilbake på hva som har skjedd i 2023. Først og fremst ønsker jeg å gi ekstra ros til et nytt initiativ i Abakus, nemlig 1. klasse task force. Med tanke på all innsatsen de har lagt inn, så er det vel fortjent at de får sin egen midtside i denne utgaven. Ellers skal vi, som tidligere år, dele ut **readme**-prisen til noen av de mest merverdige hendelsene og personene i 2023. I tillegg er det som alltid masse annet spennende å lese, så krøll deg inn under et varmt teppe og nyt resten av utgaven.

Med det så ønsker jeg alle masse lykke til på eksamener og god jul!

Redaktør



I denne utgaven

- 6 Du gjetter aldri hvem som har vært på Samf!
- 12 Utslettelser som kan skje i morgen
- 22 Bakgrunnen til høstsemesterets terror



Hvor er det blitt av alle jobbene?

Tekst: Vegard Ervik
Foto: Iver Ringheim

Denne teksten gir uttrykk for skribentens egne meninger

Nettavisen kode24 skriver at flere mennesker med tekniske utdanninger sliter med å skaffe seg fast jobb. Blant annet kan man lese om Vegard Kjørberg som har tatt en mastergrad i datasikkerhet fra UiB og ikke fått seg en relevant jobb, selv etter å ha søkt 150 jobber. Man kan også lese om Vegard Paulsen som har en bachelor i Frontend- og mobilutvikling fra Høyskolen i Kristiania som heller ikke har fått seg fast jobb. Er det bare et dårlig tegn å hete Vegard [...], eller blir det nå vanskeligere for oss med teknisk kompetanse å få oss jobb?

ULV, ULV?

I 2021 ble det gjennomført en rapport fra Samfunnsøkonomisk analyse som viste at Norge trenger 4500 studieplasser innen IT, 1500 plasser flere enn det som finnes i dag. I flere år har også ledere, rekrutterere og bemanningsbyråer vært høylutte i media om at det er for få

utdanningsplasser innen IT. Derfor er det et lite paradoks i at de samme bedriftene som skrek etter arbeidskraft ikke ansetter nyutdannede.

For å nøste opp i paradokset, er det flere sider av saken man må se på. Den mest nærliggende grunnen til at bedriftene ikke ansetter like mange som før er at det er nedgangstider. Markedet er ikke det samme som i 2021, og dermed trenger ikke bedriftene lenger 4500 studieplasser. Rart hvordan man klager på utdanningsinstitusjonene når det utdannes for få, men er tyst når det utdannes for mange.

En annen grunn til at bedrifter skriker etter kompetanse, men ikke ansetter nyutdannede, er det faktum at bedriftene ikke er helt transparente når de går til media. Ja, det er mange bedrifter som trenger mye arbeidskraft innen IT, men etterspørselen er seniorer, ikke nyutdannede. Så

selv om man kan føle seg veldig ønsket gjennom lovord på bedpresser og medieoppslag, vil de fleste bedriftene helst ha deg et par år etter at du er ferdig utdannet.

HVA MED OSS I ABAKUS?

Enn så lenge ser det ut til at det går bra med oss abakuler. Utmatrikuleringsundersøkelsen 2023 viser at 94 prosent av studentene i Abakus hadde signert jobb ved utgangen av høsten 2024, og at nesten uten unntak har alle studentene jobbt tilbud før studiene er fullførte. Dette er et veldig godt tegn, og viser at Datateknologi og Cybersikkerhet og datakommunikasjon er svært attraktive studier. Allikevel er det viktig å presisere at det på ingen måte er vanlig å få sommerjobber slengt etter seg, eller at det er lett for alle å sikre seg fast stilling.

readme

-prisen 2023

Tekst: Maja Vullum og Magnus Hanesand Layout: Madelen Hellervik Lothe



ÅRETS ILDSJEL

👑 Svenske koraner
Kragerø
Hellas



ÅRETS COMEBACK

👑 Gonoré
Tina og Bettina
👑 Israel-Palestina konflikten



ÅRETS HØYDEPUNKT

Kristin Harila
👑 Titanic-ubåten som imploderte
Da Kiwi ikke satt opp prisene i februar



ÅRETS NAVN

X/Twitter
CybDat
👑 Hans



ÅRETS PRESTASJON

Hun fra Fredrikstad på GenVors
Elon Musk med Twitter
👑 Ordstyreren som speeda gjennom statuttendingene på genfors



ÅRETS LAGPRESTASJON

Månelandingen til India
👑 Datakameratene
Rumpa bar



ÅRETS FORBILDE

Sophie Elise
👑 Bjørnar Moxnes
Danby Choi



ÅRETS ØYEBLIKK

AbaVan
Øyvind som blir shippa med Tuva på julebord, selv om han er forlovet
👑 Finland blir med i NATO



ÅRETS DUO

Bamsegutt og Tore Strømøy
👑 Erna Solberg og Sindre Finnes
Barbie og Oppenheimer



ÅRETS GJENNOMBRUDD

👑 Jordskjelvet i Tyrkia
Follobanen
Helseplattformen

STUDENTER SAMFUNDET

– bygget som gjør deg til en student

Kanskje liker du å slappe av med en kopp kaffe på Edgar, dra på konsert i Storsalen, eller drikke deg møkkings på (relativt) billig øl. Studentersamfundet er det naturlige samlingsstedet for alle studenter i Trondheim uansett hva man er ute etter.

Det store, runde, røde bygget har stått oppreist på Elgeseter gate helt siden 1929, og minimale endringer har blitt gjort siden det var oldemoren din som herja på dansegulvet i Bodegaen. Allerede i 1963 ble de første forslagene for utvidelse av Samfundet presentert i Storsalen, og endelig skal vi få det etterlengtede nybygget. Nå er det ikke lenge til det skal stå ferdig, så her kommer litt Samfundet-lore som er greit å kunne som student i Trondheim.

Studentersamfundet i Trondheim blir stiftet like etter immatrikuleringen av NTHs første studenter. Rådssalen i Hovedbygget blir imidlertid studentenes møtested, men det tar ikke lang tid før man skaffer pengene til å kjøpe et eget tilholdssted.

På grunn av Samfundets trange økonomi blir det bestemt at man skal sette opp en revy. Den første UKErevyen, Baccarat, fremføres i en uke – derav navnet UKA, som faktisk gir mening i motsetning til hvor lang UKA er i dag. Revyen bringer inn omkring 10 100 kroner, som i dagens valuta er over 400 000 kroner. På grunn av den overveldende suksessen bestemmer man seg for å sette opp en ny revy to år senere.

Etter mange suksessfulle UKE-revyer har Samfundet endelig nok penger til å starte byggingen av et nytt studenthus. Det blir holdt avstemning, og med 309 stemmer mot 1 stemme, beslutter Samfundet å bygge. I den hensikt blir det holdt arkitektkonkurranse med følgende kriterie: «Den gamle sals grunnidé [skal] beholdes i den nye, [...] og såvidt mulig nærme seg den sirkulære grunnform.»

1. oktober 1910

1912

1917

Cirkus i 1922

2. november 1927

Den tidligere sirkusbygningen Cirkus i Elgeseter gate blir kjøpt opp og erstatter Rådssalen som Studentersamfundets nye samlingssted.



Tekst: Håkon Sæther Layout: Juni Bugge

1. oktober 1929



Samfundet på 1930-tallet med Volla kretsfengsel rett bak

1930-tallet

8. mai 1945



Samfundet anno 1946-1950 (fun fact: Samfundet var gult fra 1939 til 1962)

1946-1950

Nøyaktig 19 år etter Studentersamfundet ble stiftet, står endelig bygget vi i dag kjenner som Samfundet ferdig. Samme dag kan studentene i Trondheim innta deres nye hjem som den gang ble kalt «Cassa Rossa».

Allerede samme dag som tyskerne kapitulerte i Norge, gjeninntar trondheimstudentene det nå ruinerte huset sitt. I løpet av den tyske okkupasjonen av Norge ble nemlig ikke studentenes hjem behandlet særlig pent. Gulv og trapper er hakket opp, og Storsalen, Samfundets hjerte og sjel, er dekket av hestemøkk som følge av gjentatte sirkusshow med ponnier under krigen. Likevel, etter bare ni dager med restaurering og intens skuffing av hestemøkk, står Samfundet klart – dog med en noe snedig ødør – til å ta imot 14 000 mennesker i Storsalen 17. mai.

Nybygget

Helt til det ble revet i mai 1972, var Volla kretsfengsel nærmest vegg i vegg med Samfundet. Siden den gang har det vært gjentatte diskusjoner om å utvide Samfundet, men det skulle vise seg å ta lang tid før forslaget fikk gjennomslaget man hadde håpet på. I 2016 gjenoppsto håpet om utbyggingen

Ganske nøyaktig 50 år etter rivingen av fengselet, ble nybyggets grunnstein lagt ned av Hans Kongelige Høyhet Kronprins Haakon i april 2022.

Når blir åpning? – Selve bygget skal stå ferdig i februar 2024, men på grunn av innredningen vil dørene først åpne i mars 2024.





Gløsløken

Største årsak til foreldreløse barn

NYTT NAVNEBYTTE FOR KOMTEK PLANLAGT I 2024

Tekst: Vegard ErvIK

GLØSHAUGEN – Cybersikkerhet og data-kommunikasjon (tidligere Kommunikasjonsteknologi og digital sikkerhet (tidligere Kommunikasjonsteknologi)) kommer til å bytte navn i 2024, melder flere sikre kilder til NTB. «Det blir ofte sagt at for å være best, må man kopiere de som er best», sier en anonym leder i IIK til **Gløsløken**. «Vi ønsker derfor å kopiere Marvel, og gi Komtekfans nytt navn å glede seg til hvert år.» **Gløsløken** har klart å finne ut navnene helt frem til 2026, og de lyder som følger:

2024: «Cyber security in the cloud»

2025: «Big AI cloud cyber security and ChatGPT Transformative Hyperloop»

2026: «TCP/IP»



UNGE VOKSNE VIL AVSKAFFE JULENISSEN

Tekst: Ringer IverHjem?

VERDEN – En gruppe unge voksne har startet en motstandsbevegelse for å få en stopp på julenissens gjerninger. De kaller seg selv for Santa Saboteurs (SS), og har det siste året jobbet intenst mot julenissens propaganda. Senest natt til 10. november i år gikk SS rundt og gjorde hærværk på alle butikker som hadde symboler knyttet til julenissen. **Gløsløken** har intervjuet Lil Nas Ist, en representant for SS. Da han ble spurt om hvorfor de opprettet SS svarte han: «Mamma sa jeg ikke kunne ønske meg en *blowjob* til jul, for julenissen gjør ikke slikt.» «Vi synes dette er dårlig arbeidsånd, og skal jobbe for å skaffe en ny julenisse som klarer å gi *blowjobs* til jul.»



PROGRAMVAREREGRESJON

Tekst: Beatle Maroon 5

SVERIGE – Fenrik Skliberg må tilbake til Spotify etter at det kommer frem at de har misforstått agil utvikling helt. Det viser seg at teamet som er ansvarlig for Spotify Sessions har lest feil. Det som egentlig skulle være «Backlog refinement» har blitt «Backlog randomization». Det ble klart etter at Spotify fjernet Spotify Sessions-etterkommeren Spotify Jam, og byttet den ut med «Spotify Together». Nå fungerer det slik at en person koblet til en høyttaler kan starte en «Together». For å bli med må de andre tappe mobilen sin på mobilen til verten. Når du gjør det vil du få en av fire roller. Den første rollen er «DJ-en». Disse personene kan legge til sanger i økten. Den andre rollen er «Musikkengelen». Disse kan høre musikken på sin egen mobil. Den tredje rollen er «Sabotøren». Disse vil ved tilfeldige mellomrom skippe sanger, sette sanger på pause eller spille sanger fra en tilfeldig annen spilleliste. Den siste rollen er «Tronraneren». Disse vil sparke ut verten og selv bli vert. Du får et valg om å ønske en rolle, men hvilken rolle du faktisk får er tilfeldig.



Utgavens master

Å løse bølgeligningen på stormaskiner

Performance Modeling of a Load-Balanced FDM Wave Equation Solver on Heterogeneous Clusters



Tekst: Tobias Meyer Andersen (M.Sc. 2023), Sivilingeniør i SINTEF

Numerikk lar oss bruke datamaskiners regnekraft til å tilnærme svaret på matematiske spørsmål. Vitenskapelige beregninger er helt avhengig av numerikk ettersom mange problemer er upraktiske, eller umulige å løse med eksakte metoder. Skal disse tunge regneproblemene løses raskt, kreves effektiv kode og hardware som tilbyr de ressursene programmet krever. Masteren min handlet om å implementere en *endelig differansemetode (FDM)* for å numerisk løse bølgeligningen på stormaskiner, i tillegg til å undersøke hvilke maskinegenskaper som avgjør kjøretiden.

PARALLELLISERING AV BØLGELIGNINGEN

Bølgeligningen kan beskrive en bølges bevegelse inni for eksempel en boks. FDM deler boksen inn i mange punkter hvor vi regner ut bølgens høyde hvert tidssteg i simuleringen. Punktene i samme tidssteg avhenger kun av resultater fra tidligere tidssteg, og kan derfor regnes ut samtidig. Du kan derfor implementere en algoritme som bruker alle kjernene på prosessoren din samtidig til å regne ut svaret i ulike punkter, eller får bruk for den parallelle

regnekraften i grafikkortet ditt. Det er heller ingenting i veien for å fordele punktene på flere datamaskiner som samarbeider! For at svaret skal bli rett må datamaskinene regelmessig utveksle sine lokale delresultater. Du kan bruke mer tid på beregninger hvis maskinene strategisk kommuniserer visse verdier samtidig som de regner ut andre. For store problemstørrelser vil dette trikset kunne maskere hele kommunikasjonskostnaden!

Jeg fikk bruke NTNUs stormaskin «Idun» til å kjøre beregninger på en håndfull grafikkort samtidig fordelt på opp til ti datamaskiner. Ettersom denne stormaskinen er veldig heterogen, som vil si at den består av mye ulike hardware, lagde jeg også et lastbalanseringssystem som ga hver maskin en arbeidsmengde som passet med hvor mye den rekker å regne ut.

Å FORSTÅ KJØRETIDEN

Når industrien bruker en liten formue på en stormaskin som skal løse deres beregningsproblemer, vil de vite at de kjøper

det de trenger. Utover kompleksitetsanalysen i Algoritmer og datastrukturer lærer vi lite om å modellere kjøretiden til programmer. I praksis kan man ofte måle noen enkle egenskaper ved en datamaskin, og tilnærme forventet kjøretid for et program. Dette kalles *ytelsesmodellering*, og en god modell lar deg studere to motstående problemer. Modellen kan beskrive hvilken maskin som raskest kjører programmet, og hvordan programmet skal kjøre raskest på din maskin. Ytelsesmodellering lar deg altså tilpasse koden til din hardware, og velge hardware til koden.

RESULTATER OG TANKER

Programmet regnet ut bølgens høyde i 600 milliarder punkter per sekund i snitt gjennom simuleringen. Kjøretiden ble modellert nøyaktig hvis man kjente hvor fort maskinene kunne snakke med hverandre og lese fra minnet. Oppgaven var en strålende introduksjon til vitenskapelige beregninger både fra et programvare- og maskinvareperspektiv. Tusen takk til min utrolig hjelpsomme veileder, Jan Christian Meyer.

Guide til keege turklær-merker

Tekst: Henrik Johan Kristiansen
Layout: Petter Buseth

Å pryde seg med dyre turklær er en gjenganger blant studentene på Gløshaugen, og vi abakuler er intet unntak. Klærne vektlegger funksjon over utseende, noe som ikke er så rart gitt at de er laget for å brukes ute på tur. Men aktivt bruk ute i naturen er nødvendigvis ikke tilfellet for gløshauginger, som primært bruker klærne til og fra campus. Da følger det at utseende trumfer funksjonalitet, og de fleste er nok enig i at merket er det aller viktigste. Uansett om du velger å gå med skalljakke, dunjakke, vest, bukse, eller t-skjorte må du ha et bevisst forhold til hvilket merke plagget bærer. Merket er en visualisering av personlighet, stil, interesser og ikke minst pengebok. I den anledning presenterer vi i **readme**: keege turklær-merker og hva de sier om deg.



Kanskje det mest kjente luksus-turmerket som finnes. Stiftet av klatrer og filantrop Yvon Chouinard i 1973, er Patagonia et selskap som verdsetter bærekraft og miljø. Ansatte kan få lønnede fridager gitt at de bruker tiden på å fremme en miljøsak, og 91 prosent av ansatte sier det er et fint sted å jobbe. For mange har merket blitt en slags kult, der det er enten Patagonia eller ingenting. Går du med Patagonia er du enten en bløffmaker som aldri har sovet ute en vinternatt, eller så er du en reell naturperson med tykk pengebok. PS: Nye undersøkelser viser at folk i kritthvit dresskjorte og Patagonia-vest kun har sex i misjonærstillingen!



Norrøna ble stiftet i 1929 av Jørgen Jørgensen og er Norges største friluftslivsselskap. Den tidlige visjonen til selskapet var å lage friluftstutyr som tålte røffe, norske forhold. Visjonen står fortsatt i dag, men bærekraft har også blitt en stor del av selskapets verdier. De er ett av få selskaper som oppgir en full liste over fabrikkene sine og hvor bærekraftige de er. Norrøna har relativt stive priser, men man får virkelig det man betaler for. Klærne har høy kvalitet og merket er høyt respektert både i og utenfor Norge. Går du i Norrøna er du patriotisk.

NORRØNA



ARC'TERYX
THE
NORTH
FACE

Arc'teryx og The North Face er to merker som har valgt å gå i en annen retning enn andre konvensjonelle turmerker. De har nemlig valgt å satse deler av virksomheten på *gorpcore*, turklær som er ment å brukes av urbane mennesker i urbane miljø. Det respekterer naturligvis ikke vi i **readme**, så går du i klærne deres er det mange fallgruver du må være klar over. For eksempel, Arc'teryx sin groveste skalljakke, Alpha Jacket SV, vil gjøre deg gode ti tusen kroner fattigere. For å fortjene den respekten som jakken krever, må du gå minst 30 toppturer i året, hvorav 10 er i sidelengs regnvær. Gjør du ikke det eller tilsvarende, er du en byrotte med forkastede moraler, og du er i samme kategori som de med med Patagonia-vest over hvit dresskjorte, misjonær-kategorien.



Bergans ble stiftet i 1908 og var med på å revolusjonere ryggsekker ved å innføre stålrør i ryggkonstruksjonen. Bergans var et respektabelt merke, og deres sekker ble brukt av både polfarer Roald Amundsen og av britiske fjellklatrere under de første ekspedisjonene til Mount Everest. I dag er de nedgradert til et helt middelmådig turmerke med svak merkevare. Går du med Bergans har du ingen forhold til hvorvidt det er et kult merke eller ikke. Det betyr at du enten er en fet turperson som verdsetter funksjon over utseende, eller at du fortsatt handler klær med moren din.

Bergans
OF NORWAY

Lowe
alpine

Lowe Alpine ble stiftet i 1972 av tre brødre med etternavnet ... Du gjettest riktig: Lowe. De lagde alt fra sekker til sko, men etter å ha blitt kjøpt opp av RAB i 2011, ble det i 2014 bestemt at de kun skulle lage sekker. Det var en stor skuffelse for mange turmennesker, da de lagde noen av de beste plaggene på markedet. De største tilhengerne har tatt godt vare på Lowe Alpine-plaggene sine og det er derfor mange plagg som fortsatt er i omløp. Går du med Lowe Alpine har du foreldre som er aktive turmennesker, og du ønsker heller arve deres gamle klær enn å kjøpe nye. Det gjør at absolutt alle med Lowe Alpine-plagg er skikkelig fete folk. Lowe Alpine slår aldri feil, og det er **readme** sin endelige anbefaling om du leter etter et nytt keeegt turplagg.



ALTUTSLETTENDE KATASTROFER

SOM KAN INNTREFFE NÅR SOM HELST

Tekst: Eli Fjellbirkeland Johannesen
Layout: Johanne Dahl Norland

Det er mye man kan bekymre seg for i dagens samfunn: klimaendringer, krig og eksamensperiode. Om du ser mørkt på fremtiden, kan du trøste deg med at det finnes en liten sannsynlighet for at verden slik vi kjenner den vil forsvinne i morgen. Bare senk skuldrene og slapp av, readme presenterer to katastrofer som kan utslette menneskeheten på et blunk.

ASTEROIDER

Dersom en stor asteroide treffer jorden, vil dette få katastrofale globale konsekvenser. Først og fremst vil asteroiden brenne opp alt som ligger i sin vei. Når asteroiden kommer inn i atmosfæren har den så stor fart at luften rundt den ikke rekker å flytte på seg, men heller blir komprimert. Denne komprimeringen øker temperaturen under asteroiden noe voldsomt – den kan bli mer enn ti ganger overflatetemperaturen på solen! Varmeøkningen får selvfølgelig katastrofale konsekvenser for alt som befinner seg i nærheten av asteroiden, og når den treffer jordoverflaten vil temperaturen være så høy at asteroiden vil eksplodere og «forsvinne».

Sammenstøtet vil føre til en kjede-reaksjon av andre naturkatastrofer, som vulkanutbrudd, jordskjelv og tsunamier over hele kloden. Det vil også skapes en stor askesky som vil blokkere alt lys fra solen, og det vil regne glødende stein og andre vrakrester fra sammenstøtet overalt. Askeskyen vil også gjøre at den globale temperaturen etter hvert synker, noe som vil ha stor innvirkning på alt av planter og dyr som overlevde tiden rett etter sammenstøtet.

Det er ikke sikkert at vi kan vite på forhånd når en stor asteroide er på kollisjonskurs med

jorden. NASA anslår at dersom en asteroide med diameter på to-tre kilometer treffer jorden, vil dette få alvorlige følger for hele kloden. Det meste vi observerer i verdensrommet er ganske mye større enn dette, og det hjelper heller ikke at asteroider reflekterer svært lite lys, noe som gjør dem enda vanskeligere å få øye på fra jorden. Om vi allikevel skulle oppdage at det var en stor asteroide på vei mot oss, er det ikke så veldig mye vi kan gjøre. For det første er det vanskelig å beregne banen til asteroiden, og man kan derfor ikke være helt sikker på om den kommer til å treffe jorden før den eventuelt gjør det. Vi har heller ikke teknologi som er god nok til å ødelegge eller flytte på asteroiden. Selv med atombombene vi har i dag ville vi vært sjanseløse.

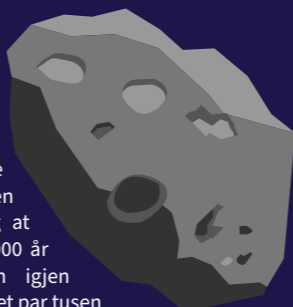
SUPERVULKANER

Supervulkaner er en type vulkaner som har mye større og mer katastrofale utbrudd enn det vanlige vulkaner har. Forrige supervulkanutbrudd skjedde for 74 000 år siden i Toba, Indonesia. Utbruddet fikk store

konsekvenser for livet på jorden. Mange mener at mennesket ble brakt til randen av utryddelse og at det tok over 20 000 år før populasjonen igjen klarte å overstige et par tusen individer.

I dag finnes rundt 30 aktive supervulkaner rundt om i verden, men den mest kjente er nok Yellowstone i USA som dekker et område på omtrent 9000 kvadratkilometer. Det er nesten tre ganger så stort som Hardangervidda. I gjennomsnitt har Yellowstone et utbrudd hvert 600 000. år. Nå er det 640 000 år siden forrige utbrudd, og mange mener at Yellowstone snart vil eksplodere på ny. Problemet med Yellowstone, og supervulkaner generelt, er at man egentlig ikke vet så veldig mye om dem. Det er ingen som vet hvilke tegn som kan indikere at et stort utbrudd er på vei. For eksempel begynte grunnen i et område i Yellowstone å bule utover, noe som førte til at grunnen i et større område ble hevet med over en meter på knappe 50 år. Deretter, i 1985, sank plutselig deler av det samme området med 20 centimeter, og nå ser det ut som at grunnen hever seg igjen. Hva dette indikerer er det ingen som vet. Det er heller ikke usannsynlig at det ikke vil komme noen forvarslar på et utbrudd, og at Yellowstone vil eksplodere tilsynelatende ut av intet.

Om Yellowstone skulle eksplodere, vil dette gå hardt utover livet på jorden. Lava og aske vil dekke store deler av Nord-Amerika, og det vil dannes en askesky som vil påvirke klimaet på hele kloden. Med kaldere klima vil det ikke ta lang tid før avlinger og matproduksjon over hele verden slår feil, og man vil stå overfor en global sultkatastrofe.



Frutiger Aero

– Fremtiden vi ble lovet, men aldri fikk

Tekst: Elena Willmann
Layout: Martine Mansdøker

Har du noen ganger tenkt over hvor annerledes det var å bruke en datamaskin før? Fargene var sterkere, kantene var rundere, ikoner var realistiske og det var bobler overalt. Dette er ikke bare en vag idé om din barndom, det er en egen designstil med navnet «Frutiger Aero».

Frutiger Aero er en designstil og estetikk som ble mye brukt i markedsføring, media og teknologi fra 2004 til 2013. Stilen kjennetegnes ved sin bruk av skeuomorfisme (designkonsept for å få gjenstander til å ligne deres virkelige motparter), blanke teksturer, skyet himmel, vann, bobler, buede linjer, frutiger fonter, nordlys og farger som grønn og blå. Bruken av blanke teksturer har også gjort at stilen blir kalt for «Web 2.0 gloss». Frutiger Aero er oppkalt etter Adrian Frutiger, som skapte Frutiger-fontene, og Windows Aero, som var stilen brukt til Windows Vista.

HISTORIEN TIL FRUTIGER AERO

Frutiger Aero har sine røtter fra 2001 til 2004, og bygger på Microsoft Windows Longhorn og det skeuomorfistiske designet til Mac OS X. Her ble Windows Aero-glasskanter introdusert og blanke teksturer tok sakte over. Verdenen hadde blitt preget av Dot com-boblen som sprakk og krigen mot terror, og det skjedde et kulturelt skifte som gjorde at Frutiger Aero begynte å ta over den da prominente «Y2K»-stilen. Fra 2004 til 2007 begynte virkelig Frutiger Aero å bli populært, med utgivelsene av Windows Media Player 10, Nintendo Wii og Apple iPod sin silhuett-kampanje. Men det var da Windows Vista og den første iPhone kom på markedet at stilen ble helt etablert og

storhetstiden begynte. Mellom 2007 og 2012 var nesten all media, markedsføring og teknologi påvirket av Frutiger Aero.

Dessverre kan ikke alle gode ting vare for alltid, og fra 2012 begynte nedgangen. Skeuomorfismen ble erstattet av designstilen «Flat Design», som karakteriseres av minimalisme, visuell organisering, renslighet og fargene hvit, lys grå og lys blå. Nintendo Wii U var den siste spillkonsollen som benyttet seg av Frutiger Aero-designet, som dessverre var en kommersiell fiasko. Dette markerte døden for Frutiger Aero.

VEIEN VIDERE

Selv om stilen er død, lever den fremdeles i våre minner. Dette er ikke bare i teknologi og UI-design, men i filmer, tv-serier og spill. Har du kanskje glemt hvordan Teletubbi-verdenen, Lazytown og Perry sitt reir i Phineas og Ferb ser ut? Eller hvordan hele Avatar-filmen er svært blå og grønn? Selv romskipet fra Wall-E har elementer fra Frutiger Aero. Det er kanskje i spill der stilen er mest representert, med eksempler som Wii-serien, Super Mario Galaxy, Spore og Sims 3.

Synes du denne stilen virker interessant så er du heldig, fordi den gjør nemlig et lite nostalgisk «comeback». De som brukte maskiner med Frutiger Aero-stilen i sin oppvekst er nå i 20-årene. Stresstet av studier og jobb gjør kanskje at man ønsker seg tilbake til gladere dager med klar blåfarge, glasseffekter og avrundede kanter. Y2K-stilen har jo allerede kommet tilbake, så det er bare et spørsmål om tid før Frutiger Aero gjør sin velfortjente entré på markedet.



Er det noen som
fårstiser her?



Foto: Ian McGregor Aksum og Johanne Dahl Norland

#119

TING DU BØR VITE SOM SIVING

SSH

Tekst: Øyvind Monsen

Et begrep du sannsynligvis har vært borti som data- eller komtek-student er SSH, som står for *Secure Shell*. Den enkleste måten å tenke på SSH er at det lar deg logge inn på en annen maskin sin terminal over internett. SSH kan blant annet brukes til å logge inn på serveren som *hoster* nettsiden din eller fikse noe i databasen. Det du kanskje ikke vet er at SSH faktisk kan gjøre mye mer enn bare det.

I likhet med HTTP er SSH en protokoll som lar ulike programmer snakke sammen. SSH-protokollen er bygget for å tillate sikker kommunikasjon, selv over usikre nettverk. Dette gjøres ved at de to partene først utfører et kryptografisk håndtrykk, i likhet med SSL som du kanskje har hørt om tidligere på studiet. Når tilkoblingen er satt opp, kan partene sende informasjon sikkert fram og tilbake. Altså kan du være sikker på at når du kjører

```
sudo rm -rf
```

på prod-serveren, gjør du det over en sikker protokoll.

Siden SSH tillater sikker kommunikasjon, har det også blitt vanlig å bruke protokollen for å autentisere i andre settinger. Når man

autentiserer med SSH kan man bruke *ssh-nøkler* i stedet for brukernavn og passord. Dette er et sett med offentlig- og privat-nøkkel som gjør at serveren kjenner deg igjen og gir deg tilgang. For at serveren skal vite hvem du er må du legge inn din offentlige nøkkel på serveren i en mappe som heter `~/.ssh/authorized_keys`.

Nå som du kan bruke SSH uten passord er det kanskje litt kjedelig å hver gang måtte skrive SSH kommandoen fullt ut med bruker og domene/IP-addressse. Heldigvis er det mulig å konfigurere SSH til å huske forskjellige *hosts*. I filen `~/.ssh/config` kan du for eksempel legge inn:

```
Host prod
  User morradi
  Hostname folk.ntnu.no
```

Deretter kan du nå bruke kommandoen

```
ssh prod
```

og du vil bli logget inn på `folk.ntnu.no` med brukernavn `morradi`.

ANDRE BRUKSOMRÅDER

Men hva om du trenger å gjøre mer enn bare terminal-ting? Du savner kanskje å kunne bruke editoren du har på PC-en din? Heldigvis kan SSH hjelpe her også. For eksempel kan du bruke «Remote-SSH» funksjonaliteten til VSCode. Denne bruker *sshfs* som lar maskinen henge seg på filsystemet til serveren, og lar deg jobbe i din lokale editor på filer som ligger hos serveren. Om det er andre programmer du vil kjøre på serveren som trenger et grensesnitt, for eksempel en simulator eller lignende, kan du bruke X-forwarding. Send med `-X` flagget til SSH-kommandoen, og hvis du nå kjører programmer med grensesnitt vil grensesnittet dukke opp på din lokale maskin i stedet for serveren!



På generalforsamling bruker Abakus preferansevalg, av typen enkel overførbar stemme (STV, fra engelsk). STV er en valgordning som sikrer proporsjonal representasjon samtidig som at kandidater kan få flertall fra kun én valggrunde. Ulempen med STV er at det kan være vanskelig å forklare eller forstå (som vist på sekundær generalforsamling), men med en grundig gjennomgang i `readme` skal vi få alle abakuler til å forstå systemet.

HVORDAN FUNGERER DET?

Når man stemmer med STV, rangerer man kandidater i stedet for å velge en favoritt. Rangeringen av alle kandidatene kalles en stemmeliste. Etter at alle har stemt, går man gjennom en prosess med tre steg som gjentas helt til alle plassene som skal fylles opp, er fylt opp. De tre stegene er: sjekk om noen har nok stemmer, overfør overflødig stemmer og eliminer.

Sjekk om noen har nok stemmer: For å bli valgt i STV, må man ha $\frac{\text{Antall stemmer}}{\text{Antall plasser} + 1} + 1$ stemmer, rundet ned. Det vil si, dersom det er 100 som stemmer og tre plasser som skal velges, må man ha 26 stemmer for å bli valgt inn. I første stemmerunde gjelder kun førstevalget på alle stemmelister. Dersom noen har nok eller flere stemmer enn det som trengs, velges de inn.

Overfør overflødig stemmer: Dersom noen blir valgt inn med flere stemmer enn det som trengs, overføres overflødig stemmer til andre kandidater. Hver kandidat får da disse stemmene lagt på stemmene de har fra før:

$$\frac{\text{Antall overflødig stemmer}}{\text{Totalt overførbare stemmer}} * \text{antall stemmer på kandidaten}$$

hvor «antall stemmer på kandidaten» er samlet antall stemmer hver kandidat hadde fått om kandidaten som ble valgt inn ikke eksisterte, og «overførbare stemmer» er stemmer fra stemmelister som har flere kandidater igjen. For eksempel, dersom en stemmeliste bare har eliminerte kandidater, er stemmen ikke overførbar.

Eliminer: Dersom ingen kandidater har nok stemmer til å bli valgt en runde, elimineres kandidaten med færrest stemmer. Stemmer på denne kandidaten overføres da til personen under den eliminerte kandidaten på hver enkelt stemmeliste der toppkandidaten er eliminert.

EKSEMPEL

For å gjøre det litt lettere å forstå hvordan STV fungerer, skal vi prøve oss på et eksempel. La oss si at det skal være et middagsselskap, og gjestene skal stemme over hvilke matretter som skal spises. I dette eksempelet er det tre matretter som skal velges, og 23 gjester som skal stemme. Etter å ha stemt, ser stemmelistene slik ut:

Førstevalg	Pizza	Pizza	Kake	Sushi	Burger	Komle
Andrevalg	Kake	Burger	Sushi	Kake	Komle	Kake
Tredjevalg	Komle	Kake	Pizza	Burger	Kake	Sushi
Antall stemmelister	4	4	5	4	3	3

Steg 1: Først teller man alle førstevalg, og da ser stemmene slik ut:

Rett	Pizza	Burger	Kake	Sushi	Komle
Antall stemmer	8	3	5	4	3

#107

TING DU IKKE TRENGER Å VITE SOM SIVING

STV

Tekst: Vegard Ervik

Ettersom man ifølge formelen trenger seks stemmer for å bli valgt, er det bare pizza som blir valgt i første omgang.

Steg 2 og 3: Siden pizza ble valgt med to stemmer mer enn det som trengs, overføres disse stemmene til andrevalgene på stemmelistene der pizza var førstevalg. Etter formelen får kake og burger én stemme hver. Ettersom en rett ble valgt inn, hopper vi over elimineringssteget denne runden.

Steg 4 og 5: Etter overføringen av stemmer fra steg 2, ser fordelingen slik ut blant de resterende kandidatene:

Rett	Burger	Kake	Sushi	Komle
Antall stemmer	4	6	4	3

Ettersom kake nå har seks stemmer, blir kake valgt inn som den andre matretten. Kake fikk ingen overflødig stemmer, så det overføres ingen stemmer

Steg 6: Siden ingen matretter nå har seks stemmer eller flere, elimineres kandidaten med færrest stemmer, som er komle. Neste kandidat som ikke er valgt inn på stemmelistene til komle er sushi, så de tre stemmene overføres dit.

Steg 7: Antall stemmer ser nå slik ut:

Rett	Burger	Sushi
Antall stemmer	4	7

Ettersom sushi nå har nok stemmer til å bli valgt, velges sushi, og middagsselskapet har fått tre retter alle kan være fornøyd med.

Versjonskontroll: Noe du tar for Git?

Jeg husker godt første gang vi fikk en gruppeoppgave om å lage en nettside i IT1 på videregående. Taktikken var at ingen måtte redigere på samme fil samtidig, og vi sendte hverandre de oppdaterte filene på Messenger.

Heldigvis er det ikke sånn vi holder på i dag. Etter å ha lært å bruke Git i «IT1901 Informatikk prosjektarbeid I», ble vanene mine endret. Jeg kan ikke lenger se for meg å samarbeide med noen uten å bruke det. Imidlertid er det sånn at Git ikke ble lansert før 2005, hva gjorde folk før det? E-posttråder som branches? I tillegg hører man innimellom om andre versjonskontrollsystemer som «Subversion» og Metas «Sapling». Hva er det som skiller disse systemene fra hverandre?

TRE MODELLER

Om man går litt gjennom historien til versjonskontrollsystemer, finner man tre tydelige paradigmer som kan tenkes på som tre generasjoner av versjonskontroll. Hver generasjon ga en betydelig fordel over den forrige, og vi kan nå si at vi bruker tredje generasjons versjonskontrollsystemer når vi utvikler.

LOKAL DATAMODELL

De enkleste versjonskontrollsystemene kom av et behov for å holde styr på endringer i filer, og muligheter for å gå tilbake i historikken. I tillegg var det et ønske om å unngå at flere redigerte på samme fil samtidig og å kunne se hvem som har endret hva i en fil. Tidligere systemer gikk ut ifra at utviklere jobbet i samme filsystem. Med andre ord lå all koden på én PC, og utviklere måtte logge seg inn på denne PC-en fra sine egne. Versjonskontrollene var på en per-fil-basis, og du kunne ikke registrere endringer over flere filer. Du kan sammenligne det med versjonsloggen i Google Docs på det punktet. Det tidligste populære versjonskontrollsystemet var SCCS, som ble lansert i 1972.

KLIENT-TJENER-MODELL

Versjonskontrollsystemet CVS ble lansert i 1986 og startet den andre generasjonen. Revolusjonen var at utviklere nå jobbet på hver sin datamaskin, og var koblet til en felles tjener som holdt på kodebasen. Når en utvikler ville endre på noen filer, kunne de sjekke ut kodebasen, og den ville bli kopiert til deres lokale filsystem. Dette gjorde at mange utviklere kunne jobbe på samme fil, samtidig! Dette høres kanskje litt ut som det vi har i dag, men forskjellen var at når du sendte en *commit*, sendte du det rett til tjeneren, og alle andre utviklere ville få oppdateringen som en automatisk *merge*. Om det ble en konflikt, måtte de håndtere den der og da. Dette er det nærmeste versjonskontrollsystemer har kommet en *jumpscare*. En annen kritisk detalj var at utviklere måtte ha nettilgang hele tiden mens de holdt på. I 2000 kom Apache ut med systemet «Subversion» som introduserte commits som kunne inneholde flere filer.

Tekst: Vettle Roos Mangrud
Layout: Elisabeth Doan

DISTRIBUERT DATAMODELL

I 2005 lanserte Linus Torvalds et nytt versjonskontrollsystem, Git, for å holde orden på kildekoden til Linux, som han også er skaper av. Det store nye i Git var at man gikk bort fra at alle var koblet til en sentral kodebase. For å redigere kode, må du klonen en kodebase til din egen maskin, og gjøre endringer på denne lokale kopien. Endringene du gjør blir ikke sendt noe sted, og du trenger derfor ikke nettilgang når du redigerer noe. Om du skal sende noe til eller fra en annen kodebase, må du kjøre en *pull*- eller *push*-prosess manuelt. Det er under disse prosessene man ofte må håndtere merge-konflikter. Det man ofte gjør er å designere én kodebase som «hovedkodebasen» og laste den opp på en skytjeneste. Om du bruker GitHub, har GitHub hovedkodebasen, men alle utviklerne har hver sin kodebase. Man kan i teorien bruke Git uten en slik hovedkodebase, men man har sett at det ofte er lettere å organisere når man har det. Ved siden av Git er det også verdt å nevne Mercurial her. Mercurial er et annet distribuert system som kom omtrent samtidig som Git. Det blir fortsatt brukt av noen, men er definitivt ikke like populært som Git.

VEIEN VIDERE

For oss som studerer programmering nå, er det vanskelig å se for seg en verden uten Git. Git har vært så dominerende så lenge, og å foreslå noe annet er utenkelig. Betyr dette at Git er det perfekte versjonskontrollsystemet? Vil vi fortsatt bruke Git om 100 år? Om jeg skal være helt ærlig er det ganske forståelig om fremtidens utviklere tenker tilbake på Git som noe komplisert og frustrerende. Bare hør på stønningen fra fortvilte abakuler på A3 hver gang de får en *merge*-konflikt eller en «detached HEAD»-feil. Git har nemlig et par problemer som venter på en løsning:

GJØRE DET MINDRE KOMPLISERT

Git er veldig fleksibelt i hvordan du ønsker å organisere versjonskontrollprosessen. Ønsker du *merge* eller *rebase*? Hvilken forgreningsstrategi vil du bruke? Denne fleksibiliteten fører til at Git kan være komplisert, og man må legge inn nye verktøy for å passe på at utviklere ikke bryter konvensjoner man setter. Metas nye «Sapling»-system er et versjonskontrollsystem som forenkler utviklingen med å fjerne litt av fleksibiliteten til fordel for konvensjoner som gjør at utviklerne gjør ting likt.

STØTTE ANDRE TING ENN TEKST

Git fungerer veldig bra når kodefilene våre er ren tekst. Sleng inn noen JavaScript-filer eller C-filer, og Git håndterer dem som en mester. Du får klare differanser og Git yter veldig effektivt. Prøv derimot å holde kontroll på endringer i en fil som ikke er ren tekst, som et bilde eller en binær-fil, og du får mye mer trøbbel. Til og med Python Notebooks er litt strevsamt å holde kontroll på, og de er nesten bare tekst. Det har kommet nye versjonskontrollsystemer, deriblant Helix Core, som prøver å forbedre måten vi jobber med andre type filer.

En kort introduksjon til hudpleiens verden

For å oppnå en hud som både ser og føles bra ut, er det viktig å følge en god hudpleierutine. Å bruke produkter som er tilpasset din hudtype er essensielt

for å få maksimal effekt av produktene du bruker. Om ikke annet spesifiseres, bør stegene gjennomføres både morgen og kveld.

0 SMINKEFJERNER

Om du har på sminke, fjern denne med en sminkefjerner som passer til både hudtypen din og sminkeproduktene du har brukt. Vær oppmerksom på at servietter kan være praktiske, men kan oppleves som irriterende og tørke ut huden over tid.

1 RENS

Vask ansiktet med en *cleanser*. Dette er spesielt viktig om kvelden, etter at huden din har vært utsatt for dagens påkjenninger. Cleanseren fjerner talg, smuss og eventuelle rester av sminke, og forbereder huden for påfølgende produkter.

2 EKSFOLIERING

Bruk en mild skrubbe eller syre (som AHA eller BHA), for å fjerne døde hudceller og fremme cellefornyelse. Dette resulterer i en hud som føles glattere, og hjelper sminken til å legge seg penere på huden. Merk at dette anbefales å gjøre kun 1-3 ganger i uken, og man kan velge å gjøre det morgen eller kveld, men ikke begge tidspunktene. Ved bruk av syrer, blir huden mer følsom for sollys og det er dermed ekstra viktig å påføre solkrem dagene etter å ha brukt slike sterke produkter.

3 TONER

Den viktigste oppgaven til en god *toner* er å balansere hudens pH-verdi og dermed hjelpe etterfølgende produkter å trekke inn i huden. Den vil i tillegg fjerne overflødig olje, myke opp huden og bidra til naturlig glød.

4 BEHANDLINGER

Dette trinnet inkluderer serum, retinol, ansiktsmasker og andre produkter som ikke må brukes hver dag. Velg produkter basert på hudens behov, som hydrering, rensing eller beroligelse. For eksempel er retinol kjent for sine anti-aldringsegenskaper, men øker hudens følsomhet for sollys, også her er det ekstra viktig å bruke solkrem de påfølgende dagene.

5 FUKTIGHETS- & SOLKREM

En god fuktighetskrem bidrar til å opprettholde fuktighetsbalansen i huden, og forhindrer tørrhet, stramhet og flassing, noe man spesielt kan føle på i tørre klimaer eller om vinteren. Dette er med på å styrke hudens naturlige barriere, og bidrar til å redusere synlige aldringstegn.

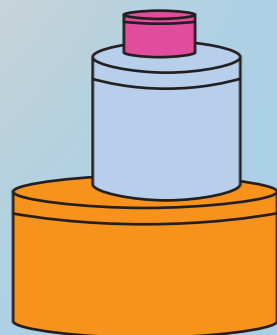
Påfør alltid solkrem om morgenen, som blant annet beskytter mot UV-stråler og kan redusere risiko for hudkreft.

ENKELTE PRODUKTER SOM IKKE BØR BLENDES:

Retinol og syrer: Unngå å bruke sterke syrer sammen med retinol, da dette kan virke aggressivt for huden. Bruk dem med et par dagers mellomrom.

Vitamin C og retinol: Disse kan brukes samme dag, men det anbefales å bruke dem henholdsvis morgen og kveld, for å unngå irritasjon.

Sterke syrer og andre aktive ingredienser: Vær forsiktig med å blande sterke syrer med andre aktive ingredienser som benzoylperoksid, da de kan nøytralisere hverandre og dermed redusere effekten når de påføres samtidig.



Tekst: Martine Meng Karlsen
Layout: Lea Aas-Jakobsen

HVORDAN FUNGERER anbefalingsalgoritmer EGENTLIG?

Tekst: Eli Fjellbirkeland Johannesen

Layout: Fredrik William Husemoen-Zhang

De fleste nettsider, enten det er nettbutikker eller underholdningstjenester, har et system for å anbefale deg nytt innhold som de tror interesserer deg. Selv om disse anbefalingssystemene er avanserte, bygger de ofte på de samme, enkle teknikkene.

CONTENT FILTERING

Content filtering er en annen maskinlæringsteknikk som blir mye brukt i anbefalingssystemer. Denne metoden bruker brukernes interesser og filmenes attributter som grunnlag for anbefalingene. Dette kan man sette opp i to matriser:

	KOMEDIE	EVENTYR		KOMEDIE	EVENTYR
PER	5	3	FLÅKLIPA	3	2
PÅL	3	4	OLSENANDEN	4	3
ESPEN	1	5	BRØDRENE DAL	5	4
ASKELODDEN	4	5	KON-TIKI	2	4

FIGUR 2

For å finne ut om Per liker «Olsenbanden», ganger man sammen de to matrisene for å finne ut om Per kommer til å like filmen. Først ganger man sammen komedie-kolonnene: 5×4 . Deretter eventyr-kolonnene: 3×3 . Til slutt plusser man det sammen: $20 + 9 = 29$. For å få en bedre forståelse av resultatet kan man normalisere dette tallet for å få det på en skala på 1-5.

HVORDAN BLIR DETTE ANVENDT I ET STØRRE SYSTEM?

Både collaborative filtering og content filtering har sine fordeler og ulemper. Derfor kombinerer man ofte de to teknikkene for å få på plass et system med gode og varierte anbefalinger. Systemene henter også inn store mengder data om brukere og innhold for å ha flere punkter å binde sammen bruker og innhold på. Netflix bruker blant annet når på døgnnet du pleier å strømme, og hvor lenge du pleier å strømme, aktivt i sine anbefalinger.

COLLABORATIVE FILTERING

Collaborative filtering er en maskinlæringsteknikk hvor man baserer anbefalingene på hva andre brukere med like interesser har likt. For et anbefalingssystem for filmer, henter man inn hvilke filmer brukeren har sett og hvilken vurdering brukeren har gitt filmen. Man kan sette hele systemet opp som en matrise, med brukere som rader og filmer som kolonner.

	FLÅKLIPA	OLSENANDEN	BRØDRENE DAL	KON-TIKI
PER	5	?	4	2
PÅL	4	2	1	4
ESPEN	5	3	4	3
ASKELODDEN	4	2	1	3

FIGUR 1

I hvert felt står vurderingen brukeren har gitt til filmen, mens verdier mangler der hvor brukeren ikke har sett den aktuelle filmen. Når systemet skal vurdere hvor godt Per kommer til å like «Olsenbanden», ser det etter hva andre brukere som ligner på Per synes om filmen. I matrisen ser vi at Per og Espen virker å ha veldig like interesser, og systemet vil da anta at de vil mene det samme om «Olsenbanden» også. Systemet antar derfor at Per kommer til å vurdere «Olsenbanden» til en treer.

MATRIX FACTORIZATION

Forestill deg en enorm matrise med alle filmene i Netflix sin filmkatalog som kolonner, og alle brukerne som rader. Det å bearbeide denne matrisen for å generere anbefalinger ville vært utrolig lite effektivt. Matrix factorization er en populær teknikk for å effektivisere disse enorme matrisene ved å splitte dem opp i mindre matriser av lavere dimensjoner, slik som illustrert i Figur 2. Ved å ta prikkproduktet mellom matrisene, slik som forklart i forrige avsnitt, får man ut verdier man kan bruke til å komme med anbefalinger. En endelig matrise for Per basert på content filtering vil se ut som i Figur 3, hvor vi ser at Brødrene Dal er den beste anbefalingen for Per.

	FLÅKLIPA	OLSENANDEN	BRØDRENE DAL	KON-TIKI
PER	21	29	37	22

FIGUR 3



Kalkulus lore

Tekst: Ian McGregor Aksum
Layout: Emil Moberg Værnes

Når du dupper av i en matteforelesning og drømmer om hvem du skal straffe når du erobrer verden, er det kanskje lett å se for deg foreleseren som første offer. Hen har tross alt fått deg til å punge ut 1200 kroner på Akademi for en bok du ikke kommer til å lese, og raser alltid gjennom lysbildene uten hensyn til menneskelig skrivehastighet. Men kan du egentlig skylde på foreleseren for at du må slite deg gjennom Matematikk 4?

TIDLIGERE FORLØPERE

Det er dessverre ikke foreleserens skyld at Epsilon-Delta-beviset eller Fouriertransformasjonen eksisterer og gir deg meritt om natten. Kalkulus er et gruppeprosjekt blant mange matematikere spredt over et tidsrom på 4000 år. De første vi vet om er egyptiske matematikere som hadde formler for å beregne ut volum og areal. I Moskva-papyrusen (1820 fvt.), der disse formlene er skrevet, finnes det dessverre ingen forklaring på hvordan de ble utarbeidet. (Hjelp, ring Graham Hancock!)

For å finne teori der arbeidet er vist, må vi vente til omkring 400 fvt., der den greske Eudoxus av Cnidus utviklet utfyllingsmetoden for å beregne volumer og arealer. Arkimedes videreutviklet arbeidet til Eudoxus og kom frem til heuristikk som ligner metoder i integral – kalkulus. Videre er Democritus den første kjente personen som skal ha tatt oppdelingen av en form i uendelig mange deler seriøst, men han var dessverre for glatthjernet til å akseptere at diskrete skiver av en kjegle og dens glatte overflate kunne likestilles. Generelt sett var grekerne litt for sta og aksepterte ikke konseptet uten geometriske bevis. Dette problemet ble ikke løst før 1600-tallet.

De greske filosofene hatet dessverre uendelighet og brukte heller bevis ved motsigelse for å utlede formler for geometriske størrelser. Det begrenset uheldigvis hvor generell teorien kunne bli, og det var ikke før algebraen forbedret seg at ting virkelig kom på plass. Rundt år 800 gjorde alle fortsatt matte med firkanter og kuber, ikke x-er og y-er. Derfor var for eksempel ikke x^4 – eller høyere – en

ting. Grekerne hadde studert masse kurver ved å ta forskjellige tverrsnitt av en kjegle, men hadde ikke et generelt system for å analysere former.

I tidlig middelalder skjedde det ikke så mye på europeisk jord, da europeerne var opptatt med å komme seg gjennom de mørke århundrer. Det ble gjort ingen matte, bortsett fra at algebra på et eller annet tidspunkt ble bra nok til å støtte kalkulusen som snart skulle bli til.

MODERNE FORLØPERE

Johannes Kepler utviklet en metode for å regne ut arealet av en ellipse ved å summere lengdene av mange stråler fra et punkt i ellipsen til omkretsen. Senere publiserte Bonaventura Cavalieri en avhandling basert på Keplers metoder, der han argumenterte for at volumer og arealer burde regnes ut ved å summere uendelig små deler. Dessverre så var Cavalieris metoder kjent for å lede til feilaktige resultater, og teoriene hans møtte sterk motstand blant andre matematikere.

Samtidig diskuterte mange matematikere i det 17. århundre ideen om en derivert. Pierre de Fermat introduserte konseptet «adequity» eller tilstrekkelighet, som representerte likhet med en uendelig liten feilmargin. Dette ble brukt til å finne toppunkter, bunnpunkter og tangente linjer til forskjellige kurver. Til slutt ble Cavalieris idé om uendelig små deler, en tidlig versjon av differensregning, og Fermats idé om tilstrekkelighet slått sammen til den første delen av analysens fundamentalteorem.

NEWTON OG LEIBNIZ

I løpet av det 17. århundret ble mange av ideene som underbygger moderne kalkulus skapt. Arbeidet var derimot ikke over før en komplett form av analysens fundamentalteorem var på plass, i tillegg til en formalisering av den tilhørende matten. Tilfeldigvis kom engelskmannen Isaac Newton og tyskeren Gottfried Wilhelm Leibniz frem til dette omtrent samtidig, og begynnelsen av det 18. århundret ble scenen for historiens største matte-duell.

Britene støttet Newton, som hadde startet arbeidet sitt som 23-åring. Den tysk-talende siden av Europa støttet Leibniz. Den sentrale problemstillingen i konflikten handlet om hvorvidt Leibniz hadde fått ideen om derivasjon og integrasjon fra Newton, eller kommet på det selv. Newton og hans tilhengere var nøye med å oppsummere og presentere tidslinjen

av Newtons arbeid og de tilhørende anklagene mot Leibniz. Leibniz sin side gjorde ikke det samme, bortsett fra Johann Bernoulli som gikk til personangrep på Newton. I tillegg viste det seg at Leibniz hadde endret på en rekke dokumenter han i utgangspunktet skrev flere tiår tidligere, før han publiserte dem i støtte for sin sak.

Dette førte til partiskhet til Newton blant mange av Europas matematikere, og *The Royal Society of London for the Improvement of Natural Knowledge* (der Newton var president) publiserte en rapport som fastslo at Newton var den første oppfinneren av moderne kalkulus. Leibniz anerkjente aldri denne rapporten og prøvde fortsatt å skrive ned sin egen versjon av historien om moderne kalkulus, men fullførte den aldri. Newton og Leibniz fortsatte å krangle frem til Leibniz døde i 1716.

Nå har vi beveget oss gjennom en lang rekke matematikere som sammen har gitt deg muligheten til å slå hodet i veggen for å banke inn en enda lengre rekke med teoremer. Men matematikerne er alle døde, så hvem skal få kjenne det når du glemmer alle teoremene på eksamen? Vi får håpe foreleseren ikke rekker frem til akkurat deg på runden sin.

Redaksjonen anbefaler

Bøker



BEFORE THE COFFEE GETS COLD

ANBEFALING FRA MADELEN



Mimrer du noen gang tilbake til tider som ikke er mer? Se for deg minner med gamle venner eller enklere tider der xx gjorde deg til det lykkeligste menneske på jord. Hva ville du gjort om du fikk oppleve en liten flik av det igjen? Det er denne boken i et nøtteskall.

Hele boken foregår i en liten café i Tokyo som tillater gjester å gjenoppleve et øyeblikk fra deres fortid. Vi følger fire karakterer som ønsker å besøke sine minner for å rette opp i noe de har gjort med sine kjære. Begrensningen er at besøket må ende før kaffekoppen blir kald.

Jeg opplever boken som omtentlig og nøkternt skrevet. Man kan ikke unngå å falle for karakterene og historiene de kommer med.

Sentrale tema er tap, kjærlighet og mellommenneskelige relasjoner. Det er en kort bok på omtrent 210 sider med et lettst språk. Tidsreise er det eneste urealistiske elementet, så sjangermessig er hjertevarm bedre dekkende enn fantasy. Boken finnes også på norsk: «Før kaffen blir kald»



MENGELE ZOO

ANBEFALING FRA FREDRIK WILLIAM



Hvis videregående har gitt deg et dårlig forhold til romaner, så er dette boken for deg som trenger å (gjen)opplive lesegledden!

«Mengele Zoo» forteller historien om den unge Mino og hans lidenskapelige kamp for naturen gjennom sitt dramatiske liv. Mino bor i en landsby i Sør-Amerikas regnskoger når et oljeselskap utsletter landbyen og myrder hele familien hans. Som eneste overlevende sendes 8-åringen ut på en usannsynlig reise gjennom kontinentene; En lang og dristig hevnreise mot alt som truer regnskogen.

Gjennom Minos øyne skildrer Nygårdshaug for oss en smittende kjærlighet for alt i naturen, flettet inn i brennende sterke romanser som får deg til å føle den uoppnåelige kjærligheten du aldri fikk oppleve som tenåring. Interessante problemstillinger vil få deg til å revurdere din moralske verdi, og kanskje vil du sympatisere med bisarre utsagn som «en ny atombombe hadde vært passende».

Dessuten er det en engasjerende thriller som er så hektende – og relativt lettlest – at du kan lese den ferdig på én helg om det er en frist for å skrive boka anbefaling den uka.



UTVIDELSE AV KAMPSONEN

ANBEFALING FRA JESPER



Sitter du tidvis og tenker: «Eksamener er skikkelig kjipe ass, men jeg føler ikke det er nok til å dra meg ned?» Da har jeg den perfekte anbefalingen for deg!

Houellebecq sin debutroman er den perfekte boken for å la den eksistensielle julekrise-stemningen synke inn. Den handler om en ensom og deprimeret dataingeniør (Næmmen!) – som lever et kjedelig og grått liv (Har du sett!) – sin reise gjennom Frankrike på et salgoppdrag fra jobben.

Det som drar denne boka fra en kjip beretning om en ensom manns liv til en klassiker av det beste slag er Houellebecqs evne til å bruke hovedkarakterens situasjon til å drøfte større filosofiske spørsmål. Om du ikke allerede følte en eksistensiell smerte i alle ledd, får du det i hvert fall når hovedpersonen overbeviser deg om at systemet alltid har vært, og alltid vil være, imot deg.

Boka er under 150 sider lang, og det er dermed ingenting som hindrer deg fra å plukke den opp fra nærmeste bokhandel og lese raskt gjennom.



Genus ønsker alle Abakuler masse
lykke til med eksamensperioden.

Vi heier på dere!

gen.us | gen.us/apply | @genusbiz

Utgavens konkurranse

Redaksjon <redacted>

Tekst: Lea Aas-Jakobsen

Siden julen nærmer seg tenkte **readme** å gi **readme** som en julegave til Internasjonal Delegasjon for Illustrasjoner, men de ville ha den sensurert! 🤯 Derfor må **readme** fjerne alle forekomster av «**readme**» i utgaven. Kan du telle antall «**readme**» du finner gjennom hele denne utgaven, så vi vet hvor mange «**readme**»-er vi må fjerne? Hvis du tror du har riktig antall, send inn antallet i en mail til konkurranse@abakus.no med emne: «antall **readme**»



SENSURERT



Svarer du riktig er du i tillegg med i den årlige trekningen av en middag for to til en verdi av 2000 kr.

sponset av Genus

VINNERE

1. Jakob Eilertsen Punnerud
2. Greta Gjengedal
3. David Kneringer Foss

PREMIER

1. premie



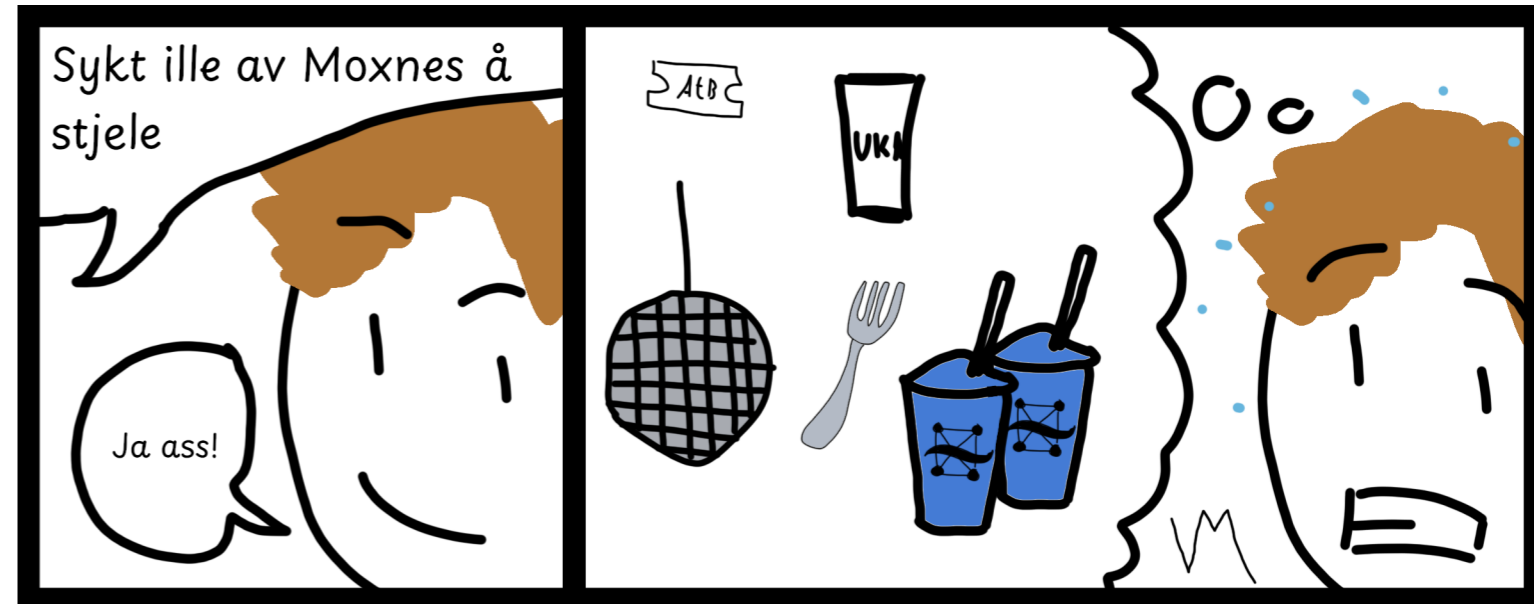
ØL x 24

2. premie



Red-/hvittvin

3. premie



tips & smått



Vil du ha snap på baksiden?
Send snap til **RYKTE!**

BARFØTT ABAKULE PÅ TUR

Etter å ha fått i seg litt for mange enheter på Heidi's, svartnet det for en abakule på tredje- til femteklassefest. Det siste vedkommende husker er at han snakket med politiet i området han bodde i og satt seg i en bil. Ikke overraskende våknet han neste dag i fyllerresten. Ifølge politiet skal han ha gått barbeint rundt i nabolaget og ringt på ulike dører. Abakulen bemerket at politiet var kompiser, og at de behandlet ham godt. Til slutt vil han fortelle at politet hadde Illustrert Vitenskap i arresten, men ikke **readme**. Der har vi en vei å gå.

RYSTEDE ABAKULER

Det er en kjent sak at det er relativt lavt under taket på LaBamba. For alle mennesker på to meter pluss er risikoen for å slå hodet alltid stor. Man skulle derimot ikke tro at to abakuler (under to meter red.anm.) fikk hjernerystelse samme kveld. Deres situasjoner var likevel vidt forskjellige, da den ene endte opp på legevakten (mot sin egen vilje) og ble vekket minst tre ganger på natten av en annen som var bekymret for vedkommendes helse, noe vedkommende ikke satte spesielt stor pris på. Hva den andre abakulen fant på vet ikke **readme**, men redaksjonen håper personen fikk behandling hos en legevakt og ikke endte kvelden i en grøft utenfor Moholt.

EN ELLER TO?

Det var etter en slitsom og alkoholfyllt kveld på en hyttetur ute i ingenmannsland at en utmattet abakule gikk og la seg for å sove. Målet var å få seg noen timer søvn før hjemreisen dagen derpå. Dessverre skulle ikke abakulen få den fredfulle blunden de håpet på. Etter et par timer søvn ble abakulen useremonielt vekket av en høylytt bjelle som ble ringt i det fjerne. Abakulen bestemte seg å ignorere tilkallelsen og la seg for å sove igjen. Dette skulle vise seg å være en vanskelig sak da gjerningspersonen bak bjelleklangen kom brasende inn på soverommet. Gjerningspersonen, i et tilsynelatende forsøk på å «oddse» abakulen, kom med replikken: «En eller to?» Det uheldige offeret, som åpenbart var opprørt over situasjonen, svarte gjerningspersonen med: «Kom deg ut!» Abakulen i nabosengen, som nå hadde våknet av livet, forsøkte hjelpende å løse situasjonen ved å si: «Jeg kan svare deg hvis du henter et glass vann til meg.» Gjerningspersonen funderte litt før de svarte: «Nei, jeg henter til meg selv» og vandret ut av rommet. Dagen derpå ble den skyldige avhørt, hvor unnskyldningen de kom med var: «Jeg må ha gått i søvne.» Forklaringen var tydeligvis en kombinasjon av alvorlig søvnmangel og stort alkoholinntak. Fra nå av anbefaler **readme** at den skyldige avstår fra kombinasjonen av brennevin og nattevandring.

