

Lyckohjul

Här presenteras en aktivitet som kombinerar matematik och programmering i Scratch. Syftet är att ge eleverna ytterligare ett verktyg för att öka förståelsen för sannolikhet och statistik. Eleverna får träna på att presentera en given procentsats i en cirkel. Förhoppningen är också att elevernas motivation för lärande i matematik och programmering ökas genom skapande av en animation.

Sedan införandet av programmering i matematikämnet i de svenska och norska läroplanerna har det pågått diskussioner om huruvida man kan lära sig matematik med hjälp av programmering. Undersökningar visar bland annat att lärare initialt förknippar programmering med att följa instruktioner, med olika aspekter av logiskt tänkande samt med samarbete och kommunikation. Programmering ses av många lärare mer som ett pedagogiskt verktyg för att öka elevernas engagemang i matematikundervisningen än som ett sätt att lära sig ett specifikt matematikinnehåll inom exempelvis algebra och geometri.

I samband med läroplansrevideringen har en studie genomförts där svenska matematikböcker för årskurs 1–6 har analyserats med hänsyn till innehållet kring programmering. Ett av syftena med studien var att se om uppgifterna bidrog till att bilda en bro mellan programmering och matematik. Författarna drog bland annat slutsatsen att programmeringsinnehållet i relativt liten utsträckning lyckas skapa en bro mellan programmering och matematik, speciellt inom algebra där fördelen med att använda variabler inte är tydlig nog.

När man arbetar med programmering är det inte alltid lätt att välja en specifik metod, men olika studier har visat att kunskapen i programmering förbättras när man arbetar med att avläsa och följa kod. Ett sätt att göra detta är att eleverna får en färdig programkod och får i uppgift att försöka förut säga vad programmet kommer att göra. Efter detta kan eleverna köra koden, eventuellt göra ändringar och skapa sin egen version.

För att eleverna ska få en bättre förståelse för matematik är det fördelaktigt att arbeta med olika representationer. Studier har visat vikten av att den abstrakta och symboliska notationen för bråk bör konkretiseras. Exempelvis kan bråk representeras av bråkcirklar, rektanglar eller liknande. Att låta eleverna dela in cirkeln i olika sektorer kan vara ett komplement till dessa resurser. Mot denna bakgrund hoppas jag att mitt exempel kan ge inspiration att arbeta med programmering där man möjligen kan se ett tydligare band mellan matematik och programmering.

Genomförande

Programmet nedan kan se både omfattande och krävande ut om eleverna ska skriva koden direkt. Därför kan det vara bättre att dela upp det i mindre enheter. Programmet i Scratch består av flera delar. Först ett program som hör till lyckohjulet, sedan ett program som markerar var hjulet stannar, i vårt fall markerat med en röd pil. I det sista programmet uppdateras alla variabler och procentsatsen beräknas.

Att skapa lyckohjulet

För att kunna dela upp cirkeln i sektorer kan vissa elever först behöva arbeta med konkreta bråkdelar eller rita figurer. Som tidigare nämnt kan användning av olika representationer ge en bättre förståelse för bråk och hur man representerar dem i cirkeln. När elever skapar cirkeln i Scratch är det inte säkert att de behöver speciellt mycket information om hur de ska gå tillväga. Här kan "trial and error" vara en bra ingång för eleverna att skapa sin egen version utifrån hur den enskilde läraren vill ha indelningen av cirkeln.

För att skapa lyckohjulet och den röda pilen gör så här:

- ♦ Gå till sprajt-ikonen i det nedre högra hörnet av ditt Scratch-fönster och välj måla. Se figuren nedan.
- ♦ Välj bitmapp.
- ♦ För att skapa en perfekt cirkel, håll nere shift-tangenten och använd vänster musknapp för att justera storleken på hjulet.
- ♦ Flytta cirkeln till mitten av fönstret.
- ♦ Välj linje för att dela cirkeln i önskad storlek.
- ♦ Färglägg de olika områdena.

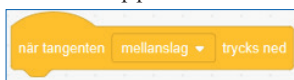
Var noga med att koppla den röda pilen till rätt färg på lyckohjulet i den sista delen av programmet som styr rotationen. Eftersom det finns otaliga färgnyanser är denna funktion väldigt viktig för att få rätt matchning.



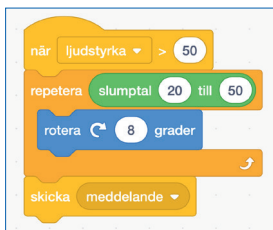
Rotation av lyckohjulet

Eleverna kan få den färdiga koden, sedan ska de diskutera vad programmet gör innan de kör det. De kan givetvis få göra egna justeringar och skapa sin version av programmet. När programmet är skapat kan elever som har svårigheter med programmering välja att manuellt skapa statistik över resultaten. I detta sammanhang kan det vara naturligt att ta upp sannolikhet om man så önskar. Här ges instruktionen att lyckohjulet kan startas genom att göra ett ljud, men givetvis finns det andra sätt att starta och stoppa rotationen.

Om du vill att hjulet ska styras av tangentbordet kan ljudnivåblocket bytas ut mot mellanslagsblocket.



I koden sätts de olika variablerna först lika med noll. Här är det bestämt att en rotation av hjulet ska vara 8 grader och att antalet varv ska vara mellan 20 och 50, vilket ger rotationen ett minimum och maximum. När rotationen upphör registreras färgen och de olika variablerna uppdateras med hjälp av meddelandet som skickas till den sista delen av programmet.



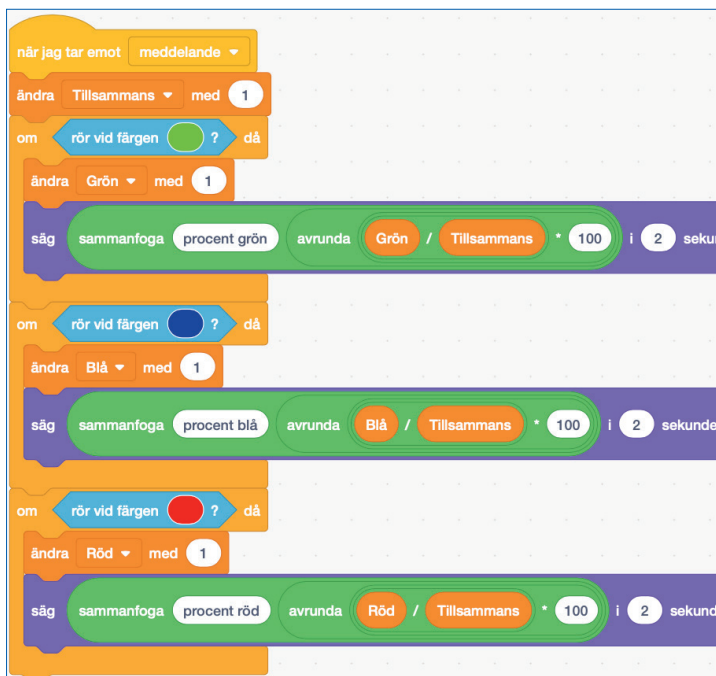
Uppdatera variabler och beräkna procentsatser

Även i den här fasen får eleverna först diskutera vad programmet gör innan de kör det. Programmet registrerar vilken färg hjulet stannar på där den röda pilen är placerad. Samtidigt uppdateras variablerna. Programmet räknar sedan ut procentandelen för respektive färg hjulet har stannat på. Procentsatsen anges varje gång hjulet har slutat rotera.

Den här delen av programmet kan vara svår. Inledningsvis kan man plocka bort de lila blocken där beräkningarna görs. Eleverna kan istället få i uppgift vid varje variabeluppdatering att beräkna antal utfall per totala antalet körningar och antingen ange det i bråkform eller som procent. På så sätt kan fler elever få utmaningar på sin nivå och ändå se nyttan av att samarbeta.

Egna erfarenheter

Jag har märkt att det hjälper eleverna att strukturera sina tankar när vi diskuterar vad programmet kommer att göra. En del elever kan behöva få lite tips om hur



lyckohjulet kan konstrueras för att de ska kunna ta sig an uppgiften och utmana sig själva att få den klar.

Hjulets rotation kan programmeras på flera sätt. Generellt sett är min erfarenhet att när eleverna arbetar i par kommer det att uppstå många givande diskussioner. Jag tror också att elever som vill skriva av den färdiga koden bör få göra det. Man kan baka goda kakor utan att vara konditor.

Ett annat sätt kan vara att eleverna får tänka ut och föreslå vilka block som behövs för att bygga upp programmet. Efter att ni enats får de sätta ihop blocken i rätt ordning. Fördelen med detta är att eleverna får en måttlig utmaning utan att gå vilse i alla val av block – en form av stöttning som vissa kan behöva. Elever med god förståelse kan givetvis prova sig fram själva.

Somliga har inte alls arbetat med Scratch medan andra är mer drivna. Konstruktionen av själva hjulet har gått bra liksom hur man kontrollerar de slumpmässiga rotationerna. Feedback från lärare är att lyckohjulet visar nytan med variabeln och tydliggör hur den uppdateras. Den stora utmaningen har varit att sätta ihop blocken som räknar ut procentsatserna. Många lärare uttrycker att det kommer att vara kämpigt för elever på mellanstadiet. Flera av dem menar att aktiviteten kan användas redan på lågstadiet men att man då nöjer sig med att bara räkna utfallen för de olika färgerna och illustrera resultatet i en frekvenstabell. I samtal med lärarna har det också dykt upp idéer om hur aktiviteten kan utvecklas eller användas i olika sammanhang:

- ♦ Att göra två lyckohjul som illustrerar de olika kombinationerna man får när man väljer till exempel klädesplaggen tröja och byxor.
- ♦ Att låta hjulet bestämma "dagens tal", som eleverna ska använda för olika beräkningar.

LITTERATUR

- Bråting, K., Kilhamn, C. & Rolandsson, L. (red) (2021). *Programmering i skolmatematiken – möjligheter och utmaningar*. Studentlitteratur.
- Bråting, K., Rolandsson, L. & Kilhamn, C. (2022). *Vad hände med programmeringen?* Nämnaren 2022:4.
- Hector, E. & Thelander, L. (2022) *Kan programmering bidra till lärande i matematik?* Nämnaren 2022:1.
- Kilhamn, C., Rolandsson, L. & Bråting, K. (2021). Programmering i svensk skolmatematik. *LUMAT*, 9(1), 283–312.
- Sentance, S., Waite, J., & Kallia, M. (2019). Teaching computer programming with PRIMM: a sociocultural perspective. *Computer Science Education*. 29(2–3), 136–176.
- Webbsida: *Exploring pedagogies for teaching programming in school*
suesentance.net/2017/02/20/exploring-pedagogies-for-teaching-programming-in-school/
- Länk till projektet: <https://scratch.mit.edu/projects/783282389>