

# r e a d m e



# «Hva er det?»

Tips til |  
første date |  
Algdåt er endret, |  
hva er greia? |  
Spennende |  
spillbugs |

readme er gratis og blir skrevet av og for studenter ved Data- og Kommunikasjonsteknologi ved NTNU.

20. årgang nr. 4, 2018

## Redaksjonen

Redaktør  
Marius Sørensen Dreyer

Økonomi  
Andreas Hammer Håversen

Layoutansvarlig  
Magnus Rand

Medvirkende  
Ole Kildehaug Furseth  
Karoline Velsvik Berge  
Kristina Hovland Berg  
Mari Sofie Lerfaldet  
Magnus Schjølberg  
Henrik Fauskanger  
Sandra Skarshaug  
Stian Steinbakken  
Simen Holmestad  
Sigurd Berglann  
Tobias Skjelvik  
Synnøve Halle  
Eivind Kløvjan  
Viktor Solberg  
Vilde Arntzen

## Kontakt

E-post  
readme@abakus.no

Adresse  
readme, Abakus  
Sem Sælands vei 7-9  
7491 Trondheim

Nettutgave  
readme.abakus.no



### Stevnemøte?

Vi forteller deg hvor du bør og ikke bør dra!

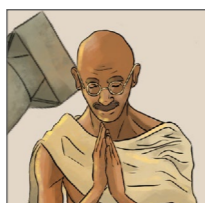
10



### INGInious?

Vi har gått til kilden og spurt om det nye øvingsystemet

12



### Spillbugs

Hvordan skrivefeil fører til atomkrig og bioterrorisme

18

# Leder

Redaktør: Marius Sørensen Dreyer

Velkommen til alle nye førsteklassinger, og alle som har kommet tilbake fra utveksling! Før du begynner på den innholdsrike lesingen, tenker jeg det er greit med en rask oppdatering. Christian Skau har dessverre gitt seg som foreleser i Diskret matematikk, Algoritmer og datastrukturer har fått nytt øvingsopplegg, og Morten Nome er fortsatt oppført som ansatt ved Institutt for matematiske fag.

Vi har – på grunn av den godt planlagte og oversiktlige samlokaliseringen – flyttet vekk fra det elskede Fraggelberget P15 og funnet vår plass i A-blokka i Realbygget. Selv om

det fortsatt er små ting som må fikses rundt om på datasalene, vil jeg si at vi har fått en real oppgradering, tross skadedyrproblemene et par etasjer under. Kanskje er det litt trist at vi må droppe alle vitsene om at heisen er treig, men det ofrer vi vel lett for en veranda med god utsikt.

Nå som samlokaliseringen sakte, men sikkert har trådt i kraft, har vi blitt flere studenter på Gløshaugen. Jeg håper dere tar alle godt imot, selv om det kanskje kan bli litt trangt til tider. Vi har god plass i A-blokka og et flott kontor å henge på, så jeg tror ikke vi er de som blir truffet hardest uansett.



# Er opptak så ille?

Tekst: Marius Sørensen Dreyer  
Foto: Magnus Rand

Linjeforeningsopptak er et hett tema i media for tiden. Tanken om at studenter må gjennom et hardt opptak for å få innpass i en linjeforening kan være skremmende, og har skapt mye diskusjon. Hvorfor har vi egentlig opptak, og har det noen hensikt?

Så vidt jeg vet har de fleste store linjeforeningene på Gløshaugen en eller annen form for opptaksprøve. Disse varierer veldig i lengde og hvor krevende de er å gjennomføre, men noe mange kanskje misforstår er om de faktisk er opptak i ordets betydning. De fleste linjeforeningene kaller det heller en prøve, og lar alle studentene deres være fullverdige medlemmer uansett om de har vært med eller ikke. Abakus har gjort det slik siden 2010.

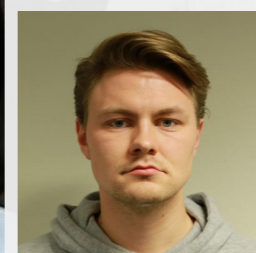
Om opptaksprøvene egentlig ikke har noe å si for om man er medlem eller ikke, hvorfor har vi dem? En naturlig forklaring kan være tilhørighet via tradisjon. For oss i Abakus er det tradisjon at de nye førsteklassingene får kle seg i listingpapir og løpe rundt på poster i Trondheim sentrum. Å vite at dette har skjedd i lang tid kan skape tilhørighet for nye studenter, men om det er nok til å forsvare alt det merkelige og ekle man ofte må gjøre er en annen sak. Må man lukte fisk, dytte muttere langs gulvet, bli dynket i mel eller drikke eddik for å føle tilhørighet til en linjeforening?

Tilhørighet kan skapes av mye, og om man ser på temaet til opptaksprøvene handler det ofte om å trække utenfor komfortsonen. Det krever noe av deg å løse oppgaver i en ukjent situasjon, og du lærer kanskje litt om deg selv når du gjør det. Om du også

gjør dette sammen med andre personer – som for eksempel under listingløpet – får du kanskje et ekstra sterkt bånd til disse. For de som har vært i militæret er det ikke vanskelig å kjenne igjen fenomenet. Ofte må du gjøre deg fortjent til å bære et avdelingsmerke eller et gitt hodeplagg, noe som kan skape vanvittig mye tilhørighet og mestringsfølelse. Spørsmålet er da om bading i fiskeslo er en god måte å få førsteklassinger ut av komfortsonen på.

Å sette et stempel på alle opptaksprøver er vanskelig. Er de unødvendige? Har de positiv eller negativ effekt? Slike spørsmål kan diskuteres opp og i mente uten at man nødvendigvis kommer veldig langt. Selvfølgelig finnes det både positive og negative sider, men så lenge ingen blir skadet og de fleste har en god helhetsopplevelse, er det vanskelig å se hvorfor de ikke skal arrangeres.

# Sosial kredittsjekk



**Magnus E. Schjøberg**

Født 20. oktober 1996 | Jobber på IKEA

Sosial score: 58

Siste hendelser:

- 10 Spilte «Super Smash Bros.» i 7 timer i strekk
- + 10 Søkte på «Det kommunistiske manifest»
- 10 Hørte på «Justin Bieber» på Spotify

Tekst: Magnus Eide Schjøberg  
Layout: Simen Holmestad

Samtidig som EU er i god gang med å innføre sin «General Data Protection Regulation» (GDPR) som strammer inn lovverket rundt personvern, går Kina i totalt motsatt retning. Mens GDPR har som mål at alle enkeltpersoner skal ha kontroll over hvilke data som samles inn om dem, er målet i Kina at alle skal overvåkes, kategoriseres og settes i bås. Til dette vil man benytte seg av masseovervåkning, Big Data®, Kunstig Intelligens™ og alle andre kule buzzwords som vi IT-studenter er så glade i.

## Når TV blir til virkelighet

«Social Credit System» er navnet på det nye systemet som er i ferd med å implementeres i Kina. Som en sterk kontrast til fokuset på personvern som har blusset opp i Vesten, virker systemet som tatt rett ut fra en episode av TV-serien «Black Mirror». Planen for dette systemet, som ble introdusert i 2014, la det frem som en måte for offentligheten å gi bedrifter og selskaper en karakter eller rangert vurdering. Det offisielle

målet med dette er å gjøre bedrifter ansvarlige for sine handlinger ved å innføre represalier for uetisk forretningspraksis. Dette tillater imidlertid i praksis også myndighetene å kontrollere bedriftene ved å innskrenke tillatelser og bevilgninger dersom de på noen som helst måte gir uttrykk for å være illojale eller kritiske til regimet.

Fremover skal systemet utvides til å rangere enkeltpersoner ut ifra pålitelighet, lojalitet og lignende. Et eksempel på hva som kan ha innvirkning på en enkeltpersons score er betalings-merknings, bøter og ens rulleblad. Enn så lenge er ikke dette så veldig spesielt – vi er vant med at slikt gir konsekvenser. Som et ledd i planen for systemet skal imidlertid flere overtredelser ha innvirkning på rangeringen. De fleste er godt kjent med at Kina har utbredt sensur og overvåkning av Internett, kjent under navnene «Golden Shield» og «Great Firewall of China». I tillegg til å blokkere alle nettsider som er kritiske til regimet og ikke håndhever sensur, planlegger myndighetene på sikt å bruke

netthistorikk til å stemple personer som upålitelige eller «farlige». Hva du søker på og hvilke sider du besøker kan med andre ord ha direkte innvirkning på din rangering, for eksempel hvis du bruker sosiale medier. Selv noe så dagligdags som å bruke for mye tid på eller juke i dataspill kan på sikt gi negativ utslag på personers rangering.

## Den (u)sosiale rangstigen

Å inneha en lav rangering kan potensielt få store konsekvenser. Du kan bli nektet lån, det kan bli umulig å finne seg jobb, ungene dine kan nektes god skolegang – listen er lang. I tillegg kan retten din til å reise med fly og offentlig transport innskrenkes på ubestemt tid. Du kan dermed ende opp nærmest som fange i et land som ser på deg som en aktiv trussel. Det er allerede opprettet «skoler» for å omskolere folk som ikke regnes som lojale til regimet. Disse skolene minner imidlertid mer om fengsler, med piggråd og vaktårn som skal sørge for lydighet. Men det er ikke det verste: For oss IT-nerder er det kanskje

spesielt én stor konsekvens som er verdt å bite seg merke i, nemlig at nettverkshastigheten din strupes kraftig. Hvis ikke dette får det til å gå kaldt nedover ryggen din, så vet ikke jeg.

Myndighetene bedriver også overvåkning ved å installere utallige overvåkningskameraer på ethvert gatehjørne. Disse kameraene benytter seg av ansiktsgjenkjenning og andre former for kunstig intelligens, noe som lar myndighetene holde et øye med potensielle trusler, og på forhånd registrere og slå alarm om potensielt farlige situasjoner og konflikter. Se for deg at du i nær fremtid står på T-banestasjonen og kaster fra deg søppel på bakken. Da har kanskje et kamera øyeblikkelig registrert og klassifisert dette, tatt et høyoppløselig bilde av deg, matchet dette med din nasjonale ID og gitt et passende trekk i rangeringen for dette.

Alt dette høres jo flott ut på papiret, men et mer dystopisk scenario kan fort tenkes: Hvis

myndighetene kan gjenkjenne enkeltindivider, er det da noe som stopper dem fra å overvåke ethvert individs bevegelser 24/7? Kanskje vil det også være negativt for rangeringen din å omgås andre som innehar en lav rangering? Vil man kunne straffes for å samles i større grupper eller organisere seg på andre måter? Er myndighetenes ultimate mål å isolere «upålitelige» individer totalt fra samfunnet? Dersom du vil se eksempler på hvilke måter dette kan utspille seg på anbefales det å ta en titt på *Black Mirror*-episoden «Nosedive» og *Community*-episoden «App developments and Condiments». Begge disse overraskende like episodene tar for seg et app-basert rangeringssystem som gir deg en rangering basert på din oppførsel, og gir et innblikk i isolasjonen som oppstår dersom du skulle havne nederst på rangeringen i et slikt system.

## Sesam sesam ...

Enn så lenge består systemet av fragmenterte og separate systemer. Et initiativ som allerede

testes ut og som inntil videre er frivillig å bruke er «Sesame Credit», som er utviklet av et underselskap av Alibaba. Dette er dessverre ikke penger du kan bruke på nattmat, men i stedet et rangeringssystem som benytter seg av data samlet fra Alibaba sine tjenester. Bare dette er en enorm datamengde, og den vokser seg stadig større. Andre separate tjenester testes også ut av andre selskaper og av myndighetene i mindre skala. Likevel, dersom de lykkes med å kombinere og forene disse datasettene og styre det via en sentral database, kan menneskerettighetssituasjonen i Kina fort forverres. Planen er å få alt standardisert innen 2020, og da vil man få se hvordan dette fungerer i stor skala. Vil Kina kunne øke produktiviteten til befolkningen og kun straffe de som fortjener straff? Eller vil vi ende opp med et system som aktivt forfølger politiske dissidenter og straffer alt annet enn total lydighet til regimet? Xi Jinping: Ballen er på din banehalvdel.



# De uformaterte

**Tekst:** Frikk Hald Andersen og Andrine Lossius  
**Foto:** Simen Holmestad  
**Layout:** Sigurd Berglann

Etter to års ventetid er det endelig tid for å møte opp ved Ohma Electra morgenen før imball. Selv om ryktene om et enklere og snillere listingløp har versert etter fjorårets oppslag i media, er det vanskelig å tro at dette vil gjelde for medlemmer av Hovedstyret.

Man kjenner litt på skammen der man går mellom alle førsteklasingene for å stille seg i køen utenfor EL-bygget. Det er ikke vanskelig å tolke blickene: «Hva gjør du her? Burde ikke alle i Hovedstyret være formatert?» Man nikker anerkjennende tilbake og forteller med blikket at «neida, man må faktisk ikke det». Det er nå tid for smukk, sløyfe og hevet paraply.

## En slitsom start

Det er vanskelig å ta inn over seg alvoret som tynger de 120 menneskene som står rundt oss. Frikk har vært funksjonær før, så mystikken forsvinner litt. Vi geleides inn i EL-bygget og nostalgien fra 2017 treffer oss idet den velkjente sakselyden runger gjennom Glassgården. På balkongen står Hovedstyret, og vi kan skimte de tomme plassene våre. Samtidig føler vi at vi ikke fortjener å stå på balkongen, ettersom vi ikke er formaterte enda.

Etter mye kjeft og utallige pushups og spenst hopp er det endelig vår tur til å dra ut på eventyr i Trondheim sentrum, iført undertøy og listingpapir. Vi starter ferden ned mot det runde røde huset der en gjeng med livlige webkommere venter oss på vårt første stopp. Etter ti minutters

tid med hendene i en balje vi ikke vil vite hva inneholder, er vi overbevist om at vi er lurt – minnepinnen og SD-kortet vi skal finne kan umulig være der.

Videre går ferden mot Domen, og vi lever i håpet om å være relativt rene når vi er ved veis ende. Dette fører til ekstremt laber innsats på neste post, og etter velfortjente fysiske straffelengder på grusplassen ved Nidarosdomen skjønner vi at slaget er tapt: Vi kommer til å måtte tømme Zalo-flasken i dusjen etterpå. Og nei, du franske turist, «we are not getting married».

## Oppløpet

Etter en poengfattig start skjønner vi fort at vi må reise kjerringa. Og etter hva redaktør Marius kalte en «Oscar-vinnende tolkning av 'Forrest Gump'», har vi fått blod på tann. Koskoms kaffe hjelper oss mot brekninger av fiskelukt, og vi er tilbake i kampmodus. Omega blir blåst av banen av et kunstverk som ville fått både Munch og Van Gogh til å stå opp fra de døde, og Andrine sjarmerer sine fagkommere i senk med Boris på Bakke Bru. Nå er vi i siget.

Vi nærmer oss slutten av dette underlige konseptet og begynner å merke påkjenningen. Heldigvis er dette en strålende formiddag, og det kryr av folk i byen. Gjennom Nedre Bakklandet skulle vi nesten tro vi var på oppløpet av Trondheim Maraton og blir heiet frem av nysgjerrige tilskuere. Det var nok ikke dette de så for seg da de satte seg

ned på Dromedar Kaffebar for en cortado og croissant.

Etter en spabehandling av den sjeldne sorten og imponerende språkkunnskaper i mandarin og spansk, er det tid for posten vi fryktet mest: Vi skal ut i en god gammeldags duell mot en PR-duo, der laks spiller en avgjørende rolle. Med Frikk som menneskelig skjold er det ingen tvil om hvem som kommer seirende ut. Videre er stikkord makrell i tomat, sirup, majones og fjær. Denne kombinasjonen er ikke noe våre sanser normalt setter pris på, men på dette tidspunktet er de så svekket at det ikke enser oss. PR står videre for å punktere oss med diverse fysiske anstrengelser før vi endelig kan vende nesen mot Glassgården med et bredt smil om munnen. Vi kan nå anse oss selv som formaterte abakuler.

## Tips og triks

- Plasser deg taktisk (les: ikke ved siden av en toppidrettsutøver med dårlig tid). Runden er cirka 6 kilometer, med x antall bakkeløp, spenst hopp og pushups.
- Vær glad! Det vil gjøre ting lettere. Vi snakker nå av erfaring.
- Barber leggene dine, ellers gjør gaffateipen vondt.
- Husk paraply, ellers blir det pushups!
- Veps er glad i sirup. Ikke bli stukket som Frikk.





# Glasløken

Jaggu, har du sett!

## «Chubby bunny» overtar Trondheim

**Tekst:** Eivind Kløvjan

MOHOLT – «Chubby bunny», spillet som har tatt studentmiljøet i Trondheim med storm, har nettopp hatt sin første turnering i studentbyen. Torsdag kveld ble **Glasløken** invitert med på første utgave av den planlagt ukentlige «Chubby bunny»-turneringen i kjellerne på Moholt. «Spillet går ut på å fylle munnen med så mange marshmallows som mulig, mens du prøver å si 'chubby bunny'», forklarte Ivar «Ivar» Hardstå, leder av komiteen Bunnykom i AF Smørekoppen og arrangementsansvarlig for turneringen. «Vinneren er den som klarer å si frasen med flest marshmallows i munnen samtidig, eller som sist spur tre ganger.»

Før turneringen begynte ble vår journalist invitert til å prøve spillet, og takket ja til dette. «'Chubby bunny' er et av en lang rekke spill som har dominert Trondheim gjennom årene», fortsetter Hardstå. På spørsmål om spillet gan være farlig: «Pfja, det er fvell shom med alle fhtudentpillene – det er kanfke igge hølt forfshfarlig, men bet er jo helt fykt lættish!» Hardsdå ønshger igge å uddale seg om det har bwidd gjemommoforb rishigoburdering for awwanshememget.

En mmy trfmadhifshon fhahr immmftrådt Ffdromdheim, fmfogf fdemme fhforaflishden bmmmbfå fmfhåpefr mfbe mgåw fbeiffbne bmfmsfhfhfshmsfhmfhshshfm mfmsfmmfmmfff.

## Skitten miljøvennlig student

**Tekst:** Stian Stein

TRONDHEIM – En student skal ha funnet et smutthull i samfunnets hygiene-stigma. «Ting jeg alltid har gjort som tidligere ikke har vært helt stuerent – for eksempel å bruke samme håndkle i flere uker, eller å ikke vaske sengetøyet på over en måned – kan jeg nå skrive av som miljøbevissthet», uttrykker den noe illeluktende studenten. «Jeg bruker de samme klærne flere dager på rad og blir hyllet som en helt for å minske det økologiske fotavtrykket mitt. Det er sykt chill: Jeg kan gå flere dager uten å dusje, og trekker man ned etter at man har vært på do så er man jo nesten som Trump-supporter å regne. Det at jeg ikke pusser tennene hver dag hjelper kanskje ikke så mye på miljøet, men vi sparer jo litt vann på det, da. Så det er jo noe.»

## Teflonpanne uten riper oppdaget

**Tekst:** Urdsig Lannberg

BERG STUDENTBY – Det var ved en rutinesjekk av et Sit-kollektiv i Berg studentby at kollektivkontrollør Hans Harr Svare oppdaget noe uvanlig. I hjørneskapet i et av kollektivene sto det nemlig en stekepanne i teflon – helt uten riper. «Gjennom mine 23 år som kontrollør har jeg aldri sett lignende.» forteller Harr Svare. Talsperson for kollektivet, Ronny Jonny Skei, ønsker ikke å kommentere saken ytterligere.



Utgavens master

# Hva faen er en metrikk?



**Tekst:** Odd Trondrud (M.Sc 2015), en person som definitivt jobber i Phonect

**Hva er en «algebra»? En «metrisk indekseringsmodell»? Vent, hva vil det si at noe er «metrisk»? Vent, vent, vet jeg hva en «indekseringsmodell» er? Hvis du ikke vet hva disse begrepene betyr, har du i hvert fall én ting til felles med meg anno 2014, da jeg uvitende om alt dette begynte på en prosjektoppgave med Magnus Lie Hetland som veileder.**

Masteroppgaven min var en fortsettelse av prosjektoppgaven. Jeg jobbet med disse greiene i et helt år. Jeg forstod absolutt ingenting da jeg begynte. En indekseringsmodell beskriver en framgangsmåte for å lage en struktur på en datamengde slik at en raskere kan få svar på visse spørsmål om datamengden. Indeksen i en bok gjør det enklere å finne fram til det du leter etter. Gitt riktig metode tar det mye kortere tid å sjekke om et tall finnes i en liste hvis den er sortert.

En metrisk indekseringsmodell er ment for mengder i metriske rom. Dette vil si at vi har

en veldefinert funksjon som forteller oss avstanden mellom to vilkårlige punkter i rommet der (1) denne avstanden alltid er lik eller større enn null, (2) hvis denne avstanden er lik null for to punkter, så er de det samme punktet, (3) avstanden fra  $x$  til  $y$  er lik avstanden fra  $y$  til  $x$ , og (4) trekantulikheten holder. Et enkelt eksempel (men sikkert ikke helt riktig hvis du spør en matematiker) på et metrisk rom er en uendelig todimensjonal flate hvor vanlige kartesiske koordinater benyttes og vektormatten du lærte på videregående kan anvendes. Metriske indekseringsmodeller tar utgangspunkt i disse fire antagelsene og strukturerer datamengden slik at det blir enklere å få svar på spørsmål som «Hvilke  $k$  punkter ligger nærmest  $x$ ?» ( $K$ -nearest neighbors) og/eller «Hvilke punkter ligger innenfor en rekkevidde  $r$  av  $x$ ?» ( $range$  query).

I masteroppgaven min beskrives tre metriske indekseringsmodeller: *Bisector Tree* (BST),

*Generalized Hyperplane Tree* (GHT) og *Nearest-Neighbour Approximating and Eliminating Search Algorithm* (AESA), og mine egne implementasjoner av disse i Julia. I tillegg implementerte jeg disse modellene i Julia-biblioteket som implementerer algebraen for metriske indekseringsmodeller som veilederen min, som skrev det nevnte bibliotek, jobbet med. Formålet var å sammenligne ytelsen mellom egen og veilederens algebraimplementasjon av disse. I tillegg til å teste ut både algebraen og biblioteket veileder hadde utarbeidet, testet jeg nok tålmodigheten hans med mine mange spørsmål fra et utgangspunkt hvor jeg verken visste hva en «metrisk indekseringsmodell» eller «formell algebra» var.

Det var slitsomt, men fornøylig; lærerikt, men ei enkelt; i perioder sjeleknusende, men til slutt belønnende; og jeg har ikke fått praktisk bruk for noe av dette i ettertid.



# Tips og triks for en Fantastisk første date

**Tekst:** Vilde Arntzen  
**Layout:** Marius Sørensen Dreyer

Forventningene er høye til første date fordi det setter standarden for forholdet videre – om et forhold eksisterer. Det er mange kleine fallgruver du helst vil unngå, men hvordan gjør du det? **readme** har svaret.

Som regel er vedkommende på første date rimelig fremmed. Derfor burde du velge et sted og en aktivitet hvor du kan snakke underveis. Hensikten med første date er jo tross alt å bli kjent med hverandre og å sjekke om kjemien stemmer. Det kan være lurt å velge omgivelser som er litt annerledes enn ellers – det gjør det ofte lettere å holde praten i gang. I tillegg vil du unngå aktiviteter som kino eller konsert, hvor muligheten for snakking er liten. Dette er aktiviteter du kan spare til en senere date!

Det kan være lurt å ikke sette lista for høyt. Det er hyggelig å dra på restaurantbesøk, men kanskje ikke på første date hvis du ikke kjenner personen så godt. Et restaurantbesøk vil få det hele til å virke veldig høytidelig, når du helst bare vil snakke litt med personen. På den andre siden vil du heller ikke ta med deg personen hjem for «Netflix and chill», da dette kan bli litt for nært og kjær når du ennå ikke kjenner personen. Det kan også tolkes som at du vil være noe mer «useriøs» og kanskje har baktanker med å invitere hjem til deg.

## Tur i bymarka

Ta med deg daten og termos med kakao på tur i bymarka. Miljøet er annerledes enn normalen og stemningen er hyggelig. Dessuten er det mye billigere enn å dra på restaurant, og sunt for dere begge. Det er et godt grunnlag for å la praten gå, og noe daten kommer til å huske.

## Aktiviteter på pub

Be med deg daten på en øl – men sørg for at dere drar til en bar som har noen andre morsomme aktiviteter i tillegg. Dere kan for eksempel dra på Lille London for å spille biljard, eller finne en pub som har shuffleboard – eller andre artige spill – der dere har muligheten til å prate om alt mellom himmel og jord.

## Sykkeltur langs Ladestien

Ta med deg sykkelen, eller lei bysykkel, og ta turen langs Ladestien. Det er fantastisk utsikt og flere steder dere kan stoppe og sette dere på veien – for eksempel Ladekaia. Ladekaia er et nedlagt kaianlegg langs Ladestien som har blitt renoveret av arkitektstudenter ved NTNU. Det har blitt et hyggelig møtested for en matbit eller kaffe. De arrangerer også quizkveld, som er en hyggelig aktivitet å ta med daten på.

## Kafé

Å gå på kafé er enkelt og hyggelig. Det er mange

## Huskeliste

- Husk lommeboka – det er kleint å be daten om å betale for deg.
- Vær interessert i daten din, og still spørsmål om dens interesser.
- Ikke snakk om tidligere forhold.
- Ikke ta matchende tatoveringer.

fine kafeer på Bakklandet hvor dere kan sette dere ned for en kaffe og ta en prat. Dette åpner også muligheten for å gjøre noe annet etter daten dersom det er stemning for det.

## Mat- eller bryggerifestival

I begynnelsen av august arrangeres Norges største matfestival, Trøndersk Matfestival, i Trondheim. Da kommer utstillere fra både Trøndelag og resten av landet for å selge lokalmat og andre matspesialiteter. Hvis du i tillegg vil nyte nybrygget øl av forskjellige smaker, arrangeres det samtidig en bryggerifestival. Her vil du ha muligheten til å smake på mye godt, samtidig som det er rom for å by på seg selv. I tillegg vil du jo kanskje få en stjerne i boken for å virke kultivert.

Trondheim er full av muligheter for en vellykket første date, så det er bare å ta initiativ og gripe sjansen. Lykke til!



# Algdats: Fremtiden er her

Et av NTNUs største emner har tatt en tidsreise. Hva er nytt? **readme** har blitt med på reisen og utforsket det nye øvingsystemet i Algdats.

Mange i tredje, fjerde og femte klasse vil ha «gode» minner fra øvingsystemet i *Algdats*. (Algoritmer og datastrukturer). Det over 15 år gamle øvingsystemet ville ikke ha vunnet noen designpriser. Det var i det minste funksjonelt, men gitt hvordan det så ut ville det vært nærliggende å tro at øvingsystemet ble laget en gang i 19-pil-og-bue. Den æraen er nå forbi. Med et helt nytt øvingsystem og et helt nytt språk har emnet aldri vært «freshere». Hvordan går det når studentene skal lære seg et helt nytt språk, mens fagstaben videreutvikler øvingsystemet samtidig som det brukes av over 1000 studenter?

## INGInious: tidenes genistrek?

Det nye øvingsystemet som tar rollen som Algdats DeLorean har fått navnet INGInious av sine belgiske utviklere ved Université Catholique de Louvain, og er ved første øyekast betraktelig mer attraktivt enn det gamle systemet. Om det er helt *knirkefritt* er en annen sak. Systemet er relativt ferskt. Det, sammen med knot i NTNU sine systemer, fører naturligvis med seg en del problemer – noe vi som tar emnet i år har fått erfare. Den første øvingen ble «avlyst» og godkjent for alle grunnet problemer med registrering av studenter, og innleveringen av de praktiske øvingene har vært trøblete.

For eksempel har man opplevd at øvinger som er korrekte har blitt avvist av systemet. Løsning: Send inn svaret igjen og igjen, frem til du enten går lei, eller systemet aksepterer øvingen din. Ikke akkurat sjarmerende. En av de mest populære funksjonene i det gamle systemet glimrer også med sitt fravær. Det nye systemet har nemlig ingen måte å rangere hastigheten til koden din på, noe som kanskje var det morsomste med de gamle praksisøvingene. Dette er dog noe fagstaben lover å få på plass, før eller siden. På tross av disse utfordringene har systemet blitt relativt godt mottatt, og vi forstår hvorfor. Det nye

systemet er mye enklere å forstå og bruke enn det gamle, og potensialet er stort for fremtidig forbedringer. Emneansvarlig Magnus Lie Hetland har følgende å si:

«*Det forrige øvingsystemet var i utgangspunktet noe jeg begynte å lage på tidlig 2000-tall, og det har senere blitt utvidet og modifisert av noen undasser o.l. oppigjennom, men har i lang tid hatt ca. den samme PHP-kodebasen, som ikke har vært super-vedlikeholdbar. Et problem er at det har vært vanskelig å få på plass nye features – og det har vært knotete å oppdatere innholdet i øvingene, så de har stagnert en del.*

*Ut over dette har det vært et praktisk problem at øvingsstaben hvert år måtte sette seg inn i det tekniske for å holde liv i serverne, og fikse ting når det gikk i stykker, etc. Så vi har i flere år lett etter et eller annet system som kunne overta, som hadde en mer moderne og ryddig kodebase, og som kunne driftes av teknisk gruppe heller enn av en skiftende fagstab av undasser, etc. Vi har vært flere som har gjort relativt grundige søk etter passende programvare, men det er svært begrenset hvilke systemer som faktisk passer.*

*Det (INGInious journ.anm.) ble foreslått til oss fordi det allerede var i bruk i et annet fag på Kalvskinnet. Etter en gjennomgang av systemets egenskaper, kom vi til at det hadde den funksjonaliteten vi trengte – noe ingen tidligere systemer hadde hatt. Instituttet hadde allerede en server kjørende, og vi kunne legge inn vårt fag i den. Teknisk gruppe var villige til å ta over drift og vedlikehold.»*

## Et ungt programmeringsspråk for unge studenter

Det nye øvingsystemet er dog ikke det eneste som er nytt. Mens man tidligere kunne levere øvinger programmert i studentens valg blant Python, Java eller C++, må man i år bruke det helt nye programmeringsspråket Julia. Og da mener vi virkelig *helt* nytt. Versjon 1.0 av språket ble sluppet 8. august, mindre enn en uke før semesterstart. Språket forsøker å plukke det beste fra et knippe

programmeringsspråk: lesbarheten til Python, den matematiske notasjonen til MATLAB og ikke minst kjøretiden til C. Alt dette gjør Julia til et utrolig hendig språk, i hvert fall på papiret.

«*Så hvorfor Julia? Noe av tanken bak Python i sin tid var at det var flere som syntes Java var vanskelig, og flere som ikke hadde lært det, men kunne f.eks. C++ eller MATLAB. Python var et enkelt alternativ, som lignet mer på bokas pseudokode, og som kunne være lettere å mestre for flere – nettopp så de kunne fokusere på faget. Men sekundært så er det et poeng at man eksponeres for flere programmeringsspråk; den gangen var noe av tanken min at jeg syntes det var uheldig at man bare skulle lære Java igjennom studiet. Jeg vil tro at flere kan si seg enige i det idag. Det har vært vanlig at ulike fag har hatt ulike språk, bl.a. for å sikre at man lærer seg flere. F.eks. hadde grunnkursene i sin tid Turbo Pascal, men idet man skulle begynne med objektorientering så begynte man å bruke C++.*

*En ting er at det er nyttig å lære seg flere språk som man kan få bruk for til spesifikke anvendelser, men det er nå også slik at jo flere programmeringsspråk man lærer seg, jo lettere blir det å lære seg nye språk senere, noe man trenger i mange yrker. Så å bli eksponert for flere enn bare ett eller to i løpet av studiet er helt klart viktig. Algdats er en del av «programmeringssøylen» i studieprogrammet ved IDI, og har et ansvar for å bidra til generell programmeringskompetanse – ikke bare ren teori.*

*I dette tilfellet er Julia et språk som er suverent velegnet til såkalt «technical computing», inkludert algoritmer, data science, numerikk, optimering, etc., etc. Med andre ord midt i blinken for et fag som dette. [...]*

Som vi ser er valget av språk velbegrunnet, men det betyr ikke at introduksjonen av språket har vært helt uproblematisk. Først og fremst er Algdats et ganske tungt emne i utgangspunktet. Det blir ikke noe lettere av at man må lære seg et helt nytt språk i tillegg, uansett hvor enkelt eller godt

det språket måtte være. Når det er sagt, er Julia et veldig enkelt og godt språk. Blant annet er det utrolig lett å oversette pseudokode til språket. Det er nesten bare å skrive det rett inn. Sånn sett kan vi virkelig se fordelene ved å bruke språket. Det flytter fokuset fra implementeringen over til selve algoritmen, noe som er ganske viktig med tanke på hva emnet faktisk skal handle om.

## Lys fremtid for Algdats

Så hva skal man si? Forandring er alltid utfordrende, uansett hvilken form den tar. Ekstra utfordrende er det når man skal prøve å røske opp og endre på systemer som har slått rot i over 15 år og som skal brukes av over 1000 studenter. Noe vi imidlertid kan konstatere med rimelig god sikkerhet er at Algdats har tatt steget til den fantastisk futuristiske verdenen i år 20-ikke-så-lenge-siden. Julia og INGInious er blitt det nye Algdats. Nye studenter vil ikke lenger måtte knote med det gamle øvingsystemet eller programmere i språk som alltid er treigere enn nabomannens.

Algdats er død. Lenge leve Algdats.

## Info:

- Det gamle øvingsystemet ble påbegynt tidlig på 2000-tallet.
- Det nye øvingsystemet INGInious ble påbegynt i 2014, og er skrevet hovedsakelig i Python.
- Siste stabile utgave av Julia, v1.0, ble sluppet 8. august 2018. Utviklingen startet i 2009.
- Algdats har i år over 1000 studenter. Forelesningene er derfor filmet, men ligger kun ute i en uke av gangen.

**Tekst:** Andreas Hammer Håversen  
**LAYOUT:** Magnus Rand

## Gammelt øvingsystem

TDT4120  
**Algoritmer og Datastrukturer**  
Dato endret: 1.09.2016

**Øvinger** **Teorioppgave - Datastrukturer**

**Blackboard** [Oppgave] [Levering] [Løsningsforslag]  
**Piazza** Innleveringsfrist: 11.09.2017 15:00

**IDI** NB! Du er ikke innlogget. Logg inn for å få frem det du har skrevet tidligere.  
**NTNU** Dersom du har levert øvingen fra før, vil det du har levert bli slettet dersom du klikker på "Vis alternativer".

**Logg inn** NB! Det du skriver i tekstfeltene blir ikke vurdert som en del av besvarelsen. De er kun til eget bruk, før du velger "Vis alternativer".

**Oppgave 1**

Anta du har en kø  $Q = \langle 4, 7, 32, 72, 3 \rangle$  hvor det bakerste elementet representerer hodet til køen. Alle deloppgavene refererer til denne køen.

**a) Hvordan vil køen se ut etter å ha kjørt ENQUEUE(Q, 3)? (5 %)**

**b) Hvordan vil køen se ut etter å ha kjørt DEQUEUE(Q)? (5 %)**

**c) Hva er det minste antallet ENQUEUE/DEQUEUE-operasjoner du trenger for at køen skal endres til  $\langle 3, 4, 7, 32, 72, 3 \rangle$ ? (5 %)**

Ønsker gjennomgang:  (Hvis krysser denne av ønsker et oppsett overfor skat til gjennomgang på en lignende måte)

**Oppgave 2**

Anta du har en stakk  $S = \langle 4, 7, 32, 72, 3 \rangle$  hvor det bakerste elementet representerer toppen av stakken.

**a) Hvordan vil stakken se ut etter å ha kjørt PUSH(S, 3)? (5 %)**

## Nytt øvingsystem

INGInious > TDT4120 - Algorithms and Data Stru... > Øving 2, teori: Datastrukturer Logged In as Andreas

**Øving 2, teori: Datastrukturer**

Question 1: Kø

Anta at du har en kø

$Q = \langle 4, 7, 32, 72, 3 \rangle$

hvor det første elementet representerer hodet til køen som definert i kapittel 10.1 i læreboken. De neste tre deloppgavene er uavhengige av hverandre; dvs. alle refererer til denne køen.

Hvordan vil køen se ut etter å ha kjørt ENQUEUE(Q, 3)?

$\langle 4, 7, 32, 72, 3 \rangle$

$\langle 3, 4, 7, 32, 72, 3 \rangle$

$\langle 4, 7, 32, 72 \rangle$

$\langle 4, 7, 32, 72, 3, 3 \rangle$

Question 2: Kø

$Q = \langle 4, 7, 32, 72, 3 \rangle$

Hvordan vil køen se ut etter å ha kjørt DEQUEUE(Q)?

$\langle 4, 7, 32, 72, 3 \rangle$

$\langle 7, 32, 72, 3 \rangle$

$\langle 4, 7, 32, 72, 3, 3 \rangle$

# Skogstur



Foto: Karoline Isvik Berge





Ting du bør vite som siving

# Infrastructure as Code

Tekst: Stian Steinbakken

Infrastructure as Code (IaC) er i stor grad det det høres ut som – å definere infrastruktur i kode. La oss ta en titt på hva dette innebærer og hvilke fordeler IaC bringer med seg.

IaC vil si å definere infrastrukturen sin – det være seg en server som står og kverner i Grimstad, private *clustere* eller maskiner i skyen – med maskinleselig kode eller definisjonsfiler. Se for deg at du har ansvaret for en server eller ti, og plutselig får mail om at hele systemet er nede. Du prøver febrilsk å finne ut hva som er problemet og huske stegene du fulgte da du satte opp det intrikate og komplekse systemet du har jobbet så iherdig med. «Hvordan var det nå igjen? Snakket denne serveren med denne, og hvordan var det denne delen hang sammen?» Dette slipper du dersom infrastrukturen er definert i kode. IaC gjør infrastrukturen i stor grad selvdokumenterende, og ikke minst lar det deg versjonere infrastrukturen din. I tillegg kan du også benytte deg av andre fordeler som kommer ved å skrive kode, som for eksempel å skrive tester.

IaC gjelder generelt, men i dagens samfunn går ofte tankene til skyen når vi snakker om infrastruktur. Når man begynner å leke med

skytjenester, gjør man dette typisk ved hjelp av et grafisk brukergrensesnitt, eller kanskje til og med via et *command line interface* hvis man er tøff. Her kan man gjøre alt fra å sette opp virtuelle maskiner og *hoste* nettstedet, til tung maskinlæring. Dette er vel og bra, men det er en del problemer med å gjøre slikt via slike grensesnitt. Først og fremst øker sjansen for menneskelig feil drastisk – naturligvis, siden det tross alt gjøres av mennesker. I tillegg er fremgangsmåten ikke dokumentert på noen som helst måte, med mindre personene som allokere ressursene manuelt dokumenterer hva de har gjort.

Alle de store skyaktørene tilbyr nå mulighet for å benytte seg av IaC. For eksempel tilbyr Amazon Web Services (AWS) *CloudFormation* hvor man kan skrive infrastrukturen sin i *.json* eller *.yaml*. I stedet for å klikke gjennom menyer på deres nettsider er det bedre å manuelt skrive definisjonsfiler og *deploye* via disse. For eksempel kan man skrive noe à la kodesnutten vist her, som ved hjelp av et par enkle kommandoer bygger applikasjonen din, og så fikser AWS resten. På denne måten blir infrastrukturen lik hver gang, dokumentert, versjonert og lesbar. Prøv det ut neste gang du skysetter!

```

“Parameters”: {
  “KeyName”: {
    “Description”: “Name of an
existing EC2 KeyPair”,
    “Type”: “AWS::EC2::KeyPair::
KeyName”,
    “ConstraintDescription”:
“must be the name of an existing EC2
KeyPair.”
  },
  “InstanceType”: {
    “Description”: “Amazon EC2
instance type”,
    “Type”: “String”,
    “Default”: “m4.large”,
    “AllowedValues”: [
      “t2.micro”,
      “t2.small”,
      “t2.medium”,
      “m3.medium”,
      “m3.large”,
      “m3.xlarge”,
      “m3.2xlarge”,
      “m4.large”
    ],
    “ConstraintDescription”:
“must be a valid EC2 instance type.”
  }
}

```

Ting du ikke trenger å vite som siving

# Lavender scare



Tekst: Eivind Kløvjan

Slå deg ned, slapp av, og nyt denne lille pausen i studiehverdagen. I dag skal vi snakke om krig, seksualitet og politikk.

Under «De glade 20-årene» og forbudstiden i USA, betød «homofili» noe helt annet enn det gjør nå. Konseptet «å være homofil» eksisterte ikke, ei heller terminologien omkring det – homofili var noe man *bedrev*. Til tross for at brorparten av befolkningen mislikte *folkene* i LGBTQ-miljøet, og lover mot sodomi gjorde det ulovlig, ble *drag-festene* de arrangerte sett på som god underholdning. Man skrev om det i avisene, flere kjendiser var åpent homofile og biseksuelle, og politiet så generelt den andre veien når store fester ble arrangert.

I de påfølgende tiårene ble situasjonen en annen. Omkring slutten på forbudstiden i USA i 1933 hadde folkets syn på homofili som et *moralsk onde* begynt å utbre seg. Desto verre var det at de nye alkohollovene i New York City – LGBTQ-miljøets aller største samlingspunkt – kun tillot alkoholsalg på «ordentlige» plasser, noe som naturligvis ikke inkluderte homofile og lesbiske nattklubber. Det var imidlertid ikke før 1950-årene at en reinspikka moralsk panikk omkring homofili oppstod. I kjernen av denne finner vi dagens tema, «Lavender scare».

Men hva faen er det?

Uttrykket «The lavender scare» ble og blir brukt av medier og historikere for å referere til en heksejakt på homofile statsansatte amerikanere rundt starten av den kalde krigen, tidlig i 1950-årene. Den er en parallell til den mer velkjente «red scare» – en tilsvarende heksejakt på statsansatte kommunister og sosialister, også kjent under navnet *McCarthyism*.

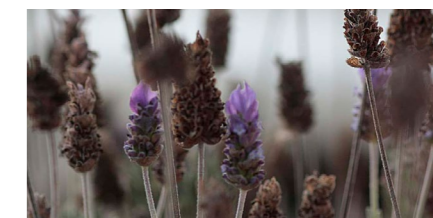
Hoveddrivkraften bak begge fenomenene, senator Joseph McCarthy, la mye energi i å overbevise folk om en sammenheng mellom homofili og kommunisme. Ett av hans mest effektive retoriske virkemidler var nettopp sammenslåingen av homofili, kommunisme og anti-nasjonalisme under paraplybetegnelsen *subversiv*. Et annet argument som viste seg å være veldig effektivt, var at homofile var ansett for å være mer utsatt for utpressing av kommunistiske spioner, nettopp fordi de kunne trues med å bli avslørt som homofile.

«425 statsansatte du ikke vil tro er homofile»

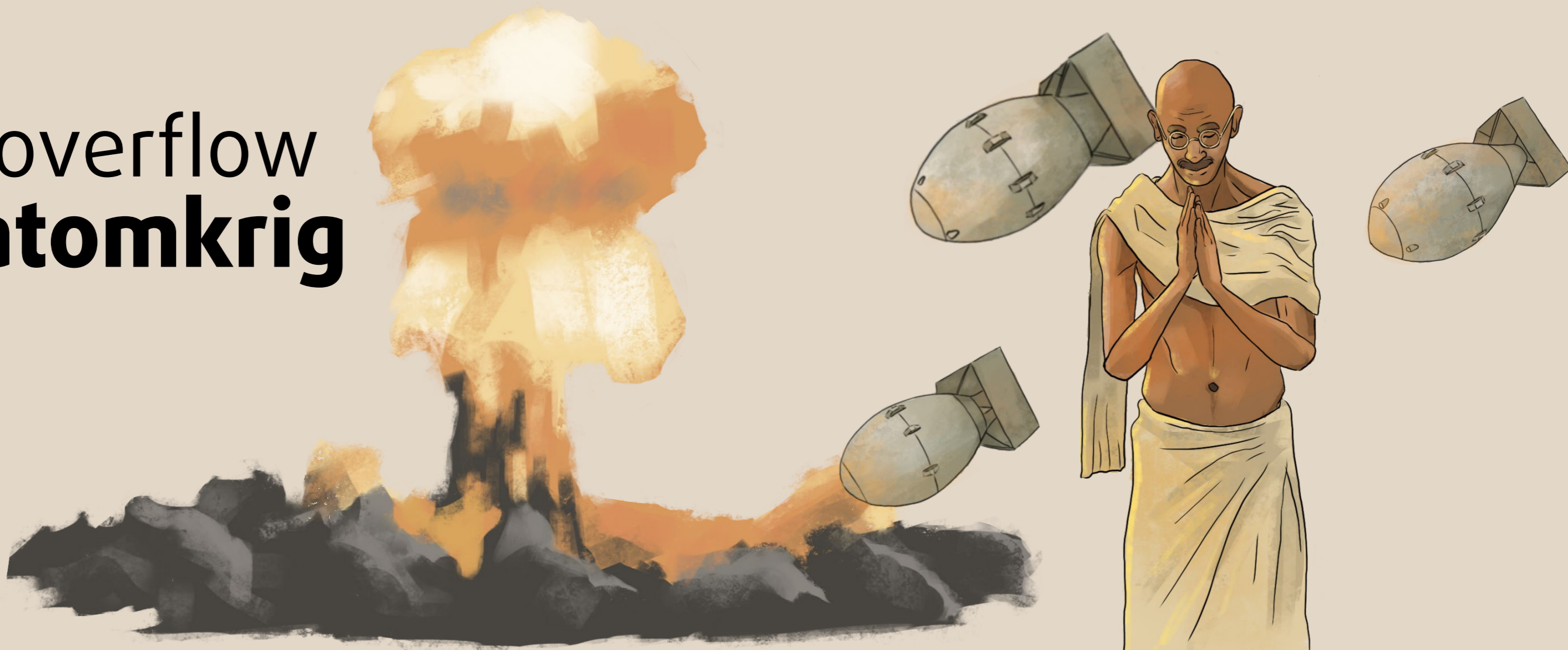
Opprinnelsen til navnet på fenomenet er uttrykket «lavender boys», som av samtidige tabloider og aviser ble brukt for å henvise til homofile menn, på en aldri så lite subtil måte.

Apropos tabloider: Disse utgjorde fronten av McCarthyistenes anti-subversiv-artilleri ved å spytte ut en kontinuerlig strøm av spekulasjoner omkring hvilke statsansatte som *denne* uken var De forente staters største sikkerhetstrussel. Det hele var så effektivt at da panikken ebbet ut rundt midten av 50-årene, hadde hele 425 «subversive» statsansatte mistet jobben som følge av homofilianklager, og en horde kjendiser hadde blitt avslørt som homofile og følgelig mistet sitt levebrød.

Det kan virke forunderlig at Lavender scare er viet langt mindre plass i historiebøkene enn sitt rødfargede motstykke, spesielt siden de negative følgene av førstnevnte påvirket langt flere folk. Årsakene til det er av kulturell og historisk art – og går langt utover omfanget til dette lille skriveriet. Det er uansett interessant å se at seksuell frihet i USA *ikke* har fulgt den lineære kurven fra totalt tabu til absolutt frihet som mange ser for seg.



# Fra overflow til atomkrig



**Tekst:** Ole Furseth  
**Layout:** Synnøve Halle

**Bugs dukker opp i all slags kode. Ved å teste ordentlig får man ofte avdekket de viktigste feilene, men noen ganger kan kritiske bugs snike seg forbi selv de mest oppmerksomme øyne.**

«It's not a bug – it's a feature» «Sid Meier's Civilization» ble en kjempesuksess da det ble utgitt i 1991, og er fortsatt blant de mest populære turbaserte strategispillene. Til tross for populariteten, var det første spillet i serien slettet ikke uten bugs. Det var nemlig opphavet til en av de mest ikoniske bugsene gjennom tidene, som til og med har blitt implementert som en *feature* i de senere spillene.

«Civilization» går ut på å lede en sivilisasjon fra førhistorisk tid til romalderen. Du tar på deg rollen som én av flere historiske ledere som

alle har ulike egenskaper. Når disse lederne kontrolleres av maskinen, må man ha en måte å bestemme oppførselen deres på. For eksempel er hver leders aggressivitet rangert fra 1–10, og er representert som et fortegnsløst heltall på 8 bit – det vil si at mulige verdier er fra 0–255. Blant lederne i spillet finner vi den ganske fredelige typen Mahatma Gandhi, med aggressivetsverdi 1. Etter hvert som spillet pågår, oppstår det imidlertid et problem: Lederne kan gjøre nasjonen sin demokratisk – noe som senker aggressivetsverdien til nasjonen med 2. De som har opplevd *integer overflow* før skjønner kanskje hva som skjer når Gandhi blir demokratisk: Istedenfor å bli –1, blir aggressivetsverdien 255, og brått virker Djengis Khan som en god nabo sammenlignet med krigshisseren Gandhi, som nå slenger atombomber i alle retninger.

I de senere spillene i serien er hyper-aggressiviteten til Gandhi blitt fikset, men utviklerne har likevel valgt å forevige bugen fra det første spillet. Dette har de gjort ved å gjøre Gandhi svært glad i atomvåpen – han er fortsatt en fredsforkjemper, men om han først må ty til våpen, gjør han det ordentlig.

## AI-lien

Da skytespillet «Aliens: Colonial Marines» ble utgitt i 2013, fikk det ikke særlig gode anmeldelser. Etter en trøblete seksårig utviklingsperiode, ble spillet lansert med underveldende grafikk og en mengde tekniske problemer, blant annet en svært uintelligent kunstig intelligens. Romvesenene, som i *Alien*-filmene med enkelhet kunne tilintetgjøre et skip fylt med topptrente soldater, minnet mer om gullfisker med hoggtenner i spillet, der de ofte virket å glemme hva de skulle

gjøre – gjerne mens de sto rett foran spilleren.

Spillet gikk raskt i glemmeboken, men det dukket opp igjen i sommer da en modder studerte konfigurasjonsfilene til spillet, og kom over følgende linje – ta deg et øyeblikk og se om du finner feilen:

```
ClassRemapping = PecanGame.PecanSeqAct_
AttachXenoToTether -> PecanGame.PecanSeqAct_
AttachPawnToTeather
```

Ordet «Tether» er nemlig feilstavet helt i slutten av linjen, noe som fører til at linjen ikke gjør noe som helst. Dette er ganske uheldig ettersom denne linjen skal la romvesenene vite hvor og hvordan de skal kjempe mot spilleren. Så snart skrivefeilen ble rettet opp, ble oppførselen til romvesenene vesentlig bedre.

## «The Corrupted Blood Incident»

«World of Warcraft» har holdt det gående i mange år nå, og har naturlig nok hatt sin andel bugs. En av de større hendelsene oppstod 8. september 2005 da de lanserte et nytt *dungeon*, eller *instance*, med vanskelige fiender som eksisterer separat fra resten av spillverdenen. Hovedfienden i denne instansen påførte en skadelig effekt på alle som angrep ham, og effekten spredte seg til andre spillere i nærheten på en viruslignende måte. «Viruset» skulle i teorien forsvinne fra en spiller etter litt tid eller når spilleren døde, og skulle i utgangspunktet ikke kunne forlate instansen. Takket være en bug kunne imidlertid viruset tas med ut av instansen og inn i den åpne spillverdenen ved å «smitte» kjæledyrene til spillerne.

Når de smittede kjæledyrene kom ut i

spillverdenen spredte viruset seg raskt. Alle ikke-spiller-karakterer – som kjæledyrene – kunne bære viruset, men var immune mot effektene, og var derfor perfekte smittebærere. Det tok ikke lang tid før det var apokalyptiske tilstander i byene og andre områder der mange spillere normalt oppholdt seg, og folk innså raskt at det var tryggere å holde seg til mer avsidesliggende strøk. Viruset ble ikke stoppet før spillserverne ble restartet sammen med en oppdatering nesten en måned senere.

Bugs er altså ikke til å komme unna – de dukker opp i alt fra ITGK-kode til spillkode verdt millioner av kroner. Neste gang du bruker en liten halvtime på å finne en åpenbar skrivefeil i koden din, kan du trøste deg med at den i det minste ikke har bidratt til bioterrorisme.

Ordspalten presenterer:

# FRISK

# #1



## Frisk /frisk/

«Norge er et land med mye vær», sies det. Hva det vil si at et land har mye «vær» er en diskusjon i seg selv, men denne utgavens ordspalte tar for seg noe helt annet. Vi nordmenn har, gjennom utspekulert språk, klart å tilegne oss et ord som gjør det å leve med «mye vær» lettere. Ordet oppfattes av de fleste som positivt, men skjuler bak sin fasade en mindre fordelaktig betydning. Som tittelen tilsier er dette ordet «frisk».

La meg komme med et knippe eksempler: «Jeg tok meg en dukkert. Vannet var friskt», «Friskt vær vi har i dag, du!» og «Plutselig kom det en frisk regnbyge.» Gjennom disse setningene får man inntrykket av at «friskt» er noe ønskelig og fordelaktig, men semantikken sier egentlig at «friskt» er synonymt med «kaldt».

Ved hjelp av ordet «frisk» har Norge klart å gjøre et av landets største ulemper om til en særegen og *norsk* fordel; det står det respekt av. Hatten av – og hetten på – for Norge!

Tekst og layout: Magnus Rand

# create-react-app og sånt

21

Tekst og layout: Sandra Skarshaug

Har du lyst til å prøve deg på litt *frontend*-utvikling, men vet ikke helt hvor du skal starte eller hvordan du setter opp et prosjekt? Da bør du lese videre.

Dersom du er ganske ny i frontend-verdenen, men har litt erfaring med JavaScript, ville jeg – basert på mine egne erfaringer – prøvd meg på React. Det er et effektivt og fleksibelt JavaScript-bibliotek for å lage brukergrensesnitt i webapplikasjoner. Ikke helt sikker på hva React innebærer? Viktige stikkord er små, innkapslede og gjenbrukbare komponenter, hvor disse komponentene kan ha *states*, men samtidig få *props* tilsendt fra andre komponenter. React holder styr på en virtuell *DOM*, og oppdaterer kun de endrede elementene fra denne i den faktiske *DOM*-en. I tillegg kan du benytte HTML-lignende syntaks, kalt *JSX*, som fungerer som en utvidelse av JavaScript. Google disse begrepene hvis du ikke kjenner til dem fra før! En av grunnene til at jeg anbefaler deg å prøve React, er at du kan komme fort i gang med utvikling ved hjelp av et behagelig verktøy som heter *create-react-app*, som resten av artikkelen vil handle om.

## Så hva gjør create-react-app?

Jeg sier at *create-react-app* er behagelig. Hva legger jeg i det? Verktøyet er laget for å slippe unna tidkrevende oppsett nødvendig i vanlige React-prosjekter. Dette gjøres ved å abstrahere bort konfigurasjonen ved hjelp av en pakke som heter *react-scripts*, som inneholder det

som skal til for å få et fungerende prosjekt. Videre setter det opp et utviklingsmiljø for deg, med en eksempel-app som er klar til å kjøre.

## Hvorfor bruke det?

Dersom du bruker *create-react-app* for å sette opp React-prosjektet ditt, slipper du å konfigurere og installere Webpack og Babel. Webpack er et verktøy for å transformere blant annet *.js*-filer og *.css*-filer til en optimalisert *bundle*, mens Babel blir brukt til å transpilere moderne JavaScript-syntaks – som ikke enda er støttet i alle nettlesere – til kode som er kjørbart i alle miljøer. Men, alt dette er jo ganske greit å ikke tenke på når du skal utforske frontend-verdenen, og derfor er det fint at *create-react-app* gjemmer disse skumle greiene for deg. I større prosjekter er det derimot fordelaktig å kunne skrive disse konfigurasjonsfilene selv.

Ikke nok med det, du får også litt mer snacks automatisk satt opp: *hot reloading* og en *progressive web app* (PWA). Det førstnevnte betyr at hvis du gjør endringer og lagrer vil du med en gang se endringene i nettleseren. En PWA er på sin side siste skrik innen webutvikling. I hovedsak har disse litt ekstra kode og metadata som gjør at de kan installeres på en mobiltelefon og nesten fremstå som «native» mobilapper.

## Hvordan komme i gang?

Sørg for å ha Node – en plattform for å kjøre JavaScript – installert med minst versjon 6. Du trenger også en pakkebehandler, enten npm

eller yarn – der npm allerede er installert når du har Node. Kjør `$ npm create-react-app prosjektnavn` i terminalen for å opprette et prosjekt. Naviger inn i prosjektet, `$ cd prosjektnavn`. Der kan du starte opp utviklingsserveren ved å kjøre `$ npm start` og *bam*, så har du en kjørende app på port 3000! Sjekk den ut i nettleseren på `http://localhost:3000`.

## Hva er de ulike filene?

Som nevnt skjuler *create-react-app* en del skumle filer for deg. Det viktigste du sitter igjen med er:

- **README.md** prosjektinformasjon i markdown
- **package.json** prosjektmetadata
- **node\_modules** mappe med avhengigheter
- **public** inneholder statiske filer
  - **index.html** har en `<div>` som appen legges i
- **src** inneholder koden vi vil endre
  - **App.js** eksempelkomponent
  - **App.css** styling for eksempelkomponent
  - **index.js** forteller hvor appen skal rendres
  - **index.css** styling for appen

Et lite tips på tampen er å lage en ny mappe i **src** som heter **components**, og legge alle komponentene du lager i applikasjonen her – det er litt ryddigere.

... Da er prosjektet satt opp! Nå er det bare for deg å lese dokumentasjonen på [www.reactjs.org](http://www.reactjs.org), samt å kanskje følge veiledninger – for eksempel på [www.codecademy.com](http://www.codecademy.com) eller [www.egghead.io](http://www.egghead.io) – for å lære deg mer React!

# Salt Bae

Historien bak legenden

*Forfatter: Viktor Solberg  
Tegning og layout: Kristina Hovland Berg*

Som sivilingeniørstudent her på *haugen* får du gang på gang dunket inn i den realfagsorienterte hjernebarken din at det er viktig å kunne teorien bak praksisen. Det hjelper jo tross alt ikke å ha et verktøy tilgjengelig hvis du ikke vet *hva* du skal bruke det til eller *hvordan* du skal bruke det.

Denne logikken – tenker undertegnede – er naturlig å ta et steg videre. Vi overfører derfor tankegangen til noe som står alle våre hjerter nær: *memes*. En kan virkelig ikke sette pris på et godt *meme* dersom en ikke kjenner til den grunnleggende teorien bak det. Len deg derfor godt tilbake og la deg forbløffe. Dette er nemlig en gjenfortelling av hvordan en helt ordinær mann fra Tyrkia – kun armert med salt, kjøtt og sabel (samt en god dose pågangsmot) – erobret hjertene til både Instagram-elskere så vel som hjertene til matentusiaster over hele verden.

## Internett-berømmelse

Mannen bak det lettsaltede Internett-fenomenet – kalt for *Salt Bae* på fagspråket – er Nusret Gökçe. Han kommer fra Tyrkia og startet sin reise mot Internett-berømmelse i 1983. Da ble han født inn i en søskenflokk på fem – kun et steinkast utenfor Tyrkias største metropol, Istanbul. Der opparbeidet han seg en upåklagelig arbeidsmoral. Allerede som 13-åring begynte han nemlig å arbeide lange dager som lærling hos en lokal slakter. Hans motivasjon: å åpne sin helt egen restaurant. Godt over det neste tiåret

tilbrakte han hos mange forskjellige restauranter. Han brukte i tillegg et år av livet sitt på å reise til Buenos Aires for å lære så mye som mulig om kjøttindustrien.

Etter mye hardt arbeid åpnet han i 2010 sin første restaurant i Istanbul, og hans kulinariske eventyr var for alvor i gang. Denne restauranten ble den første av flere i en restaurantkjede med det smått narsissistiske navnet *Nusr-Et Steakhouse*. En dag fikk han nemlig celebret selskap av en tyrkisk forretningsmann. Han ble så imponert over spisestedet at han bestemte seg for å investere tungt i den nyoppstartede restaurantkjeden. Slik vokste den tyrkiske forretningen seg fra én restaurant i 2010 til seks restauranter i 2017 – dette var kort tid før det kulinariske eventyret for alvor satte fart.

## Hmm, trenger salt

Dette tar oss tilbake til nåtiden. I dag har nok de fleste av oss for vane – eller uvane – å benytte fritiden vår til å hyppig scrolle gjennom Instagram. Og hvis du gjorde nettopp dette ved januar-tider i fjor så er nok sjansen stor for at også du fikk med deg en Instagram-video av en langhåret mann, med svarte John Lennon-briller. Med både stil og presisjon skar Johnny Depp-etterligneren seg gjennom et enormt stykke kjøtt, før han på grasiøst og elegant vis danderte retten med en solid neve salt. Dette var altså Nusret Gökçe som – kanskje uten å være klar over det selv – gjorde den i dag velkjente gesten uødkelig.

I løpet av kort tid fikk nemlig Instagram-videoen over femten millioner visninger og ble med det verdenskjent. Men hvordan gikk det egentlig videre med restaurantkjeden til vår tyrkiske mann? Var toppunktet allerede nådd?

«... en helt ordinær mann fra Tyrkia – kun armert med salt, kjøtt og sabel ...»

Tvert imot. I skrivende stund – snaut halvannet år senere – har antall *Nusr-Et Steakhouse*-restauranter mer enn doblet seg til hele fjorten stykker – en av disse midt på selveste Manhattan i New York. Her flokker det seg ivrige matentusiaster, kjendiser og andre matlystne folk – som kan sies å klatre på den øvre delen av lønningsstigen – for å bevitne *Salt Bae* i egen person. I denne restauranten spaserer han nemlig fra bord til bord og utfører rutinen sin – som oftest til stor fornøyelse blant publikum. I tillegg har han i dag over femten millioner følgere på Instagram. Her legger han flittig ut bilder og videoer for å bygge videre på det Internett-fenomenet *Salt Bae* har blitt i dag. Innleggene inkluderer ofte poserende kjendiser som har tatt turen innom restauranten hans, utelukkende for å få en neve salt drysset over middagen sin.





Utgavens konkurranse

# Mestertyven mot mobillåsen



Tekst: Viktor Solberg

En mørk og stormfull aften er vår beryktede mestertyv atter en gang ute på tokt. Han befinner seg i Trondheims mørke gater, og den velkjente lyden av sprø grus som knaser mellom asfalt og sko pryder lydbildet. Vår mestertyv stopper opp og trekker den fuktige trondheimsluften dypt ned i lungene. Som en hauk skanner han folkehavet foran seg, og etter en stund får han øye på et passende offer.

Knappt et øyeblikk senere har mestertyvens smidige hender vært på besøk i samtlige lommer og grabbet til seg alt av verdisaker. Han leter gjennom sekken sin, som nå er fylt til randen med tyvegods, og fisker opp en mobiltelefon. På baksiden av mobiltelefonen er det limt fast en huskelapp som avslører følgende regler om mobilens skjermlås:

- Streken starter nederst i venstre node.
- Streken går totalt gjennom fire noder. Dette inkluderer den første noden.
- Streken kan kun gå gjennom en enkelt node én gang.
- Streken kan kun bevege seg horisontalt, vertikalt eller på skrå mellom to noder som ligger inntil hverandre.
- Dersom feil kode blir tegnet inn 10 ganger på rad, blir mobilen låst for godt.

Hva er sannsynligheten for at mestertyven klarer å finne riktig kode for mobilen blir låst, dersom han vilkårlig gjetter seg frem etter disse reglene?

Hva venter du på? Bruk dine små grå, kom frem til et svar og send inn dette til konkurranse@abakus.no med tittelen «MMM».

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

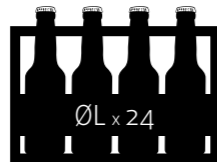
sponset av Genus

## Vinnere

1. Trude Jostad
2. Mattis Araya
3. Sevre Vestrheim

## Premier

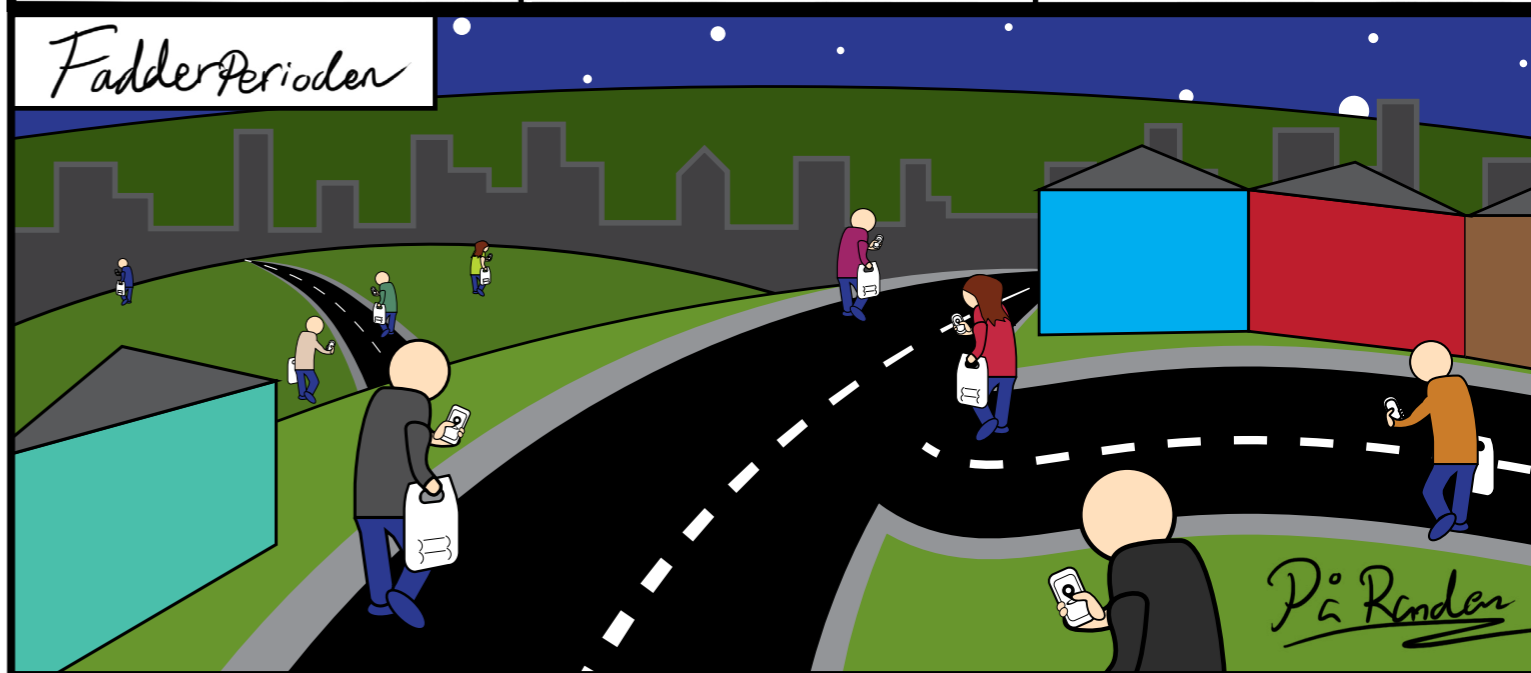
1. premie



2. premie



3. premie



# Smått & nett



– det du leser først!

## Utrolige korttriks?

Ababand sto ved baren på Provincialen før å kjøpe øl da en spennende kar kom bort for å prate. Han spurte om de ønsket å se noe de aldri hadde sett før. Bandet trosset barndommens regler om å ikke prate med fremmede, og ble med inn på et bakrom. På bakrommet møtte de enda en mann, som omsider begynte å vise dem diverse korttriks. Seansen varte i en god time, og avsluttet med at herren sa: «Om dere forteller dette til noen andre kommer de aldri til å tro på dere.» **readme** vet ikke om historien er troverdig eller ikke, men den virker iallfall svært tvilsom.

## Klokka er hva, sa du?

Ryktene har fortalt oss at et par komtekere fra første klasse var med et par datajenter hjem på badekarnachspiel etter immatrikuleringsballet. Natten ble lang, og de fant omsider sengen klokken 06:00 etter en del shots og mye god stemning. Stemningen snudde brått da den ene førsteklassingen sov gjennom tre alarmer han hadde satt kvelden før – han skulle nemlig på intervju med Start NTNU klokken 10:00. Syvsoveren ble vekt av verten klokken 9:30, styrtet en Redbull og sprang ut døren ikledd dress og en skjorte med ølflekker på. Det hele spisset seg til da førsteklassingen mistet bussen, og ble nødt til å løpe hele veien fra Kongens gate til Gløshaugen, der intervjuet skulle holdes. En time senere kom han tilbake til nachspielet, og kunne fortelle at intervjuet gikk glimrende. Hvordan komtekeren fikk til dette er ukjent for **readme**, men vi anbefaler at han i det minste melder seg inn i Abarun.

## Kos under kilten

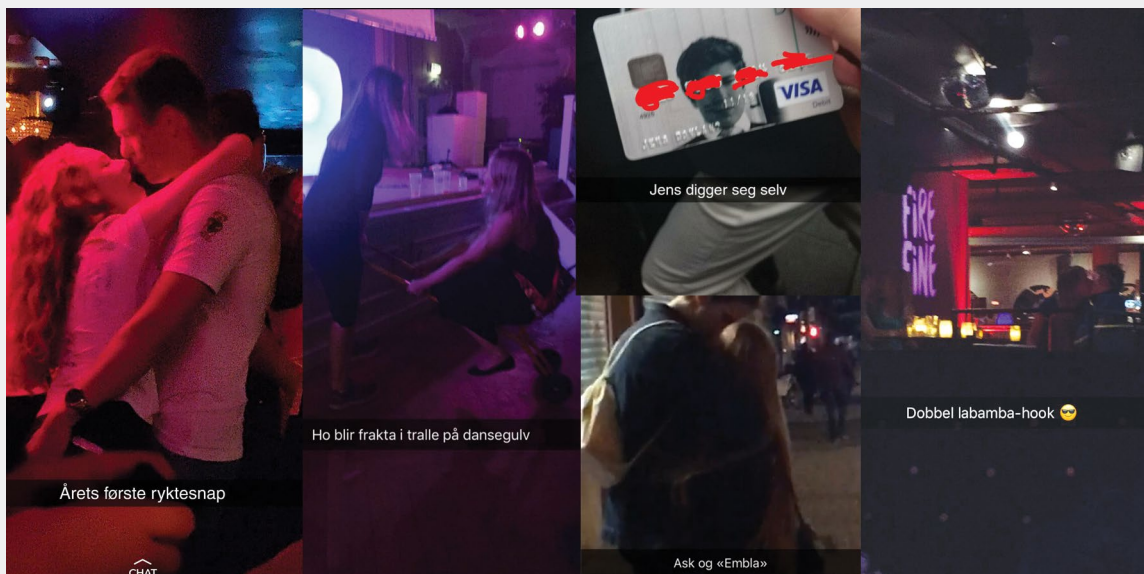
Under Abakus sitt immatrikuleringsball var det en herremann fra Hybrida og en kvinne fra Janus som fant tonen under hovedretten. Fra det **readme** har hørt viste ikke paret mye interesse for maten som ble servert, men heller for å spise hverandre. Kvinnen forsøkte å dekke til den intense spyttvekslingen med en hatt, men til liten nytte. Andre kjærtegn ble også observert, som for eksempel en hånd på vei oppover kilten til herremannen. Dette ble litt for meget for de i umiddelbar nærhet, og vår kjære PR-leder William endte med å komme til unnsetning. Han hadde nemlig vært på kiosken for å kjøpe en pakke kondomer til de nyforelskede. Kondompakken ble lagt på bordet, sammen med en oppfordring om at paret skulle fortsette seansen et annet sted. Noe mer om hva som skjedde med turtelduene vites ikke av **readme**, men vi håper selvfølgelig de hadde en fin kveld videre.

## Hookathon

Et fadderbarn ble under årets fadderperiode observert å hooke med 42 andre individer på Samfundet på kun 3 timer og 4 minutter. Det ryktes at dette var hele to minutter kortere tid enn personens tidligere rekord. **readme** gratulerer!

## Magasindødsfall

Tre **readme**-ere har omkommet under forsøket på å skaffe nok sladder til Smått&nett. Ettersom det ikke sendes inn noe sladder, må den mest sosialt tilbakestående komiteen finne sladder – et farlig og smertefullt oppdrag. **readme** sender sine kondolanser til de avdødes pårørende.



Send tips til  
[tips@abakus.no](mailto:tips@abakus.no)

Tidligere utgaver:  
[readme.abakus.no](http://readme.abakus.no)

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

