

Nasjonal Forkunnskapstest i Programmering

Sondre S. Bolland

Institutt for Informatikk
Universitetet i Bergen

`sondre.bolland@uib.no`

1 Introduksjon

Med kunnskapsløftet 2020 er programmering nå pensum for alle elever ved studiespesialisering på videregående. Kommende høstsemester (2023) vil dermed alle nye studenter fra VGS allerede ha en viss kjennskap til hva vi underviser. Men i hvor stor grad kan de programmering og hva har dette å si for vår undervisning? Kan disse studentene allerede de grunnleggende delene av pensum i *Introduksjon til programmering* -emnene vi tilbyr? Har vi studenter som kan rett og slett hoppe over emnet i sin helhet? Hvordan kan vi tilrettelegge undervisningen til å komplimentere kunnskapen de allerede sitter med?

Vi ønsker å opprette en **forkunnskapstest i programmering** for å kartlegge kompetansen hos innkommende studenter. Norsk matematikkråd har siden 1984 gjennomført sin forkunnskapstest for å kunne overvåke nivået på forkunnskapene til begynnerstudenter på matematikktunge studier [12]. En lik standardisert test kan opprettes ved informatikk- og informasjonsvitenskap-studier i Norge. Resultatene kan brukes til å veilede framtidens programmeringsundervisning ved universitet og høyskoler.

I seksjon 2 gjør vi rede for programmering i læreplanene på grunnskole og videregående. Seksjon 3 oppsummerer et kartleggingsprosjekt for programmeringskunnskaper hos begynnerstudenter ved institutt for informatikk ved universitetet i Bergen, utført høsten 2022. Seksjon 4 gjør rede for Norsk matematikkråds forkunnskapstest som gir inspirasjon til dette prosjektet. I seksjon 5 inviterer vi til deltakelse i **Nasjonal Kommissjon for Forkunnskapstest i Programmering**.

2 Programmering på Grunnskole og VGS

De gjeldende lærerplanmålene går under navnet Kunnskapsløftet 2020 (LK20) [11], og tredde i kraft august 2020. LK20 inneholder nye kompetansemål for matematikkfaget som spesifikt nevner programmering. For grunnskolen finner vi de følgende kompetansemålene for ulike trinn i matematikkfaget [6]:

2. trinn: Lage og følge regler og trinnvise instruksjoner i lek og spill.

4. trinn: Lage algoritmer og uttrykke dem ved bruk av variabler, vilkår og løkker

- 5. trinn:** Lage og programmere algoritmer med bruk av variabler, vilkår og løkker.
- 8. trinn:** Utforske hvordan algoritmer kan skapes, testes og forbedres ved hjelp av programmering.
- 9. trinn:** Simulere utfall i tilfeldige forsøk og beregne sannsynligheten for at noe skal inntreffe ved å bruke programmering.
- 10. trinn:** Utforske matematiske egenskaper og sammenhenger ved å bruke programmering.

Algoritmisk tenkning og til dels programmering begynner nå allerede i andreklasser og fortsetter ut grunnskolen.

I matematikk 1 på videregående vektlegges algoritmisk tenkning og problemløsning i læreplanen. I teoretisk matematikk nevnes programmering eksplisitt: *"Formulere og løse problemer ved hjelp av algoritmisk tenkning, ulike problemløsningsstrategier, digitale verktøy og programmering"* [8]. Læreplanen legger ikke føringer på om det skal brukes blokkbasert eller tekstbasert programmering i undervisningen, men i eksamen høst 2022 finner vi oppgaver for Python-programmering (se appendiks A).

I praktisk matte så nevnes ikke programmering eksplisitt i læreplanen, men legger opp til økt fokus på problemløsningsstrategier for å løse matematiske problemer. I undervisvurderingen står det: *"Læreren skal være i dialog med elevene om utviklingen deres i programmering og strategier for å løse problemer"* [7]. Også her finner vi Python i høsteksamen 2022 (se appendiks A).

I andre- og tredjeåret blant de som går et studieforbredende utdanningsprogram velger elevene mellom matematikk for realfag (R1 og R2) og matematikk for samfunnsfag (S1 og S2). Både S2- og R2-matte nevner programmering i kompetansemålene: *"Utforske rekursive sammenhenger ved å bruke programmering og presentere egne framgangsmåter"* [10] [9], og i R1-matte finner vi Python-programmering i høsteksamen 2022 (se appendiks A).

Fagene *Informasjonsteknologi 1* (IT1) og *Informasjonsteknologi 2* (IT2) har eksistert som valgfag siden Kunnskapsløftet 2006 [4], med oppdaterte læreplanmål i 2021 [5], som inneholder blandt annet:

IT1:

- utvikle nettsider ved hjelp av markeringsspråk
- bruke algoritmisk tenkning og programmering for å utforske en problemstilling og presentere resultatet
- beskrive ulike typer algoritmer og vurdere effektiviteten av egen programkode

IT2:

- anvende objektorientert modellering til å beskrive et programs struktur
- utvikle objektorienterte programmer med klasser, objekter, metoder og arv
- generalisere løsninger ved å utvikle og bruke gjenbrukbar programkode

Dermed vil alle nye studenter høsten 2023 ha hatt programmering gjennom videregående, og alle kull etter vil også lært programmering på grunnskolen.

3 Kartlegging av Programmeringskunnskaper - Pilotstudie

Høsten 2022 ble de utført et kartleggingsprosjekt ved institutt for informatikk ved universitetet i Bergen. Motivasjonen for prosjektet var å kunne tilrettelegge undervisningstilbudet for de inkommande studentene som potensielt kunne mye av pensum fra før av. Hensikten var å kartlegge hvor utbredt programmeringskompetansen var før påbegynt studie, og innføre tiltak basert på denne informasjonen. Emnet *Innføring i programmering* (INF100) underviser grunnleggende Python-programmering og tar sted i førstesemesteret hos alle studielinjer ved instituttet. Emnet er også obligatorisk for alle studenter ved Det matematisk-naturvitenskapelige fakultet. Ved starten av dette semesteret var 680 studenter oppmeldt til INF100.

Kartleggingen bestod av en spørreundersøkelse hvor studentene selvrapporterte deres erfaring med programmering, og en forkunnskapstest med 19 flervalgs- og utfyllingsoppgaver som omhandlet Python-programmering (tidligere eksamensoppgaver). Forkunnskapstesten ble gjennomført i lik stil som en eksamen: oppgavene ble distribuert ved bruk av Inspira¹, laptopene var låst ved bruk av Safe Exam Browser² og ingen form for kommunikasjon eller hjelpemidler var tillatt. Alle studenter i emnet kunne melde seg på forkunnskapstesten, og ble gitt som insentiv at ved 80% poengsum ville man bli fritatt for 6 uker med kursarbeid (får automatisk godkjent de første 6 obligatoriske, ukentlige laboppgavene).

	%
Kan ingenting	49.6%
(1) Kan litt	27.98%
(2)	12.16%
(3)	7.57%
(4)	1.61%
(5) Dyktig	0.00%
Gjennomsnitt	1.65 / 5.0

Table 1. Svar til spørsmålet "Hvordan vil du rangere din programmeringsevne i Python (1 - 5)?" (n=436).

Studentene rapporterte at 50.4% har noe form for erfaring med programmering. De ble bedt om å rangere sin kompetanse i Python-programmering på en 5-punkts likert-skala (se tabell 1). Selv om over halvparten av studentene rapporterte at de hadde programmeringserfaring var ikke disse fullt så utlært. Med en gjennomsnittlig likert-score på 1.65 av 5 hadde de fleste begrensede ferdigheter. Av de 56 som deltok i forkunnskapsprøven var det kun 23 som bestod. Opprinnelig var kravet 80%, men etter en vurdering av besvarelsene og en bekymring for at enkelte oppgaver var for vanskelige ble dette kravet redusert til 64%.

¹ <https://www.inspera.com/>

² https://safeexambrowser.org/about_overview_en.html

De to mest kjente programmeringsspråkene blant studentene var Python (n=159) og Javascript (n=77). Ettersom IT2 underviser Javascript kan vi spekulere om de studentene med mest erfaring (etter dette kurset) ikke hadde stor kjennskap til Python, og dermed følte seg ikke dyktig nok til å delta eller presterte dårlig på kartleggingsprøven pga. lite kjennskap til den aktuelle syntaksen.

Basert på disse resultatene har det ikke blitt sett på som kritisk å gjøre tiltak for tilrettelegging i INF100 enda. Enkelte forslag har blitt diskutert om å opprette to forskjellige retninger innad i emnet for de som har tidligere erfaring og de som er nybegynnere. Men ingen spesifikke planer har blitt lagt. Av de 23 studentene som bestod forkunnskapstesten er det fortsatt tvil over deres kompetanse i den grad at de kan bli fritatt fra emnet i sin helhet. Kartleggingen vil fortsette og tilbudet om fritak for så mange ukers kursarbeid blir stående.

Men en viktig forskjell fra studentene i 2022 og 2023 er at de kommende vil ha hatt programmering som læreplanmål i en rekke fag utgjennom videregående, og forskjellen i programmeringskompetanse kan være til den grad at endringer må gjøres for 2024.

4 Forkunnskapstest i Matematikk

Siden 1984 har Norsk matematikkråd utført en forkunnskapstest i matematikk som måler begreper og ferdigheter fra pensum på grunnskolen. Testen spør også om kandidatens bakgrunn fra videregående, kalkulatorbruk og holdninger. Forkunnskapstesten ble iverksatt for å skaffe frem informasjon som kunne brukes til å overvåke nivået på forkunnskapene til begynnerstudenter, og til å understøtte matematikkrådets argumenter i saker der rådet ønsker å skape positiv utvikling for matematikkfaget i Norge [14]. Testen gjennomføres hvert andre år ved en rekke utdanningsinstitusjoner og oppgavene har stått uendret siden 2001. I studieåret 2017/2018 deltok 7170 begynnerstudenter [13]. Resultatene fra denne testen får ofte medieomtale:

2012

- [Ungdomsskolematte er helt gresk for lærerstudenter \(VG\)](#)
- [Studenter sliter med ungdomsskole-matte \(Adresseavisen\)](#)

2007

- [Noen må begynne på null \(NRK\)](#)

2000

- [Mattekrise \(VG\)](#)

Flere artikler har blitt publisert, men noen eldre lenker er utdaterte.

Matematikkrådets forkunnskapstest har gitt inspirasjon til å selv opprette en lik test for programmering. Ettersom dette prosjektet har pågått i 38 år er det mye erfaring og praksis å hente.

5 Nasjonal Kommissjon for Forkunnskapstest i Programmering

Vi ønsker å opprette et samarbeid mellom instituttene for informatikk og informasjonsvitenskap på landsbasis. Danne en **Nasjonal Kommissjon for Forkunnskapstest i Programmering**. Denne kommisjonen vil utarbeide og overse utførelsen av en standardisert programmeringstest som gis til alle begynnerstudenter ved ICT-studier utgjennom landet.

Ønsker du og din institusjon å delta i dette prosjektet ønsker vi gjerne at du melder din interesse ved å sende epost til: sondre.bolland@uib.no.

Ved deltagelse kan du være med å utarbeide hvordan studentene skal testes eller enkelt bare utføre testen på dine studenter, og alt i mellom.

References

- 1p 2022 høst lk20 løsning. https://matematikk.net/side/1P_2022_h%C3%B8st_LK20_L%C3%98SNING, hentet: 2022-12-13
- 1t 2022 høst lk20 løsning. https://matematikk.net/side/1T_2022_h%C3%B8st_LK20_L%C3%98SNING, hentet: 2022-12-13
- Eksamensoppgaver. <https://www.matematikk.net/side/Eksamensoppgaver>, hentet: 2022-12-13
- Læreplan i informasjonsteknologi lk06. <https://data.udir.no/kl06/inf1-01.pdf>, hentet: 2022-12-12
- Læreplan i informasjonsteknologi lk20. <https://data.udir.no/kl06/v201906/laereplaner-lk20/INF01-02.pdf?lang=nob>, hentet: 2022-12-12
- Læreplan i matematikk 1.10. trinn (mat01-05). <https://www.udir.no/lk20/mat01-05?lang=nob>, hentet: 2022-12-12
- Læreplan i matematikk fellesfag vg1 praktisk (matematikk p) (mat08-01). <https://www.udir.no/lk20/mat08-01>, hentet: 2022-12-12
- Læreplan i matematikk fellesfag vg1 teoretisk (matematikk t) (mat09-01). <https://www.udir.no/lk20/mat09-01>, hentet: 2022-12-12
- Læreplan i matematikk for realfag (matematikk r). <https://data.udir.no/kl06/v201906/laereplaner-lk20/MAT03-02.pdf?lang=nob>, hentet: 2022-12-12
- Læreplan i matematikk for samfunnsfag (matematikk s). <https://data.udir.no/kl06/v201906/laereplaner-lk20/MAT04-02.pdf?lang=nob>, hentet: 2022-12-12
- Læreplan kunnskapsløftet 2020. <https://sokeresultat.udir.no/finn-lareplan.html?fltypefiltermulti=Kunnskapsl%C3%B8ftet%202020&filtervalues=all>, hentet: 2022-12-12
- Matematikkrådets forkunnskapstest. <https://matematikkraadet.wixsite.com/matematikkraadet/forkunnskapstesten>, hentet: 2022-12-12
- Nortvedt, G.A., Bulien, T.: Rapport. norsk matematikkråds forkunnskapstest 2017 (2018)
- Rasch-Halvorsen, A., Johnsbråten, H.: Norsk matematikkråds undersøkelse blant nye studenter høsten 2001 (2002)

A Eksamensoppgaver

Høsteksamen 2022 i matematikk T [2] (se figur 1).

Høsteksamen 2022 i matematikk P [1] (se figur 2).

Høsteksamen 2022 i matematikk R1 [3] (se figur 3).

Oppgave 3

```
1 def f(x):  
2     return (1 - 2 * x) / (x - 2)  
3  
4 x = 8  
5  
6 while x >= -8 :  
7  
8     print(x , f(x))  
9     x = x - 1  
10
```

```
8 -2.5  
7 -2.6  
6 -2.75  
5 -3.0  
4 -3.5  
3 -5.0
```

Lars har skrive ein programkode. Ovanfor ser du koden, og resultatet Lars får når han køyrer programmet.

Når programmet har skrive ut dei seks linjene, kjem ei feilmelding.

- Kva ønskjer Lars å bruke programmet til, og kvifor får han ei feilmelding?
- Foreslå endringar Lars kan gjere i koden for å unngå feilmeldinga.
- Skisser grafen til funksjonen f som Lars har definert i linje 1 og 2 i koden.

Fig. 1. Oppgave fra høsteksamen 2022 i matematikk T.

Oppgave 5

```
1 def f(x):  
2     return 3 * x - 15      # Definerer funksjonen f gitt ved  $f(x) = 3x - 15$   
3  
4 x = 0  
5  
6 while x <= 10:  
7  
8     if f(x) == 0:  
9         print(x)  
10  
11     x = x + 1  
12
```

Lars har skrive programkoden ovanfor.

- a) Kva ønskjer han å finne ut?
Kva blir resultatet når han køyrer programmet?
- b) Kva vil resultatet bli om han endrar funksjonsuttrykket til $x^2 - 6x + 8$?

Lars endrar funksjonsuttrykket til $x^2 - 144$ og ser at han må gjere noko med programmet.

- c) Foreslå endringar Lars kan gjere.

Fig. 2. Oppgave fra høsteksamen 2022 i matematikk P.

Oppgave 5

Marianne har skrive følgende program:

```
1 def f(x):
2     return (6*x-3)/(x-1)    # Definerer funksjonen f(x) = (6x-3)/(x-1)
3
4 h = 0.00001
5 def Df(x):
6     return (f(x+h)-f(x))/h
7
8 a = 1.5                    # En startverdi
9 while Df(a) < -3:
10     a = a+0.001
11
12 b = f(a) - Df(a)*a        # Regner ut konstantleddet
13
14 print("y = -3x +", b)
```

Bestem verdien av variabelen b som blir definert på linje 12.

Fig. 3. Oppgave fra høsteksamen 2022 i matematikk R.