



Sofiero, Sofiåre

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

18. årgang nr. 1, 2016

Redaksjonen

Redaktør
Stein-Otto Svorstøl

Økonomi
Sandra Skarshaug

Layoutansvarlig
Patrick Skjennum

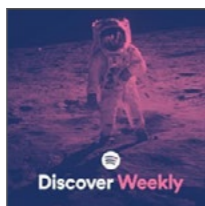
Medvirkende
Odd Magnus Trondrud
Hans Henrik Grønsløth
Ragnhild Cecilie Neset
Karoline Velsvik Berge
Stian Steinbakken
Ingvild Oxaas Wie
Sigurd Berglann
Mats Byrkjeland
Synnøve Halle
Eivind Kløvjan
Signe Carlsen
Marie Svanes
Bjørn Iversen
Fredrik Gram
Ask Jentoft

Kontakt

E-post
readme@abakus.no

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

Nettutgave
readme.abakus.no



Spotify ser deg ...
Men det er kanskje greit?

6



Folk er syke
Og det er din feil

12



Funksjonell salsa
Alle gode ting er «immutable»

18

Leder

Redaktør: Stein-Otto Svorstøl

Gleden over vårstemningen og det nye året kan bli kortvarig: LaBamba er, sammen med alle de andre kjellerne på Moholt, stengt av SiT som følge av mangler ved brannsikringen. Mens vi venter på rapporten som beskriver manglene og hvordan vi kan utbedre disse, er det lett å leke med ideer. Abakus kan jo leie et hus frem til LaBamba er klar igjen?

Det finnes mange grunner til å leie et hus. Det gir mye plass, noe Abakus kunne hatt nytte av. Foreningens interessegruppeordning blomstrer, og stadig flere søker penger til innkjøp av utstyr. Dette må lagres et sted. Med et hus ville det vært mer enn nok lagringsplass. I tillegg må komitéene stadig reservere rom rundt om på Gløshaugen – **readme** kunne gjerne laget neste utgave i et LaBamba-hus.

Slik **readme** ser det, vil det være gunstig å leie et hus. Sett fra et økonomisk perspektiv, virker det heller som et drapsprosjekt: Tiden for leieforholdet vil være uvisst, og man vil bruke fondpenger på noe som kun gagnar de som er med i Abakus akkurat nå. Hva hvis arbeidsmarkedet atter en gang knekker for oss – hvor skal abakulene feste da?

Vi trenger et forsamlingslokale. Spørsmålet er om vi bør vente på at SiT kutter oss ut for godt, før vi sikrer et slikt knutepunkt. Inntil videre jobber LaBamba aktivt med ulike løsninger – vi gleder oss.

Uenig? Vi ønsker å prøve ut en ordning med leserbrev. Har du noe på hjertet, send en tekst til redaktor@abakus.no. Kanskje kommer akkurat din tekst (maks 1000 tegn inkludert mellomrom) på trykk!



Drikk deg lykkelig

*Tekst: Signe Carlsen
Layout: Patrick Skjennum*

Lykkepromille er et prosjekt med mål om å redusere de negative konsekvensene av alkohol i studentmiljøene. Det er også et prosjekt som påstår at du har det morsomst på fest hvis du holder deg til to glass vin.

Sløv, men lykkelig?

Det er ingen tvil om at budskapet til Lykkepromille er positivt, og at saken som tas opp er svært aktuell. Prosjektet handler ikke om å få studenter til å avstå fra alkohol. Lykkepromille vet godt at åtte av ti studenter synes det er morsommere på fest med alkohol enn uten, og at sju av ti synes det er lettere å snakke med fremmede etter de har drukket. Det de ønsker er derimot å opplyse om hva slags alkoholforbruk som gir flest mulig positive, og færrest mulig negative konsekvenser.

Oppskriften på den optimale promille for en jente er ifølge Lykkepromille å drikke to enheter den første timen, så én enhet annenhver time. Da skal du ende opp med en promille som gjennom hele kvelden holder seg over en halv, men aldri overskrider én. Ifølge prosjektet er det i dette området alkoholen får deg til å skille ut lykkehormonet dopamin. Stiger promillen over én, klarer ikke hjernen lenger å utnytte seg av dopaminets effekt, samtidig som du i økende grad blir sløvere. Men fungerer Lykkepromilles anbefalinger i praksis?

Alkohol som frikort

Det er tydelig at prosjektet ikke blir tatt seriøst av studentene. Betyr dette at vi ikke ønsker å ha det morsomst mulig på fest? Svaret kan ligge i hva en vanlig student egentlig prøver å oppnå med alkohol. Skal du ut, er det ikke usannsynlig at du ønsker mer enn å bare komme i godt humør. Du

higer muligens etter å oppleve litt av alkoholens såkalte «negative» effekter. Si ting som ikke gir mening fordi du har lyst. Danse så stygt at det gjør vondt å se på, uten å bry deg. Bli kastet ut av alle utestedene i Trondheim på samme kveld fordi du hadde det sinnsykt gøy. Dumme deg ut, men le av det i etterkant.

Det kan tenkes at hovedårsaken til at Lykkepromille ikke har slått an blant flertallet av studentene, er at prosjektet har misforstått alkoholens ønskede rolle i festlivet. De fleste synes det er greit å gjøre litt teite ting i fylla. Selv om det er flaut, er det likevel utrolig morsomt. Lykkepromille prøver å ta fra studentene alkoholens rolle som «frikort» – et frikort jeg tror de aller fleste ønsker å beholde. Med en promille under én har ikke fylla skylda. Dette kan vi ikke ha noe av.

Et oppgjør med grammatikk-nazistene

Tekst: Eivind Kløvjan
Illustrasjon: Synnøve Halle

«Og/å», «your/you're», «its/it's», «then/than»: Hva har disse typiske skrivefeilene til felles, og hva har det hele med programmeringsspråk å gjøre?

Hvorfor blir vi så irriterte over skrivefeil? Det synes jeg er litt underlig. Det oppleves ofte som en slags følge av teknologi. Overalt i kommentarfelt på YouTube, Facebook og Twitter finner man en endeløs strøm av banale skrivefeil man ikke skulle tro at oppegående mennesker skulle kunne begå. Og det er vel stort sett de banale feilene vi irriterer oss mest over.

Da jeg spurte redaksjonen i **readme** om hvilke typiske skrivefeil på norsk eller engelsk de opplevde som spesielt irriterende, hadde jeg en hypotese: At mesteparten av feilene de nevnte ville være av en bestemt type, nemlig noe kalt

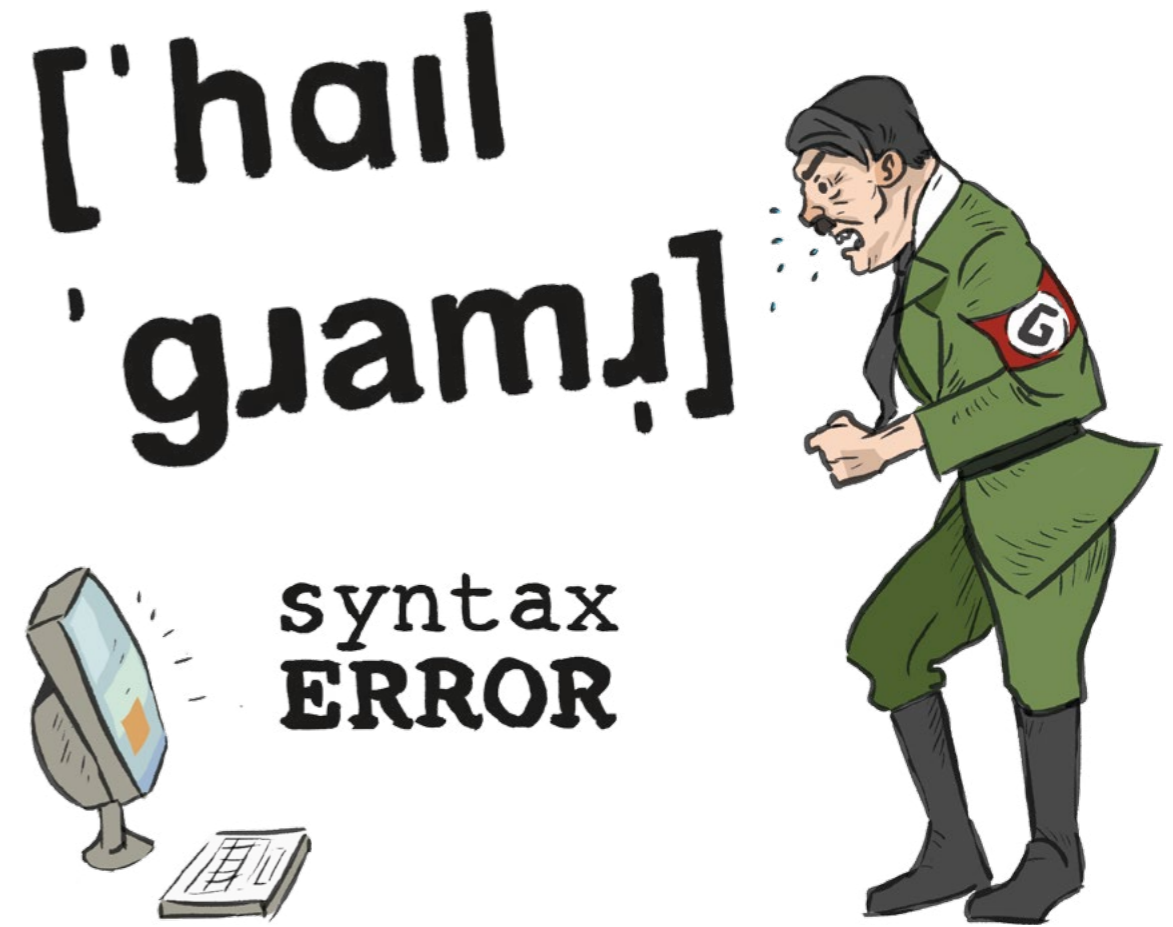
«homophone substitution error». Kort fortalt er det en type leksikalsk feil, der et ord erstattes av et annet med lik eller nesten lik uttale – et såkalt «homofon». Eksempler på slike leste du i ingressen, og sant nok: Det var nesten utelukkende slike feil jeg fikk til svar. Jeg kan fortelle deg med en gang at dette er en særlig utbredt type feil, som begås av morsmålstalere så vel som fremmedspråklige. Det hele munner ut i ett spesielt fenomen som gjelder for alle naturlige språk: De er i prinsippet muntlig.

Når lingvister studerer språk – det være som et generelt fenomen eller ett spesifikt språk – er det stort sett det *muntlige* som havner under lupen. Dette er rett og slett fordi det er det språk er: Det er muntlig språk mennesket lærer seg som barn, og som det har en medfødt evne til å tilegne seg. Skriftspråk som er avledet av disse naturlige, muntlige språkene er simpelthen forsøk på å kode

lydene i én bestemt dialekt til skriftegn. Målet er at man skal kunne utlede uttalen av et ord fra skriftegnene.

«Hapi berth dey two ewe»

Dessverre fungerer dette ofte dårlig. Uttalereglene til de fleste skriftspråk er gjerne grusomt uregelmessige. Engelsk er spesielt utsatt for dette fordi det i praksis bare er en sammenslåing av et utall ulike språk: Ord som egentlig bør rime, rimer ikke i det hele tatt («few», «sew»); ord som skrives på akkurat samme måte, uttales vidt forskjellig («attribute», «attribute»). Videre har vi at muntlige språk, altså dialekter, gjennomgår endringer lenge før det standardiserte skriftspråket – hvis skriftspråket i det hele tatt rekker å endre seg. Selv om et uttrykk ikke følger de standardiserte lovene i *skriftspråket*, betyr ikke det at det ikke er å regne som en del av språket i sin helhet.



Typen feil beskrevet tidligere er spesielt utbredt fordi denne typen feil *bare* kan oppstå i skriftlig språk. Det er også hovedårsaken til at man gjerne finner slike brølere omtrent like ofte hos mennesker som har språket som sitt morsmål, som hos de som har lært seg det i senere alder. Det at du har *snakket* et språk så å si hele ditt liv har lite å si for hvor godt du skriver det. I tillegg legges det ofte mye fokus på skriftspråk når man lærer seg et nytt språk.

«Syntax nazi»

Merk at når jeg bruker uttrykket «naturlig språk» i denne teksten, er ikke dette noen hippieoppfordring: Det er rett og slett et teknisk uttrykk som beskriver språk som oppstår gjennom bruk og repetisjon, framfor aktiv, planlagt konstruksjon og sammensetning. Motsetningen til naturlige språk er konstruerte språk, og herunder finner

vi en klasse språk vi alle er godt kjent med: programmeringsspråk. Visste du forresten at et annet ord for grammatikk er «syntaks»?

Jeg har ingen vendetta mot skriftspråk: Disse har tross alt sine finurlige kvaliteter. *Syntaktisk* har for eksempel «og» og «å» vidt forskjellige funksjoner. Men denne grammatiske forskjellen er ikke reflektert i det muntlige språket – den tar språkbrukere på intuisjonen, basert på hvilken språklig kontekst ordet er plassert i. Programmeringsspråk er nødvendigvis skriftlige, fordi denne syntaktiske informasjonen må være eksplisitt tilgjengelig. I alle fall er det slik det har vært til nå. Fremskritt innen kunstig intelligens, via såkalt «Natural Language Processing», har ført til at vi i dag er mye nærmere enn vi noen gang har vært til å programmere datamaskiner med det enkleste programmeringsspråket av alle: naturlig språk.

Blir du blenda nå?

Så hvorfor blir vi så irriterte når folk skriver feil? Det spørsmålet har jeg dessverre ikke noe konkret svar på. Kanskje vi lar oss irritere fordi noen av de mest banale, underliggende syntaktiske reglene i språket ikke opprettholdes, og vi er hemningsløse perfeksjonister. Vi klarer kanskje ikke å akseptere at andre ikke tar disse reglene intuitivt. Men det er lurt å tenke over at slike «feil» ofte vil være helt akseptable i muntlig språk. Og dessuten: Når den *bevisste* kunnskapen om språk hos folk flest er så lav som den er, har vi da rett til å irritere oss over at andre ikke tar ting på intuisjonen? For denne typen feil som de selvutnevnte «grammar nazi»-ene elsker å påpeke, er faktisk ikke å regne som «grammatiske» feil i det hele tatt, men *leksikalske*. Så det, så.

Ukentlige oppdagelser

I mylderet av mer eller mindre vellykkede anbefalingssystemer har Spotifys «Discover Weekly» gjort noe riktig. Lyst til å vite hvordan det fungerer?

*Tekst: Ragnhild Cecilie Neset
Layout: Karoline Velsvik Berge*



Valgene kveler

Før Internett fantes, ble valgene dine stort sett tatt for deg. Din lokale platesjappe, kosmetikkforretning og bokhandler hadde egne ansatte som jobbet med å plukke ut hvilke produkter som fikk æren av å utnytte de få tilgjengelige hyllemeterne. Forbrukerne hadde en håndfull TV-kanaler, aviser og radiokanaler å velge mellom, og svært liten påvirkningskraft på innholdet. Når handel og underholdning i stedet gjøres via nettet, økes utvalget i det uendelige. I teorien åpner man både for et bredere og bedre utvalg, der du kan finne vidunderprodukter som er helt perfekte for deg. Men hva er vitsen dersom valgmulighetene blir så store at du ikke klarer å finne disse produktene?

Anbefalingssystemer forsøker å forutse hvor godt en bruker kommer til å like et produkt, og har blitt en stor del av markedsstrategien til en rekke bedrifter. Generelt deles

anbefalingssystemer inn i to kategorier: *kollaborative* og *innholdsbaserte*. Det viser seg at en smart kombinasjon av disse to, og gjerne ispedd litt maskinlæring, gir de beste resultatene. Slik er både Netflix sitt anbefalingssystem og Spotify sitt vidunderbarn, «Discover Weekly», bygget opp.

En deilig musikk-sandwich

Kollaborativ filtrering bruker data fra andre brukere til å bestemme hva du kommer til å like. Ved *eksplisitt datainnsamling* blir brukerne bedt om å gi sin mening, for eksempel ved å rangere sanger eller filmer. I *Utgavens Algoritme* på side 24 er det beskrevet en metode for å bruke rangeringsdata til brukeranbefalinger.

I motsetning til hva en skulle tro er det *ikke* antall avspillinger som er utgangspunktet for «Discover Weekly» sine kollaborative data, men funksjonen alle benytter seg av – spillelistene.

Hver forekomst av en sang i en spilleliste telles som en «stemme» på den sangen, og spillelister som har mange følgere eller som er satt sammen av Spotify-teamet blir vektlagt ekstra. Videre prøver din «Discover Weekly» å finne fyllet i en sandwich bestående av sanger du liker godt. Det vil si at hvis to av dine favoritter ofte er å finne i nærheten av hverandre i andres spillelister, og det finnes en tredje, og for deg ukjent, sang som dukker opp sammen med disse to, regnes det som sannsynlig at du både kommer til å like denne sangen og synes at den passer sammen med de to andre.

Kollaborativ filtrering er bygget på en ganske enkel modell, men har et par ulemper. En får gjerne mye mer data om de artistene som allerede er populære, mens noe av poenget med anbefalingssystemer er å servere fersk og ukjent musikk til akkurat de som kommer til å digge den.

«Deep chamber pop»?

For å gjøre opp for ulempene ved kollaborativ filtrering introduserer vi innholdsbasert filtrering, som kombinerer beskrivelser av hvert enkelt element med data relatert til hver enkelt bruker. En nyhetsartikkel kan for eksempel beskrives med forfatter, lengde, overordnede kategorier og mange «tags» som dekker forskjellige temaer og egennavn. En ulempe med innholdsbasert filtrering er at noen elementer har et stort *semantisk gap*. De er rett og slett vanskelige å beskrive. Musikk faller definitivt i denne kategorien. Hvor mange ganger har du ikke besvart spørsmål om hvilke musikkjangere du liker med «eh... litt sånn forskjellig»?

Strømmetjenesten Pandora har ansatt en rekke analytikere som sitter dag ut og dag inn og lytter til nye sanger og forsøker å beskrive dem best mulig. Men dette er en dyr og begrenset løsning. I en musikkverden der det virker som de aller

fleste adjektiver kan kombineres vilkårlig for å lage nye mikrosjangere, har Spotify vært smart nok til å utnytte de eneste som faktisk forstår seg på og bryr seg om dette: Musikkbloggere. Spotify har kjøpt opp «The Echo Nest», et program som skanner mengdene av anmeldelser og musikkblogger på nettet, og som blant annet bruker «Natural Language Processing» for å finne stemninger, kategorisering og andre beskrivelser av nyttingselsene.

Spotify ser deg

Innholdsdata kombineres med lyttemønstre for hver bruker og produserer en stadig oppdaterende «profil», der sjangre og artister har en poengsum som representerer hvor godt du liker dem. Lignende sjangre er gruppert sammen, slik at raringene – sangene du har lyttet på, men som ikke passer inn med resten av profilen din – står på ensomme øyer og ikke regnes som ekte medlemmer av din

musikksmak. Jeg er evig takknemlig for at vorset sist helg ikke fyller min «Discover Weekly» med Broiler og Katastrofe.

I korte trekk tar «Discover Weekly»-algoritmen utgangspunkt i låtforslag tatt fra de kollaborative spillelistedataene, sjekker at de ikke krasjer med smaksprofilen din og smekker dette inn i en personlig spilleliste. Den forsøker å registrere hva som var dårlige eller gode valg ved å tolke spillemønsteret ditt, som hvilke sanger som ble lagt til i andre spillelister, hvilke artister du sjekket ut profilen til og hvilke sanger du hoppet over. Musikk lyttet til inne i «Discover Weekly»-lista er ellers ekskludert fra din smaksprofil, for å unngå å styre preferansene dine. Spotify har fått mye skryt for å klare å lage en algoritme som kommer med originale og spennende forslag, men mener selv at det aller viktigste elementet i algoritmen er de menneskelagde spillelistene.



Gløsløken

Det du leser nå

Dragvoll blir asylmottak

Tekst: Mats Byrkjeland

NTNU – Regjeringen har vedtatt at etter at NTNU campus Dragvoll slås sammen med Gløshaugen vil lokalene bli brukt som asylmottak for å håndtere den store mengden syriske flyktninger som strømmer til landet. Statsminister Solberg (H) lover at hver flyktning som bosettes ved Dragvoll asylmottak vil få tildelt en egen pult å sove på, frokostabonnement i SiT-kantinen, samt blomsterkompendiet til *Examen Philosophicum* på statens bekostning. Per Sandberg (FrP) misliker sterkt dette vedtaket: «Frokostabonnement er ikke billig. Når ikke alle dagens studenter ved NTNU har råd til frokost, er det noe forbanna tull å gi det gratis til hundrevis av innvandrere». Sandberg får støtte av finansminister Siv Jensen (FrP): «I stedet for å spandere rundt to hundre blomsterhefter på flyktninger i Norge, kunne vi sendt hundre tusen hefter der de kommer fra, og dermed hjelpe dem med exphil der de er.»

Facebook-status blir delt mange ganger

Tekst: Ragnhild Cecilie Neset

NORGE – «UTROLIG vakkert» og «dette er bare så sant, fine deg». Slik lyder kommentarene under innlegget som Johanna Hansen skrev søndag formiddag. Nå, bare ett døgn etter, er innlegget blitt delt over 100 000 ganger, fått nesten 300 000 «likes» og 150 000 kommentarer. Flere kjendiser har både likt og kommentert innlegget, deriblant blogger Såfint Elise Eliziana og programleder Stian Flopp. Begge er å finne i **readme** sitt spesiallagde debattprogram om den norgesvirale Facebook-statusen. Ekspert i sosiale medier, Lisa Like, har jobbet på sprenge med å analysere innlegget helt siden det dukket opp i hennes egen Facebook-feed søndag kveld: «Jeg tror nok dette er et innlegg som rører det norske folk på en helt spesiell måte. Ifølge mine beregninger kan vi i løpet av de neste to døgnene forvente en popularitetsøkning på 523,4 prosent på LCS-skalaen («Like, Comment, Share»-skalaen, journ. anm.).»

Antivirusmotstand: en ny, farlig trend

Tekst: Mats Byrkjeland

NORGE – En bølge av antivirusskepsis ser ut til å flomme over landet. Flere og flere velger å la være å installere virusbeskyttelse på sine digitale enheter. BI-student og antivirusmotstander Hermann Naivenlund forteller: «Antivirusprogrammer forpester datamaskinen. De maser om at filene mine er usikre, de ber om oppdateringer hele tiden, og scanning tar altfor lang tid.» Naivenlund forteller også at han har deaktivert oppdateringer fra Windows. «Jeg foretrekker min Vista ren, som om den kom rett av DVD-en.» Ekspert i programvaresikkerhet ved IDI, Dr. Häcker Meinfreihm, oppfordrer folk til å holde maskinene oppdaterte: «Gjennom beskyttelsesprogramvare har vi nesten utryddet ormer og virus som *Code Red*, *ILOVEYOU* og *Melissa*. Brannmur er også en viktig mekanisme for å holde maskinen sin trygg». På spørsmål angående brannmur svarer Naivenlund: «Brannmur? Du mener mykvarens svar på Berlinmuren? Nei takk.»



Tekst: Jean Niklas L'orange (M.Sc. 2014), programvareutvikler i Rendra
Layout: Patrick Skjennum

Jeg er glad i datastrukturer, så da finnes det vel ikke noe bedre å bruke masteroppgaven sin på enn akkurat det! Oppgaven min dreide seg om å ta en relativt ny datastruktur – RRB-trær – og implementere et par optimaliseringer fra en lignende datastruktur.

Det å finne en masteroppgave er ikke alltid lett, og om du er glad i algoritmer og datastrukturer er det kanskje enda vanskeligere. Forskere er hele tiden på ballen og gira på å gjøre de eksisterende algoritmene og datastrukturene mer effektive. Ofte betyr det at du må sammenligne én eller flere algoritmer eller datastrukturer på en spesifikk arbeidsoppgave, fordi alle de «nye» tingene allerede har blitt designet.

Det er mulig å komme utenom dette, men det er ikke alltid like lett. Enten må du finne en obskur datastruktur fra 70-tallet, eller en relativt ny en, og ofte viderearbeide denne fra «Further Work»-seksjonen i artikkelen som beskriver den. Problemet er at det er veldig få

slike datastrukturer fra 70-tallet som er verdt å ta en titt på. De nye datastrukturene er på sin side ofte veldig avanserte og krever mye forståelse som ikke kan læres fra pensumet i «AlgDat» og «AlgKons».

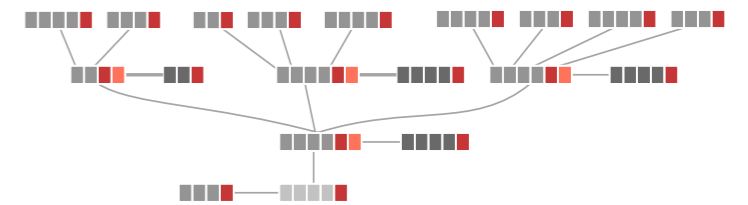
Korte og uforståelige artikler

Jeg endte opp med å se på en av de nyere datastrukturene der ute, RRB-trær. Det er en datastruktur fra 2011 som bygger på en av de viktigere datastrukturene i programmeringsspråket Clojure: *Persistente vektorer*. Tanken min var å implementere tre forbedringer man har funnet på persistente vektorer som i teorien kunne implementeres på RRB-trær.

Dette høres kanskje greit ut, men det å prøve å forstå en ny avansert datastruktur som kun har en artikkel på seks sider er mye vanskeligere enn du kan forestille deg. For å gjøre ting ekstra utfordrende, hadde forfatteren av artikkelen dødd ett år tidligere. Mye av tiden

Utgavens master

RRB-trær



gikk dermed med på å forstå hvorfor i alle dager dette var korrekt, og å skrive ned bevisene for det i masteroppgaven. Deretter gjaldt det å implementere datastrukturen med de forskjellige ytelsesforbedringene.

Resultater, resultater

Etter å ha stanget hodet i veggen i et par måneder – noe man visstnok pleier å gjøre i en masteroppgave – kom endelig resultatene. Dessverre var enkelte av dem litt suspekter. Etter et par dager med undersøkelser, fant jeg ut at årsaken var *garbage collectoren* som i visse tester gjorde kjøretiden mye dårligere. Da det var fikset, ble resultatene forståelige og det viste seg at forbedringene gjorde datastrukturen langt mer effektiv!

Selv om du finner en interessant masteroppgave, kommer du nok til å føle deg litt hjelpeløs de første ukene og månedene. Heldigvis løsner ting etter hvert, ting blir morsomme og du lærer utrolig mye.

Årets høydepunkt

Tekst: Marie Svanes
Layout: Sandra Skarshaug



Tradisjonen tro var det en del trøtte tryner som hadde møtt opp utenfor Hovedbygningen på Gløshaugen en tidlig torsdags morgen for å delta på «vinterens vakreste eventyr»: Abakus i Åre.

Vi i første klasse var uvitende om hva som ventet oss, men vi ble lovet en heidundranes tur med god stemning og allsang allerede på bussturen. Med «Nu kör vi» som turens motto skulle det bare mangle at det var «Fanta & Rosé» ispedd klassikere av Broiler og D.D.E. som ljommet gjentatte ganger på bussen. Til alles store glede uteble allsangen, og lydnivået var heldigvis slik at det var mulig å ta seg en lur for å forberede seg på det som ventet.

Vel fremme fikk vi beskjed om at ingen kunne sjekke inn før klokken 15:00, noe som ga oss

rikelig med tid til å utforske alt det spennende Åre har å by på. Dette innebar at vi gikk for å sjekke ut hvilke leskende drikker Systembolaget hadde på menyen. Vi benyttet også anledningen til å handle inn store mengder chips, blandevann og polarbrød. Du kan si det slik at vi ikke skal spise polarbrød på en stund nå.

Varierende forhold

Boforholdene meldtes om å være varierte. Det var en del som manglet noe så elementært som en stekeovn. Hvordan de overlevde hele turen uten frossenpizza vites ikke. En del klaget også over færre sengeplasser enn oppgitt, men å få «spooner» litt sammen med vennene sine må jo bare bli sett på som en bonus. Det var nok uansett en del som ikke hadde så stort behov for sengeplassen sin, da nattet ofte varte ut i de tidlige morgentimene.

De som dro i slalåmbakken med en gang på torsdagen hadde nok vært i Åre før. Det viste seg nemlig at det skulle bli verre og verre for hver dag å klare den bragden det er å stå opp tidsnok, og i tillegg ikke gå rundt med et konstant behov for å spy. Likevel ble det noen veldig fine dager i bakken, og spesielt på fredagen var forholdene fantastiske. Jeg velger å stille meg uforstående til de som ikke utnyttet bakkene i det hele tatt, da Åre faktisk er et av Nord-Europas beste skisteder. Tenk at folk faktisk ble med til Åre bare for å drikke. Uhørt!

«Nu kör vi»

Kvelden kom og mange hadde allerede konsumert nok alkohol til middagen slik at vorset var godt i gang allerede klokken 17. Mer alkohol ble inntatt og «Fanta & Rosé» ble igjen

spilt på «repeat» til alles store glede. Festen ble tatt videre til det sagnomsuste Bygget, der musikken var høy og stemningen på topp. Heldigvis var det ikke en *abakule* som endte kveldens festligheter ved å bli lagt i bakken av fire svære vakter og sendt videre med politiet.

I år klarte faktisk Arrkom å arrangere Vinterlekene som de hadde skrytt så mye av. Lovnader om mye gratis sprit og «masse moro» lokket mange håpefulle abakuler til å starte dagen på best mulig måte. Stemningen var høy og konkurranseinstinktet muligens enda høyere. Aktiviteter innebar blant annet shottestafett og drinkmiksing, og selv om det til slutt bare sto igjen én vinnergruppe, følte nok alle seg litt som vinnere siden de tross alt hadde fått gratis drikke og startet vorset allerede.

Fredag: Ny dag og nye muligheter til å drikke seg enda fullere enn foregående dag. Noen var fortsatt fulle etter torsdagen, men så bare på det som en mulighet til å fortsette kalaset før de merket bakrusen. Den ærede ledertrøya ble også delt ut til den som var fullest, og det virket som det til slutt ble om å gjøre å ende opp med den, til tross for at den neongule fargen falmet litt hen av en blanding av spy og Jägermeister. Hvorfor noen velger å la sin egen kropp bryte seg selv ned av alkohol på denne måten er heller usikkert.

Herlig søndagsmorgen

Siste dagen gjensto, og folk tenkte like greit at man bare burde ta den helt ut. Verandan var atter en gang stedet hvor festen ble tatt videre, og ut i fra barens skyhøye priser og promillenivået som lå i luften, vil jeg anta at

stemningen var på topp denne kvelden også. Når det i tillegg gikk an å tape alle pengene sine på blackjack, var det ingen som kunne klage. Til tross for utsjekk halv ti dagen derpå, satset mange på at det gikk fint å ha nach til morgengry. Dette gikk kanskje fint for de fleste, men det var nok en del som skulle ønske de hadde lagt seg litt tidligere dagen før, eller bare ikke nattet til frokost for så å miste bussen.

Til tross for at vi glemte igjen en mann i Åre vil jeg tørre å påstå at det var en meget vellykket tur for alle som deltok. Kroppen og lommeboka takker og bukker for at festlighetene er over for en stund. Likevel kan vi vel si at vi gleder oss til neste gang «Nu kör vi» nok en gang runger over høytaleranlegget på bussen.

Mikroskopisk krigføring

Tekst: Bjørn Iversen
Illustrasjon: Ask Jentoft

Antibiotika. 1900-tallets mirakelmedisin. Antibiotikumets mange typer har reddet millioner av menneskeliv, men nå virker det som at dens slagkraft ebber ut. Hva er årsaken til denne endringen?

I 1928 oppdaget den skotske biologen Sir Alexander Fleming *penicillin*. Denne oppdagelsen skulle vise seg å revolusjonere datidens medisin. Sykdommer som før var dødsdommer, som tuberkulose og lungebetennelse, fikk straks en langt lavere letalitet. Gjennom historien har mange andre typer antibiotika blitt oppdaget og fremstilt, som *ampicillin*, *amoxicillin* og *benzylpenicillin*. Massiv bruk av antibiotika har dessverre ført til økt resistans hos mikroorganismene. Vi ser ut til å gå inn i en tidsalder der vi ikke lenger har våpen for å bekjempe bakteriene.

En kjapp innføring i mikrobiologi
Det er ingen dum idé å ha litt kunnskap om bakterier, antibiotika og virus når man snakker

om antibiotikaresistens, så her er et lynkurs i mikrobiologi: De aller fleste bakterier kan plasseres i en av to grupper: *Gram-negative* eller *Gram-positive*. Disse inndelingene er basert på hvordan celleveggen til bakterien oppfører seg. Gram-positive bakterier – som *streptokokker* – har tynne, lett gjennomtrengelige cellevegger med ett enkelt lag. Gram-negative bakterier, derimot – som *E. coli* – har tykke, mindre gjennomtrengelige cellevegger med to lag. For at et antibiotikum skal ha noen effekt mot en gitt type bakterie, må det kunne trenge gjennom celleveggen.

Bakterier er altså celler. Men hva er egentlig et virus? Majoriteten av de lærde karakteriserer virus som ikke-levende, siden de er avhengige av andre former for liv for å overleve. Et virus er en slags «masse», med en bunt DNA, og dets nærmeste slektning, *RNA*, som i bunn og grunn er DNA, bare med oksygen. Når et virus inntar kroppen, har det et ønske om å infiltrere en celle. Om ønsket

oppfylles, vil viruset bruke cellens maskineri, og noen av enzymene i cellen, til å lage mange nye, modne og smittsomme virus, som kan gjenta prosessen. Virusenes fremtredelse foregår altså noe forskjellig fra de kanskje mer folkekjære bakteriene.

Hvorfor blir bakterier resistente?

Antibiotika tar livet av en bakterie på én av følgende måter: Enten ved å hindre bakteriens evne til å reparere sitt skadede DNA, ved å stoppe dens evner til å lage det den trenger for å skape nye celler, eller ved å svekke dens cellevegger til den faktisk sprekker. Hvilken av de tre måtene det gjøres på, avhenger av hvilken type antibiotika og hvilken bakterie det er snakk om. I motsetning til bakterier har ikke virus noe stoffskifte. Et virus må derfor, som nevnt tidligere, infiltrere en celle før det kan begynne å formere seg. Om man da skal prøve å fjerne viruset med medisiner eller andre «giftstoffer»,



kan man ikke gjøre det uten å angripe cellene viruset har infiltrert. Det er derfor antibiotika ikke virker mot virus.

Nå vet vi hvordan antibiotika fungerer, men hvorfor blir bakteriene resistente? Resistens kan i grove trekk oppnås på to ulike måter: Mikroorganismene kan være utrustet med forsvarsmekanismer mot midlene fra naturens side, eller så kan de oppnå resistens via naturlig seleksjon. Bakterier lever under et sterkt seleksjonspress. Unyttige egenskaper velges dermed bort for å overleve. Naturlig resistens har altså liten utbredelse i antibiotikafrie miljøer.

Problemene oppstår i økosystemer der bakteriene utsettes for enorme mengder antibakterielle midler. I disse miljøene overlever kun bakteriene med resistens, og disse kan da formere seg fritt. Det er med andre ord en åpenbar sammenheng mellom bruk av

antibiotika og resistens. En svært dum idé er å ikke fullføre en antibiotikakur, siden enkelte bakterier kan overleve, og erverve seg resistens. Det vil vel de fleste unngå.

Er vi alle dødsdømte?

Dersom skrekksenarioet skulle realiseres – at vi ikke lenger har antibiotika som faktisk fungerer mot bakteriene – vil moderne medisin gå under. Det er en rekke inngrep vi ikke vil kunne utføre, fordi risikoen for å utvikle alvorlige infeksjonskomplikasjoner er for høy til at inngrepene er forsvarlige å gjennomføre. Sykdommer som i dag blir betraktet som harmløse, for eksempel syfilis eller urinveisinfeksjon, vil igjen begynne å kreve mange menneskeliv.

Kan vi ikke bare lage ny antibiotika, da? Jo, det kan vi, men den farmasøytiske industrien tjener ikke penger på det. Da er det jo ikke noen vits.

Vi spiller tross alt om penger, ikke menneskeliv. Siden antibiotika mister effektiviteten sin ganske raskt, er det mer lukrativt å legge penger i forskning på medisiner som ikke mister virkningen sin så snart de kommer på markedet, som antidepressiva eller smertestillende.

Dersom vi ikke fremstiller ny antibiotika, hva blir løsningen? Det er i hvert fall ganske åpenbart at vi må bruke midlene langt mer forsiktig enn det vi har gjort til nå. Leger har til ganske nylig skrevet ut antibiotika som en slags mirakelkur for enhver sykdom og ethvert symptom. Men det er ikke kun helsesektoren som har ansvar. Det har du og jeg også. Vi må bruke de legemidlene vi har med respekt, og vi må følge retningslinjene som kommer med dem. Å følge retningslinjene betyr å ikke selvbehandle, ei heller å droppe en dag med medisin bare fordi du ikke gidder. Det er vår og våre barns fremtid det er snakk om. Ta ansvar, spre budskapet.

GENFORS 4MARS





Ting du bør vite som siving

Branching-strategier

Tekst: Stein-Otto Svorstøl

Versjonskontrollsystemet Git er det beste alternativet når man skal skrive kode sammen med andre. Men med tiden blir det tydelig at Git alene ikke sørger for godt samarbeid, så hvordan kan du bruke Git for å fremme det?

Git har noe som kalles en *branch* som kan sees på som et arbeidsområde. Endringer i din branch påvirker ingen andres endringer. Man kan flette sammen en branch med en annen, slik at endringer fra to slike arbeidsområder til sammen blir ett enkelt arbeidsområde. *Master*-branchen er den branchen man først får se når man kikker på et prosjekt på GitHub, og den man vanligvis gjør endringer på. Det er ugunstig at flere jobber samtidig på denne, ettersom det kan medføre rot og konflikter. Derfor bør man ha en strategi for hvordan man skal dele opp i nye brancher.

Leken feilsøking

En *branching-strategi* er en strategi for når man skal opprette en ny branch fra master, hva den skal inneholde og hva den skal kalles. Årsaken til at man vil ha en slik strategi, er at når man skal jobbe sammen kan det være vanskelig å

se sammenhengen mellom hva som ligger i ulike brancher. Om jeg har en branch som heter «branch» som fikser en feil, mens «branch2» introduserer en helt ny funksjonalitet, kan ikke du som utenforstående vite noe om forskjellen. Når det oppstår en ny feil, og du ser gjennom historien til prosjektet, vil det være umulig å se hvor feilen kom inn, basert på brancher. Det er mer sannsynlig at ny funksjonalitet introduserer nye feil, enn at feilrettinger introduserer feil.

Alt til sitt bruk

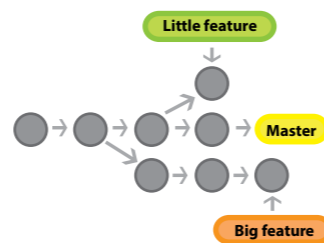
Det er tre vanlige strategier for branching, og hver av dem tar utgangspunkt i hver sin entitet: Programvareutgivelser, funksjonalitet og oppgaver. Dersom du tar utgangspunkt i programvareutgivelser, lager du en branch for hver versjon av programvaren. I dagens smidige hverdag er dette litt avleggs. De to andre strategiene tar utgangspunkt i henholdsvis funksjonalitet og deloppgave. Hver enkelt strategi har fordeler og ulemper, og *du* må finne frem til hva du føler passer best til ditt prosjekt. For eksempel er det lite passende å branche etter oppgaver, dersom hver oppgave blir så liten at

du bruker lenger tid på å skrive Git-kommandoer enn å skrive kode.

Navngivning

Når man først har bestemt seg for hva utgangspunktet for en ny branch skal være, bør man også bli enige om hva de skal kalles. Feilrettinger kan begynne med «bugfix/», og ny funksjonalitet med «feature/». Har man en *issue tracker* som den på GitHub, kan man også ha med nummeret på saken der. Det kan i tillegg være lurt å ha med en kort beskrivelse. Et eksempel kan være «bugfix/#58-email-validation».

En branching-strategi er nødvendig for å få mest mulig ut av Git, men sørg for at alle er med – det er ikke kult med en strategi ingen følger.



Ting du ikke bør vite som siving

Umami



Tekst: Signe Carlsen

Har du noen gang tenkt over hvorfor du synes kjøtt, tomat og parmesan smaker så godt? Ingen av delene er hverken særlig søtt, salt, surt eller bittert, men de er likevel smakfulle. Svaret på dette mysteriet ligger i den femte smakssans, umami.

Mer enn bare salt

Umami ble oppdaget av den japanske forskeren Kikunae Ikeda i 1909, og ordet kan oversettes til «velsmakende». Mens vi i Norge fortsatt ikke har noe godt adjektiv til å beskrive smaken, finner vi ordet «savory» på engelsk. Bano Singh, som har en doktorgrad i umami fra UiO, beskriver smaken som saltaktig, men at det smaker mer enn salt. Litt som buljong og soyasaus; det er salt, men gir også en svært tilfredsstillende smak som fyller hele munnen og varer lenge.



Sjøgress og grillkrydder

Så hva er det egentlig som gir denne smaken? Svaret er proteiner, nærmere bestemt aminosyren *glutamin*. Ren glutaminsyre alene gir ikke så mye utslag hos dine smaksreseptorer, men fremstiller du glutaminsyrens salter, glutamater, får du umamismaken i sin mest konsentrerte form. *Monosodiumglutamat* (MSG), er en svært flittig brukt smaksforsterker i potetgull og buljongterninger, samt krydderblandinger som grillkrydder og Gastromat. Flere forskere har påstått at bruk av kunstig MSG ikke er trygt, men det er fortsatt uenighet om dette er tilfellet.

Uansett trenger du ikke fortvile, for du finner haugevis av matvarer med et høyt innhold av naturlige umamikomponenter. Smaken ble først påvist i sjøgress, mens her i Norge får vi mye naturlig umami gjennom tomater, kjøtt, reker, sopp, modne oster og soyasaus. Trodde du at du likte ketchup på grunn av dens gode balanse mellom sødme, salt og syrlighet? Tro om igjen. Ketchup, med et høyt innhold av tomatpuré, er nemlig en umamibombe uten like.

Vegetarpropaganda

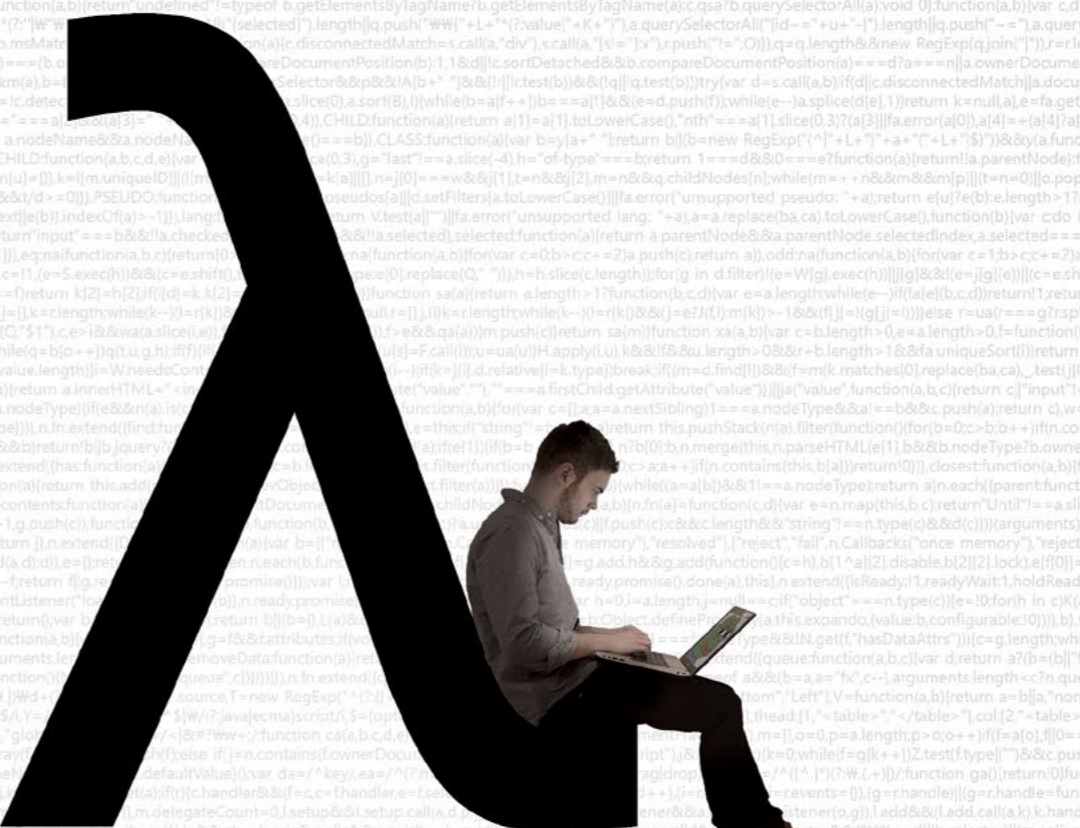
Til slutt lurer du kanskje på om alt dette pratet om umami kan ha noe godt for seg. Er dette kunnskap for en bedre verden? Svaret er ja. Ved å bli bevisst på å bringe umamismaken inn i vegetarretter gjennom buljong, soyasaus, tomat og sopp, kan kjøttforbruket reduseres uten at det føles ut som om noe mangler på smaksfronten. Informasjon om umami er noe som bør nå ut til alle som tror at vegetaromat er kjedelig.

Denne artikkelen har forhåpentligvis fått deg til å innse to viktige ting: For det første at umami er flott og godt, men aller viktigst, det finnes ikke et eneste tema som det ikke er mulig å gjøre om til vegetarpropaganda.



Funkky funksjonell

Tekst: Stian Steinbakken
Layout: Sigurd Berglann



Har du lyst å skrive mer kompakt og lettlest kode, som lett kan skaleres til et flere prosessorer, samtidig som du tenker nøyere over løsninger dine? Funksjonell programmering er tingen.

Hva er greia?

Funksjonell programmering er et programmeringsparadigme som bygger på evaluering av matematiske funksjoner. Hva betyr dette? Hvis $f(x) = y$, vil funksjonen alltid returnere y når den får inn x . Det assosieres én, og kun én statisk output for hvert input. Hva funksjonen returnerer er uavhengig av hvilket stadium programmet er i. I tillegg baserer den funksjonelle framgangsmåten seg på uforanderlig (*immutable*) data og unngåelse av tilstandsendringer. Dette gjør at funksjonelle programmer er svært oversiktlige og forutsigbare, noe som er en essensiell del av motivasjonen for å konstruere funksjonelle programmer.

Billige strenger

Du husker muligens at strenger i Python og Java er uforanderlige. Endrer du en streng så har du egentlig ikke endret den, du har brukt den, kastet den bort som den gamle kona du har gått lei av, og erstattet den med en ny og yngre streng. Funksjonelle programmer er fullstendig hensynsløse og gjør dette med all data. Hva er fordelen med det? Har du noen gang jobbet med to kopier av en liste, gjort endringer på den ene, for så å merke at alt ikke er som det skal være i den andre? Her ligger det referanser til et objekt flere steder. Hver du *multithreading* inn i blandingen har du en oppskrift på et evinnelig kaos. Alt dette slipper du i funksjonell programmering, fordi du ikke har variabler og objekter som endrer seg.

For å sette det hele i et praktisk perspektiv kan vi ta for oss en enkel for-løkke. I «tradisjonell» programmering ville vi laget en for-løkke med

en inkrementerende variabel, ofte i . Med en funksjonell framgangsmåte tar vi oss ikke denne friheten. En mulighet hadde vært å skrive oppgaven som skal utføres like mange ganger som løkken ville gjort det. Dette er funksjonell, men ikke veldig praktisk. La oss se hvordan vi kunne gjort det i Clojure:

(take 25 (squares-of (integers)))

Her blir i i ekte funksjonell stil – én funksjon kjørt 25 ganger, på 25 forskjellige konstanter. Dette helt uten hjelp av variabler, og koden er kort, elegant og konsis.

Den funksjonelle ideen er at hver funksjon tar inn minst ett argument, og returnerer noe. Du har nå en serie med uavhengige, selvstendige funksjoner. Videre kan du koble disse sammen i endene og få resultatene du er ute etter. Du kan se for deg et samleband. Samlebandet består av

en rekke maskiner som hver utfører sin oppgave. En maskin vet for eksempel at den tar inn øl, pakker dem i esker, og sender dem videre. Den har ingen formening om hvordan ølen ble laget, eller hva som skjer med eskene den sender ut. Den gjør bare sin jobb, fullstendig uvitende om omverdenen. Her er maskinen funksjonen vår, og den tar inn øl som argument og returnerer esker. Så lenge vi kobler denne maskinen opp med en annen maskin som gir den øl, og en som tar i mot esker, så er vi i gang!

Du har kanskje sett det før?

Funksjonell programmering har sine røtter i lambda-kalkyle, noe som muligens leder tankene dine mot lambdafunksjoner fra Python, Java eller et annet ikke-funksjonell programmeringsspråk. Og det er nettopp det lambdafunksjoner er! Flere programmeringsspråk støtter funksjonelle prinsipper. Du har muligens jobbet med *streams* i Java? I så fall har du drevet med funksjonell

programmering. Gratulerer. Som sagt er grunnideen at rekkefølgen et program utfoldes i, og historien til programmet, ikke har noen innvirkning på hvordan funksjonene kjører, eller hva de returnerer. I imperativ programmering, som er det de fleste av oss er vant til, har tilstanden mye å si for hvordan programmet oppfører seg. Hvis en global variabel har blitt endret mens programmet kjører, kan dette ha en stor innvirkning på resten av forløpet. Dette slipper vi unna med i funksjonell programmering – konseptet tilstand eksisterer rett og slett ikke.

Så hvorfor funksjonell programmering?

Bortsett fra å være et høyst praktisk tillegg til imperative språk, er funksjonelle programmeringsspråk per i dag på mange måter kun en stor hipster «circlejerk» for spesielt interesserte. Paradigmat har eksistert siden programmerings morgen, men det er på

vei inn i langt større grad. For eksempel har Twitter tatt i bruk Scala for sine mest krevende komponenter, og AT&T bruker Haskell i sine nettverkssikkerhetssystemer. Begge to er funksjonelle språk. Hovedgrunnen til at funksjonell programmering er spådd en lysende framtid, er vennligheten overfor multithreading. Slik som situasjonen er i dag begynner vi å bli et menneske som greener for hvor mange transistorer vi faktisk kan få på et mikroskopisk område. Dagens transistorer på størrelse med atomer vil være enden på Moores lov, og da er vår foreløpige løsning flere kjerner. Vi sliter allerede med å skrive programmer for fire kjerner, så hva skjer når parallelliteten øker til 8, 16 eller 512? Det er her funksjonell programmering blir en essensiell komponent. Funksjonelle programmer kan kjøres på uendelig mange kjerner samtidig, fordi funksjonene ikke er avhengige av tilstanden til programmet eller hverandre. I tillegg bringer det med seg fin og konsis kode, som er svært vedlikeholdsvennlig.

Den moderne verden



Tekst: Ingvild Oxaas Wie
Illustrasjon: Synnøve Halle

Måling av data fra hverdagen er en økende trend, men hvor langt kan det egentlig gå?

Data er gøy

I dagens moderne verden flyr teknologi rundt i hodene våre, for så å falle ned i våre hender. Utviklingen stopper aldri: Det er nesten alltid en ny dupperings under treet til jul. En ting som har blitt større de siste årene er såkalte *wearables*: Teknologi man kler på seg, og som «tracker» livet ditt. Hvor mange skritt har du gått i dag, hvor lenge sov du, og var du urolig store deler av natten? Man kan i tillegg legge inn informasjon selv, for eksempel hvor mye man spiste, hva man veier og mye mer. Her finnes det endeløse muligheter.

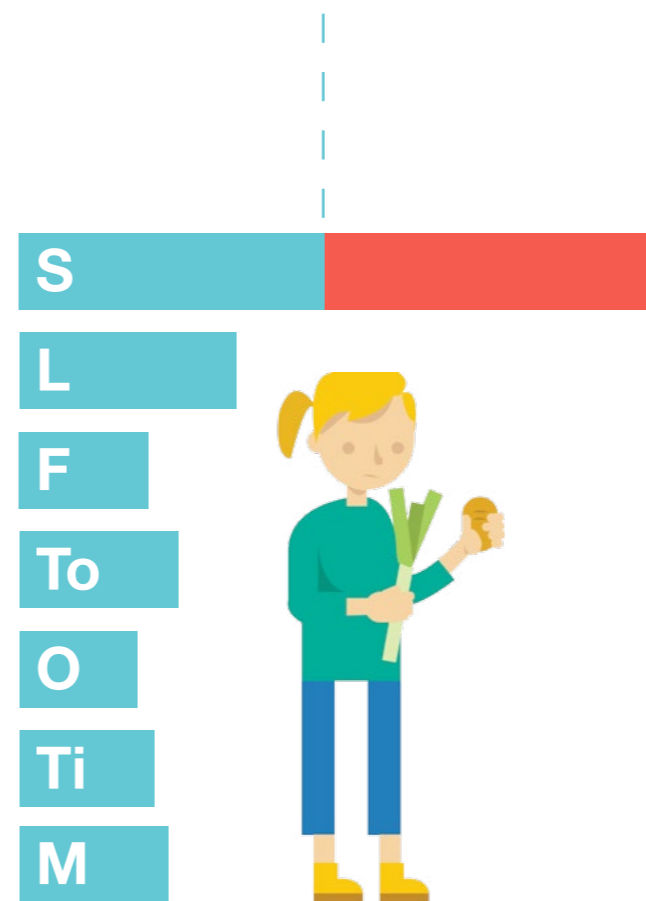
Alt dette gir deg masse data, og som datastudent synes du kanskje data er gøy.

En av de mest brukte wearablesene er Fitbit, et lite armbånd som automatisk sporer søvn, skritt og puls. Dersom du ser deg rundt i en forelesningssal finner du fort et par som har dette armbåndet på seg. Ved bruk av apper kan dataen fra armbåndet lagres, og du kan sette daglige mål som du ønsker å nå. En annen mulighet du finner i flere av disse appene er kalorimåling, hvor appene begynner å fortelle deg hvor mye du bør spise hver dag. Da kan du føre inn hva du har spist, slik at du kan se om ditt daglige kalorimål blir nådd. Om du er av den eksentriske typen kan du få hele livet ditt ned som data – dette kan jo være ganske praktisk. Du får en oversikt over hva som skjer i livet ditt og hva du kan gjøre for å snu uvanene.

Kontroll over livet

«Quantified self» er en bevegelse for de som måler livet sitt, både manuelt og med teknologi. Med teknologien stadig pustende i nakken, og mens prisene på dingsene blir lavere og lavere, er det flere som vil teste ut hva disse har å by på. Med «quantified self» kan du analysere livet i detalj: Du kan finne ut hvorfor du ikke klarer å sove før klokken 2 på natten – må den kaffekoppen klokken 17 byttes ut? Burde du ta deg en treningsøkt en time før leggetid, eller kanskje ikke? Når du sporer deg selv får du mye data, og denne dataen kan tolkes. Dersom du ikke har spist frokost på en hel uke, og vekten har rast ned mer enn den burde, er det kanskje ikke så lurt å hoppe over frokosten. Som moren din sier: «Frokost er dagens viktigste måltid!»

Du er **1200 kcal** over ditt daglige mål!



Så da kommer spørsmålet: Kan man egentlig «tracke» livet sitt uten at det går for langt? Det er fint å få vite at du bare går 3000 skritt dagen, som tilfeldigvis er veien til og fra skolen. Dette kan være flott motivasjon til å tvinge seg selv ut i fysisk aktivitet, slik at når måleren viser 10 000 skritt, kan du nyte en øl. Du er flink. Problemerkene starter først når målingene blir en større del av livet enn du hadde planlagt, og alt blir til tall. Det du spiser er tall på en måling som forteller deg hva du skal gjøre hver dag. Du kan plutselig ikke ta den tidligere nevnte ølen eller spise taco, fordi du da overskrider dagens mål. Livet ditt er kontrollert av en app. Det kan gi kunnskap, men det kan også gi så mye annet. Hvor fører egentlig alt dette?

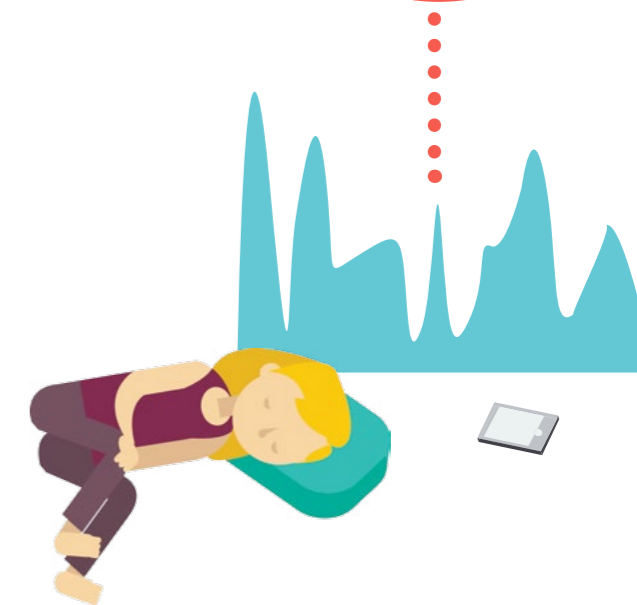
Sykt perfekt

I dagens samfunn har det å prestere blitt noe som føles som et press blant mange. Bidrar disse «prestasjonshjelpemidlene» til dette? De kan til en viss grad hjelpe med å holde fokuset på det du skal gjøre – de er din egen personlige trener som forteller deg alt du skal gjøre for å nå dine mål. Men det kan ende opp med at du sier nei til en kveld med vennene dine fordi du ikke kan spise de ekstra kaloriene, eller fordi det kan ødelegge søvnrutinen din. For noen mennesker blir dette livet deres. Alt de gjør er å skrive ned deres daglige egenskaper, og dersom de ikke gjør det en dag, kan det gå dem til hodet.

Hvor langt kan dette gå? Det kan gå langt, altfor langt. Noen som alltid har vært blid og sosial, blir plutselig fraværende fra deres

I natt sov du **3 timer** mindre enn gjennomsnittet.

Du var **34 %** mer urolig.



omgangskrets, og de eneste gangene man møter dem er på treningssenteret, eller når det passer til deres «plan». De kan ikke ta den ekstra filmkvelden med gjengen, fordi det kan føre til en ubalanse i dataene de samler inn. I lengden kan dette bli farlig, man vet jo egentlig ikke helt hvor man burde stoppe. Livet ditt blir noen tall på en skjerm som skal kontrollere deg som person. Du kan miste ditt sosiale liv, og kanskje verst av alt, miste kontrollen fordi du vet ikke hva som er «sunt» for deg lenger. Tracking kan få deg i fysisk aktivitet, og det kan lære deg mer om deg selv, gi deg kontroll på ting som du kanskje sliter med. Men det kan også gå for langt: Når du har begynt må du vite når du skal stoppe, ellers kan du plutselig ødelegge deg selv eller bli gal. Tenk på det, for det viktigste av alt er en balanse i hverdagen.

Ny kommode – en anekdote

Tekst: Fredrik Gram
Layout: Karoline Velsvik Berge

Artikler om prokrastinering i et studentmagasin er uunngåelig på lik linje med røverhistoriene til kompisen din i militæret – og tilsvarende originale. Hør heller på meg fortelle om da jeg dro på IKEA og kjøpte meg noe fint.

Angående prokrastinering

Det er ikke «Buzzfeed» som hviler i lankene dine. Dette er ikke en sånn «topp 10 måter å prokrastinere på»-story, og det var aldri meningen heller. Såkalte nedtelinger brødfør folk meninger de egentlig ikke har, og kan ansees å være en av de ledende grunnene til at man kaster bort tid i utgangspunktet.

Om dette innlegget faktisk var en rangering av tidstyver? Da hadde jeg paradoksalt nok valgt å plassere nedtelinger øverst. Nå ville det ikke vært særlig spennende å lese videre skjønt at du vet hva som befinner seg på topp, ville det vel? Katta er ute av sekken. Kast bort tiden din på noe annet.

Poenget er

Prokrastinering er ikke temaet, men snarere drivkraften bak eventyret jeg utsatte meg for tidligere i vinter. Himmelen var grå og bakken var brun – standard trønderisk juleføre. Alt lå liksom til rette for Fouriertransformasjoner, filtrering i frekvensdomenet, og angivelig noen kopper med kaffe der jeg omsider hadde funnet en komfortabel posisjon foran pulten. Og da, fullstendig uten forvarsel, skjedde det: Jeg fikk et *intest* behov for å kjøpe et nattbord.

Slik jeg anså situasjonen, var det uansett på høy tid at jeg gikk til anskaffelse av et nytt møbel. Over en lengre periode hadde jeg oppbevart et par ting på gulvet som utvilsomt kunne gjort seg bedre noen hakk høyere opp. Og ettersom viljen til å dra på IKEA faktisk hadde åpenbart seg, ville det vært dårskap å ikke ta affære – situasjonen kom nok aldri til å utspille seg igjen. Drømmen om et møblert rom var altså betinget

av hvorvidt jeg valgte å kaste meg på første buss mot Travbanen. Kausalitet er ustendig, dere.

IKEA-frykten

Dette er tydeligvis en greie. Litt sent ute i verdenshistorien til å gis latinsk navn, riktig nok, men like fullt en greie. Man har å gjøre med noe alle menn i en alder av 30 pådrar seg. Enten det, eller når de får stasjonsvogn. Og barn. Symptomene ser ut til å omfatte svette, sinne og ettertrykkelig re-evaluering av handlingene som ledet opp mot å ytre at «nei, nå tror jeg søren meg jeg drar på IKEA».

Så der var jeg da, i et forsøk på å orientere meg gjennom avdelinger, blant møbler så fullstendig pregløse at valget om å gi dem egennavn trolig var en spøk ingen tok. Jeg skal ikke lyve: Det var virkelig anstrengende. Ikke det at jeg har barn. Og hvorvidt det finnes en Volvo der ute registrert på mitt navn får heller være et anliggende til en

annen gang. Nei, anstrengelsene skyldtes følelsen av aldeles fortapelse, stadig tilspisset under jakten på «snarveier» som visstnok befant seg mellom et knippe avdelinger. Forsøket på å lokalisere dem førte meg bare lengre ned i kaninhullet.

I en tilstand av forvirrelse skjedde det endringer i agendaen, og nattbord ble erstattet med kommode – eller «nattkommode» om du vil. Avgjørelsen virket spontan, men fremskyndet likevel prosessen. Kall det gjerne en forsvarsmekanisme.

Selvbetjeningen var omsider innen rekkevide, og jeg fant mitt utvalgte møbel: Brusali, hvit nattkommode; reol 4, plass 6; 120 x 37 x 62 cm; tung som faen.

25 kilo veide den

Jeg innser at dette ikke høres *veldig* tungt ut på papir; en mellomstor hund veier gjerne

tilsvarende. Det foreligger derimot noen avgjørende ulikheter mellom hunder og kommoder: Husdyr er tilbøyelige til å «kveiles» om kroppen til den som løfter dem, slik at vekten fordeles over flere muskelgrupper. Møbler, på sin side, er ganske rigide, og som følge «kveiler» de seg i svært liten grad. Ikke kommer de når man kaller på dem heller.

Første etappe gjennom selvbetjeningen endte brukbart – jeg hadde tralle. Det jeg derimot *ikke* hadde var bil, og det er begrenset hvor langt man kommer med tralle. Jeg måtte ty til buss, hvorav nærmeste stopp av, tja, «grunner», befant seg ti minutter med gange fra bygget.

Så ble det bæring, da

Jeg måtte veksle mellom fem positurer, hvorav ingen var nevneverdig komfortable. Min favoritt var midlertidig den med pakken

hvilende som et tømmer over skulderen. I andre enden av smertespekteret var da jeg bar den opp under kortsiden med fingrene; det gjorde vondt, og jeg så bare papp.

Men nok tekniske detaljer. Du vil sikkert heller høre om hvordan jeg klarte å gå i feil retning av nærmeste holdeplass, og som følge måtte slepe på pakken én kilometer lenger enn nødvendig? Hvordan jeg bannet, slet og så ut som en dust? Det får bli en annen gang.

Jeg kokte

Det er ingen spøk: Det røyk faktisk fra armene mine da jeg vrentge av meg genseren. I en halvtime hadde jeg, så langt fysikkunnskapene mine rekker, påført tilsammen null arbeid på kommoden. Så om du registrerte meg stående og ryke ved siden av en brun pakke på Gildheim holdeplass, vet du nå hvorfor: Jeg hadde vært på IKEA og kjøpt meg noe fint.

Utgavens konkurranse

Herr Huangs hotell-heis-hysteri



Tekst: Eivind Kløvjan

Året er 20XX. Kina er verdens stormakt, og har i anledning Månefestivalen bygget verdens høyeste bygning: Et hotell med én million etasjer! Billionæren Huang Heng har bestemt seg for å bo fast på dette hotellet, og ønsker derfor å leie en hel etasje til eget bruk. Siden Heng framholder at 8 er et lykketall, ønsker han å leie etasjen som har heisknapp med tallet 888 888.

Hotelladministrasjonen er *minst* like overtroisk som Heng. De har bestemt seg for å hoppe over enkelte av etasjenumrene på heisknappene:

- Alle etasjenummer som inneholder ulykkestallet 13
- Alle resterende etasjenummer som inneholder tallet 4, men ikke dersom det også inneholder 8

Dessverre krever hotellresepsjonisten, som holder til på etasje 0, at Heng oppgir det *reelle* etasjenummeret til etasjen for å kunne leie det – altså det heisknappnummeret etasjen ville hatt dersom hotellet ikke hoppet over noen av etasjene på heisknappene.

Kan du hjelpe Heng med å finne etasjenummeret?

Send svaret til konkurranse@abakus.no med emne «HHHHH».

Blant alle konkurransesvar fra 2015 ble Vemund Santi trukket ut som vinneren av en middag for to, til en verdi av 2000 kr. Gratulerer! sponset av Genus

Vinnere

1. Johannes Ervum
2. Markus Andersen
3. Mads Bjørgan

Premier

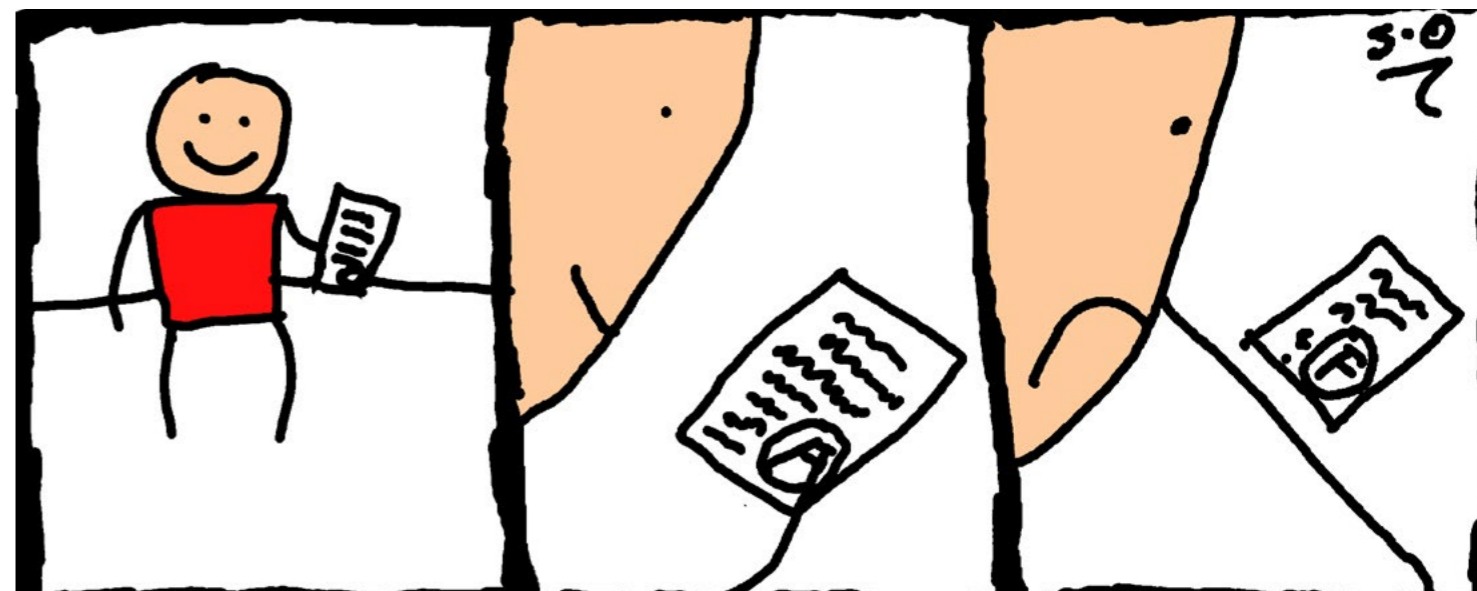
1. premie



2. premie



3. premie



TING JEG MAN TENKER PÅ LESESAL

JEG MÅ SÅ SYKT PROMPE.



SHIT, LUKKET JEG ALLE FANENE FØR JEG KOM HIT?

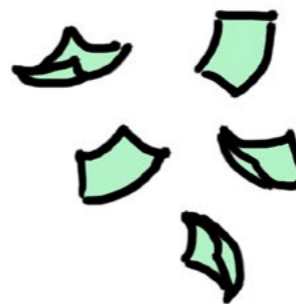


HVA ER LIVET? HVORFOR LEVER MAN EGENLIG? ER DETTE NOE FOR MEG? KLARER JEG DETTE? ER JEG FUNN E NOE?



SSH

HYORDAN JEG SER FOR MEG FRAMTIDEN MIN ETTER EN GOD DAG MED KODING.



Genus

... ETTER EN DÅRLIG DAG



Smått & nett

– det du leser først!



Rumperister

Vi kan alle enes om at det er deilig å kle seg naken og la alt flagre en gang i blant, spesielt etter en lang dag på skolen. PR-Mathias tok det kanskje litt langt, og arrangerte nakenvors. Det ble ensomt – bedre lykke neste gang!

Spøkelsespøk

En Arrkomer ikke ukjent for den gjengse S&N-leser var ikke verre på det, og inviterte alle jentene han matchet med på Tinder Plus til vors. Hans rause personlighet falt i god jord, og han fikk med seg en av gjestene på en mer privat fest på eget rom. De private festlighetene varte så lenge at noen måtte inn og sjekke hva som foregikk. Da var de begge forsvunnet ut av vinduet. Erfaringsmessig er snøsex noe kjølig, så hvis vedkommende har noen tips å dele, er det bare å sende det til **readme**, så vi kan informere i neste utgave.

Fort gjort

Det meste kan skje når man inntar alkohol, særlig i Åre. Det er lett å glemme å gå på do, for eksempel. Da må man ty til buksa – en ærlig sak. Men å sende bilder av det på Snapchat, det kunne man vært foruten. Intet gledelig syn, bare herlig fylleangst.

Verdig vinner

Vinneren av ledertrøya i Åre har virkelig stått på. Det er viktig å ikke gi seg, selv ikke når turen er over og bussen skal gå. Dersom man drikker alkohol til frokost og tar feil av klokken 13 og 15, blir det fort toget hjem. 240 kroner fattigere, og en dundrende hodepine rikere – **readme** gratulerer!

Abadate

Det er tydelig at det er potensiale i Abakos, spesielt for single, mannlige abakuler. Da Ababrygg-Vemund inviterte Abakos til å spise grøt, dukket det nemlig bare jenter opp. Måltidet var meget hyggelig, men det ble lite kos – Vemund er jo ikke singel.

Risikostyring

readme gratulerer Arrkom-Vilde og Magnus i 5. klasse med deres Facebook-offisielle forhold. Vi er meget imponert over at de turte å gjøre dette på nach. Vi anbefaler et emne i risikostyring som første date.

Jentepissoar

Dette med klær ligger visst ikke helt for PR-Mathias, for også om natten vandrer han gjerne naken omkring. En natt i Åre skulle han en tur på toalettet, men bommet grovt og endte opp på jenteværset. «Rom er rom», tenkte han, og urinerte på gulvet. Det falt ikke i smak hos jentene, og da de klagde dagen etter forsvarte han gjerningen. Uklokt.

Verktøy for jobben

Å ha sex er ikke bare-bare. Kompleksiteten øker i takt med seng og stilling. Kombinasjonen «doggy style» og køyeseng er ikke akkurat optimal, og det er stor fare for skader på alle parter. Da er det godt man har venner med vanlige senger. **readme** vil gjerne rose 1. klassingen for kreativ arealbruk i Åre, men vil påpeke at det kan være en god idé å spørre utlåneren på forhånd.

