













OPPGAVER og IDEER








Oppgaver med, og i regneark og Python




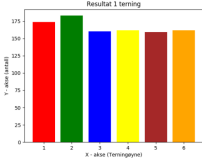

Matematisk emne:	Sannsynlighet, med fokus på en og to terninger.
Arbeidsmetoder:	Gruppearbeid, individuelt arbeid og klasseromsdiskusjoner. Plukk gjerne relevante oppgaver for din gruppe.
Antall oppgaver	20 oppgaver med ulik varighet/vanskegrad + 2 websider med oppgaver innebygget via trinket.io (python 3.6)
Filformater	Filene er i format Google dokumenter og Google regneark, og har kun lesetilgang. For at elevene skal kunne redigere, må de ta en kopi av filen. Dersom man har tilgang til plattform med mulighet for å tildele en kopi til hver elev (OneNote, Google Classroom etc) anbefales det.
Regneark Laget for google.sheet.	Du finner regneark som kan kalles “programmerte” regneark. Disse inneholder effekter skapt med bl.a mye bruk av hvis -setninger og logisk sammenheng mellom ulike celler som vi betrakter som variabler Disse er også et ledd i det å bli tryggere og mer effektive i bruk av regneark i matematikk. Elever vil kunne klare å lage egne “programmerte” regneark med ulike effekter.
Python Webbasert: trinket.io	I hovedsak vil oppgavene vise elever som ikke har sett python tidligere: 1) Hva et python-program kan være 2) Hvordan starte et pythonprogram 3) Finne igjen matematiske element og kunne endre disse i kodene 4) Se fordelene med python ved høye antall av simulerte kast
Mer python utforskning Kan brukes individuelt eller i gruppearbeid 	For interesserte grupper av elever, kan noen andre ressursbanker benyttes: Enkle og vanskeligere oppgaver, grunnopplæring i python og matematikk (Løses individuelt eller i grupper) Disse sidene vil bli oppdatert med flere videoer og oppgaver. De vil bli redigert etter som erfaring tilsier det. Generell begynner python-opplæring og matematikk : Website: Python og matematikk i ungdomsskolen (Kan brukes som eget opplegg i grunnopplæring python) Python opplæring med noe sannsynlighetsregning : Website: Python og noe sannsynlighet (for elever med behov for mer utfordringer)






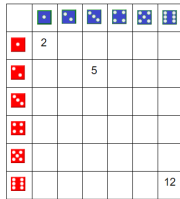


Elevhefte	Elev-veiledning kan du finne her. Denne bør lærer selv fylle ut etter å ha testet oppgaven, fått et eierforhold til den, og og gi beskjeder til elevene for hver oppgave som lærer ønsker de skal gjøre. F eks. om og hvordan gruppearbeid/ læringspartner-arbeid skal foregå. Det kan være nyttig å gjennomføre læringsvideoer og individuelle oppgaver som hjemmearbeid i forkant av undervisningsøkten.
Lærerveiledning	Tabellen nedenfor.

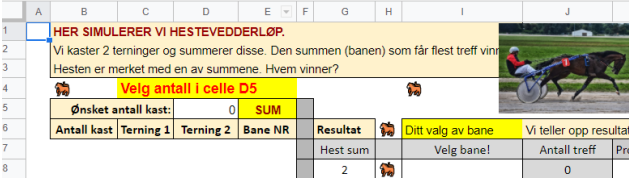
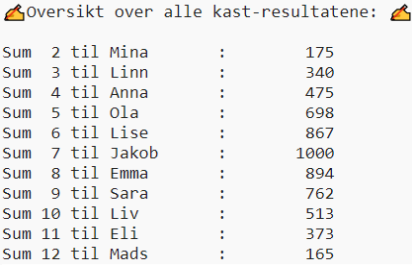
Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder
		Informasjon og ferdige filer/videoer for lærer		Filer til elevene (ligger også i oppgavedokumentene)	
Tid i min				NB! Elevene må lage kopi for å kunne redigere i regnearkene.	
10	Oppgave 1	Intro: Lurt å vise elevene hvordan de skal jobbe og forventning om å forstå. Lærer kan på forhånd ha gått igjennom begrepene.	Et eksempel på “programmert” regneark. Elevene kan lære ord og begreper knyttet til sannsynlighet.  Hjelpetil: Ord og uttrykk (Kan deles med elever) Får se bruke av if eller hvis-setning	 Elev-filen (sannsynlighet - fakta med spørsmål)	
15	Oppgave 2	Intro: Lurt å vise elevene hvordan de skal jobbe	Her lærer/repeterer eleven begrep innen sannsynlighet. De kan da også se eksempel på regneark som gir responser via if eller hvis-setninger	 Elev-filen (Gunstige - Mulige. Begreper med spørsmål)	
10	Oppgave 3	Intro. Forutsetter forkunnskaper i kombinatorikk. Trene på sannsynlighet og kombinatorikk. Skal kun skrive i blå felter. Husk at om man skiver brøk, må man skrive = foran!	Et eksempel på regneark som er “programmert” med ulike responser ved bruk av hvis-setning. Slike ark kan elever lage i andre fag med andre spm og responser.	 Elev-filen (Regneark med respons)	

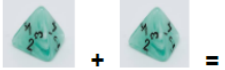
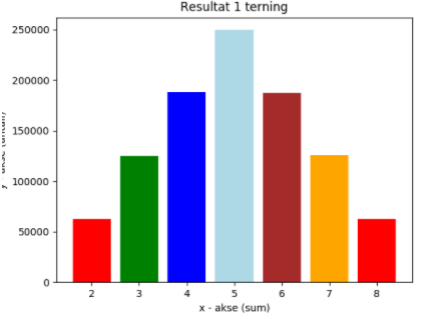
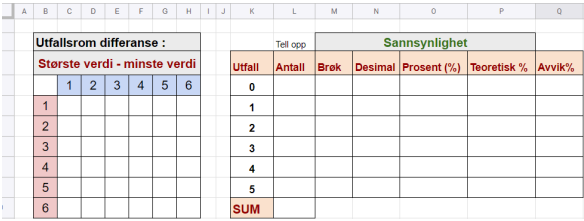
Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder
15	Oppgave 4	<p>Her skal elevene trille terninger og føre inn resultatet</p> <p>Dette kan gjøres med vanlige terninger eller via regneark eller pythonkode. Fint om de får prøve alle metodene, slik at de får se mulighetene</p>	<p>Erfaringer:</p> <ol style="list-style-type: none"> 1) De får se at regneark kan brukes som terning. 2) De får se det 1 python-koden som triller 1 terning (denne triller og samler opp resultatet på slutten). 3) Samme som ovenfor men med kodevinduet også. 	<p> 1)Kaste 1 terning</p>	<p> 2)Python uten kode</p> <p> 3)Python med kodevindu også</p>
10	Lærer organiserer felles innhenting av klassens resultat	<p>Resultatet samles inn i regnearket Klassenes samlede resultat (lærer fører inn og deler, eller gruppene fører inn selv) Disse resultatene skal brukes i oppgave 5.</p>	<p>Resultatet skal samles sammen når gruppene har fått sine resultat. Fin anledning til å prate om få og mange kast og resultat.</p>	<p> Klassenes samlede resultat</p>	
30-40	Oppgave 5	<p>Bearbeide data fra oppgave 4</p> <p>Videoene kan gis som lekse hjemme, og forberede eleven til å lage og vise de ulike tallrepresentasjonene.</p> <p>Disse skal vi også se på i python-program senere.</p> <p>Forberedende tenkning og spørsmål om store talls lov. Lurt å ta en oppsummering felles.</p>	<p>Kan brukes som omvendt undervisning (Lekse)</p> <p> Hvordan skrive brøk og prosent</p> <p> Hvordan lage stolpediagram</p>	<p>Bruke verdier fra oppgave 4 i regnearket:</p> <p> Klassenes samlede resultat</p>	

Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder
30	Oppgave 6	<p>Å vite hvordan man kan simulere mange kast. Første eksempel blir å “programmere” dette i regneark. Se video hjemme eller felles undervisning i klasserommet.</p> <p>Ressursark: Formler vi bruker og skal lære oss (kan deles med elevene)</p>	 Trinn 1: Tilfeldige kast i regneark	<p>Bruk gjerne denne slik at alle har samme cellereferanser når vi øver</p> <p> :Mal som kan brukes</p>	
10	Felles	<p>Nå kan det være på plass å ta opp temaet store talls lov. (kan velge å ta dette senere om det passer bedre)</p> <p>Læringsressurs som kan deles med elevene: Store talls lov, eller bare vises og snakkes om</p>	<p>Her er et regneark lærer kan bruke og vise resultat fra 1 til 10000 kast, Regnearket vise bare de antall kast som er ønskelig, Skriv kun i celle C3 (senere kan elevene programmere dette)</p> <p>Ressurs:  Regneark-fil med mulige resultat 0 -10000 kast. (Test med 5 ,10, 100, 1000, 10000 kast)</p>		<p> Python-kode som viser det samme.</p> <p>Her kan lærer vise 1-5 000 000 kast eller mer. Vise effekten og hvor raskt det går.</p>
30	Oppgave 7	<p>Elevene kan få teste et pythonprogram. Det kan være lurt å forklare at alle program har en kode i bakgrunnen av det vi ser på skjermen.</p> <p>Trinket som vi bruker her, viser pythonkoden i et vindu og resultatet i det andre vinduet.</p> <p>Lurt å vise linje for linje hva resultatene blir i denne korte koden Elevene kan også se videoen om dette.</p>	 Videoforklaring på begge kodene til høyre. (Denne videoen finner du i oppgaven også) <p>Vis elevene: print(“Tekst”) print() --tom linje Variablen terning1=randint(1,7)</p>		<p> Python nr 1 Et meget enkelt program (uten forklaringer)</p> <p> Samme program: Python nr 2 Et meget enkelt program med forklaringer</p>

Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder
15	Oppgave 8	<p>Målet er her å lære om variabler og bytte ut verdier til noen variabler i python.</p> <p>Sammenligner variabler i programmeringsspråk, med variabler i form av celleverdier i regneark</p>	 <p>Videoforklaring på hvordan du bruker python her</p>		 Pythonprogram “Trill og regn ut resultat av et utfall”
30	Oppgave 9	<p>Mål:</p> <ul style="list-style-type: none"> -finne variabler i kodeer -endre variabler -lage nye variabler -kopiere kode og gjøre nødvendige endringer 	<p>I denne oppgaven skal elevene øve mer på hva variabler er ved å endre og beregne verdier av variablene.</p> <pre> 2 #-----variabler----- 3 antall_kast= 1000 4 gunstige= 6 5 antall_gunstige= 0 6 desimal=0 7 prosent=0 </pre>		 Program: Trille en terning og skrive ut resultat
20	Oppgave 10	<p>Programmet viser alle verdiene ved kast med terning</p> <p>Hovedmål:</p> <ul style="list-style-type: none"> - erfare simuleringer - se tidsbesparelse ved bruk av programmering - teoretisk sannsynlighet - elevene kan beregne tidsbruk ved vanlig kast og differansen ved bruk simuleringer 	<p>Her vises alle kast og vi viser ved mange kast diagram.</p> 		 Program: Pythonkode - viser resultat på alle kastene

Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder
40	Oppgave 11	<p>Nå er vi tilbake i regneark. I denne oppgaven legger vi vekt på logikk-trening, og det å se sammenheng mellom celler.</p> <ol style="list-style-type: none"> 1) Vis dem først hva de skal oppnå ved å bruke regnearket som er ferdig. 2) I stedet for å bare vise videoer, kan det være lurt å lose dem gjennom ved å vise dem starten, for deretter å utfordre dem å finne ut av det selv. Mange klarer det om de vet hva de skal frem til. <p>Det er muligheter til å kopiere formler så langt man ønsker det.</p>	<p>Ferdig løsning som lærer kan bruke til demonstrasjon:</p> <p> "Programmering" i regneark</p> <p> Demonstrasjon</p> <p> Videoopplæring programmere antall kast</p> <p> Videoopplæring telle-tabell ved antall kast</p> <p>Kommandooversikt (kan deles med elevene)</p>	<p> Arbeidsmal: Avansert regneark - terningkast</p>	
5-10	Oppgave 12	<p>Denne oppgaven er ment som forarbeid til å tenke igjennom hvilke summer som er mulige når man kaster 2 terninger. Kan gjøres som forarbeid hjemme.</p> <p>Det kan også være fint å vise elevene denne lenken.</p>			
30	Oppgave 13	<p>Mål:</p> <ul style="list-style-type: none"> -lære elevene smartkopiering og låse rad og kolonne ved summering i tabell -logisk tenkning 	<p> Videopplæring i smart kopiering</p>	<p> Mulige summer og smart kopiering</p>	

Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder										
10-15	Oppgave 14A	<p>Felles i klassen. Her er et regneark som er "programmert" til å simulere et hesteveddeløp der summen av 2 terninger avgjør farten.</p> <p>Elevene blir utfordret på hvilken strategi de vil velge for å kunne vinne. Hvordan kan de med dårlig plassering øke sine sjanser for å vinne?</p>	 <p>Flinke og/eller interesserte elever kan bli utfordret på å lage lignende regneark.</p>	<p>🔗 "Programmert" regneark: Hesteveddeløp</p>											
10-15	Oppgave 14B	<p>Felles i klassen. Man kan alternativt eller i tillegg vise og spørre de samme spørsmålene ved å bruke python som simulator.</p> <p>Elevene kan ha de samme refleksjonene og samtidig øve mer på å oppdage hvordan koder kan skrives</p>			<p>🔗 Python Hesteveddeløp</p>										
20-40	Oppgave 15	<p>Felles: Innledende spørsmål om sannsynlighet ved sum av 2 terninger.</p> <p>På gruppe:</p> <ol style="list-style-type: none"> 1) Teste og prøve et "programmert" regneark. Øve smart formelbruk. 2) Lage spørsmål og svar med respons i regneark. 	<p>Tar noe mer tid enn oppgitt om man velger å ta utfordringene.</p> <p>Sannsynlighetquiz</p> <table border="1" data-bbox="869 1029 1496 1129"> <thead> <tr> <th>Nr</th> <th>Spørsmål</th> <th>Svar som brøk. Husk = for svar.</th> <th>Poeng</th> <th>Respons</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Hvor stor sannsynlighet er det å få sum > 11 ?</td> <td>1/36</td> <td>1</td> <td>Så bra</td> </tr> </tbody> </table> <p>Å lage slike regneark med poeng og respons kan være en fin trening. Når svarene skal gis som brøk må cellen formateres til brøkceller - det kan være en utfordring her. Det hadde vært lettere om svarene var vanlig tall.</p>	Nr	Spørsmål	Svar som brøk. Husk = for svar.	Poeng	Respons	1	Hvor stor sannsynlighet er det å få sum > 11 ?	1/36	1	Så bra	<p>🔗 Mal oppgave 15: Oppgave sum 2 terninger</p> <p>🔗 Et start-regneark til siste utfordring</p>	
Nr	Spørsmål	Svar som brøk. Husk = for svar.	Poeng	Respons											
1	Hvor stor sannsynlighet er det å få sum > 11 ?	1/36	1	Så bra											

Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder
15	Oppgave 16	<p>Anvende kunnskap i ny sammenheng.</p> <p>I denne oppgaven undersøkes summer med pyramide-terning.</p>	<p>Demofiler for lærer.</p> <p>🔗 En enkel løsning i python</p> <p>🔗 En løsning i python med fordeling</p> <p>👤👥🖨️ Pyramide-terning</p> 	<p>🔗 Pyramide-terning - notatfil</p>	<p>🔗 Pyramide-terning: Python-koden</p>
15	Oppgave 17	<p>Tester, undersøker, pyramide-terninger i pythonprogram</p> <ol style="list-style-type: none"> 1) Tester tid og kast 2) Ser på diagram 3) Sjekker koden 			<p>🔗 Python-koden for pyramideterminer</p>
20	Oppgave 18	<p>I denne oppgaven utforsker man differansen mellom resultat av 2 terninger. Bør introduseres i fellesskap.</p> <p>Man får mer øving på både bruke regneark og tolke python-kode.</p>		<p>🔗 Regnearket for notater</p>	<p>🔗 Pythonkoden: Differanse 2 terninger (Størst - minst)</p>
Tid	Oppgaveark	For lærer	Innhold + Opplæringsvideoer-+docs	Regnearkfiler	Python-koder

30	<p>Oppgave 19</p>	<p>For viderekomne: For de elevene som gjerne kan tenke seg å lage mer avanserte regneark som ledd i å utvikle nyttige verktøy på nye måter, kan denne oppgaven gi muligheter for det. Den bygger på kunnskap fra oppgave 11. Man kan fjerne alle formler i malen om man ønsker høyere vanskegrad. Nå er det 3 start-formler.</p>	<p>Et ferdig regneark som lærer kan bruke å vise mulig løsning:</p> <p> Simuler sum av 2 terninger med regneark</p> 	<p> Regneark som mal til oppgaven</p>	<p> Sum 2 terninger med Demofil i python</p>
	<p>Oppgave 20</p>	<p>For viderekomne: Lage regneark som simulerer 2 terningers: -differanser -multiplikasjoner -bruke andre terninger med sum, differanse og multiplikasjon og utforske utfallsrom og sannsynlighet</p>	<p>3 ferdige filer for lærer:   </p> <ul style="list-style-type: none"> - differanse med negative tall - differanse absoluttverdi - produkt 	<p> Regneark som mal til oppgaven</p>	<p> Differanse med python. Kan brukes som demo:</p>