Skolverket

Aktiviteter Del 4

Här finns ett antal aktiviteter att välja mellan. Det ena handlar om att bestämma omkrets för cirkeln i Python och det andra innebär att programmera Buffons nålproblem i Scratch för att på så sätt kunna beräkna ett närmevärde på π.

Ni ska planera en lektion där eleverna skapar programmet och där de också förväntas argumentera varför deras program är korrekta. Om ni väljer den första uppgiften finns det stöd för planeringen i form av frågor att diskutera med kollegor.

Beroende på kunskapsnivån i Sctatch/Python respektive den matematiska kunskapsnivån kan ni behöva ge olika mycket matematisk respektive kodningsmässig leding.

Orkestreringen av lektionen ska alltså vara anpassad för att både hinna ge nödvändig matematisk bakgrund och lagom mycket kodrelaterade tips och förklaringar.

Det behövs också tid för att samla upp intressanta elevproduktioner, belysa likheter och skillnader och låta eleverna argumentera för varför deras program löser problemet.

Cirkelns omkrets på nytt sätt

Lennart Rolandsson, Uppsala universitet

Denna aktivitet kommer att beskrivas meh hjälp av programspråket Python. Det går naturligtvis att använda aktiviteten med andra programspråk. Python kan köras med en vanlig webläsare på websidan repl.it. I Python finns två funktioner som används för att rita polygoner, left (vinkel) och forward (sidans längd). Argumenten i de funktioner som används beskrivs med vinkeln som sköldpaddan vrider sig, respektive sträckan som sköldpaddan vandrar. Om vi repeterar dessa funktioner uppstår en polygon. I följande tabell finner du tre exempel där funktioner repeteras 4, 6 och 9 ggr. Låt eleverna själva arbeta med dessa algoritmer för att undersöka vad som krävs för att programmet skall rita en cirkel. Diskutera i mindre grupper vad eleverna har kommit fram till. (Slutsats: kortare längder och fler vinklar).

Sköldpaddan ritar en kvadrat	**Sköldpaddan ritar en sexsidig månghörning**	**Sköldpaddan ritar en niosidig månghörning**
from turtle import * fred = Turtle() fred.shape("turtle") fred.reset()	from turtle import * bob = Turtle() bob.shape("turtle") bob.reset()	from turtle import * alice = Turtle() alice.shape("turtle") alice.reset()

for i in range (4):
    fred.forward(100)
    fred.left (90)
	for i in range (6):
	    bob.forward(100)
	    bob.left (60)
		for i in range (9):
		    alice.forward(100)
		    alice.left (360/9)

Tabell 1. Tre exempel på funktioner som ritar månghörningar.

För att eleverna skall få en känsla för vad som menas med en ”sluten polygon”, kan sköldpaddans vridning beskrivas steg för steg tills den vandrat ett varv, eller 360 grader, och hittat tillbaka till sitt ursprungsläge. Man bör eventuellt fråga sig huruvida en månghörning kan ritas för alla möjliga typer av månghörningar. I så fall, hur många grader bör sköldpaddan vrida sig i varje steg för att den skall rita en polygon med 4, 6 eller 9 sidor? I följande tabell med tre exempel har en funktion definierats som ritar polygonen. Funktionen anropas med polygon (antal sidor, längden per sida).

Månghörning	**Månghörning**	**Månghörning**

from turtle import *
pensize(2)
pencolor(‘green’)
 	
def polygon (n, side_length):
    for i in range (n):
        left (225)
 	
    forward(side_length)
 	
polygon (8, 100)
	from turtle import *
	pensize(2)
	pencolor(‘blue’)
	
	def polygon (n, side_length):
	    for i in range (n):
	        left (360/n)
	
	    forward(side_length)
	
	polygon (9, 100)
		from turtle import *
		pensize(2)
		pencolor('red')
		
		def polygon (n, side_length):
		    for i in range (n):
		    left (360/n)
		    forward(side_length)
		
		n = 12
		for i in range(n):
		    polygon (i, 100)

Tabell 2. Olika funktioner som ritar månghörningar.

Med exemplet till höger kan elever undersöka hur en polygon närmar sig en cirkel då antalet hörn i polygonen ökar (värdet på *n* går att sätta till andra värden än 12). Utmaningen är hur sidans längd skall förändras då antalet sidor i polygonen ökar för att inte cirkeln skall växa utanför skärmen.

Att tänka på innan lektionen

Eleverna förväntas söka efter den vinkel som sköldpaddan måste vrida sig, för att cirkelns omkrets skall motsvara värdet för cirkelns diameter multiplicerad med π. Det finns dock pedagogiska problem. I denna aktivitet uppstår ett behov av matematisk förståelse för att inte aktiviteten skall riskera att bli en övning utan reflektion och djupare kunskap. Ni bör därför fundera på vad som är rimligt att elever på egen hand kan hitta med en sektor och en sekant, om sektorn närmar sig formen av en kil som blir smalare och smalare. Några diaktiska frågor att ta ställning till:

- Vilka typer av ”stöttor” behöver eleverna för att hitta sekantens längd och vad som händer med sekanten då den ”närmar sig” cirkelns periferi?
- Skall man diskutera sidor eller hörn, då polygonens algoritm presenteras?
- Ska man i introduktionen av aktiviteten visa sekantens längd med en bild och förklara hur den beräknas $(2 \cdot radien \sin\frac{360}{2\cdot n})$?
  Ska eleverna få en färdig algoritm där de undersöker förhållandet mellan omkretsen och diametern på cirkeln, det vill säga värdet på π (genom att öka talet n, antalet sekanter)?

Jakten på ett gränsvärde – utvidgning av cirkelns omkrets

Exemplet med cirkelns omkrets går att applicera på andra kurvor, som till exempel sinus, cosinus eller tangens (obs! singularitet), där eleverna får i uppgift att undersöka kurvans längd på liknande sätt. Principen är liknande där sekantens längd beräknas med $\frac{\pi}{2\cdot 180}$, $\sqrt{f(x)\cdot f(x) - f(x+h)\cdot f(x+h)}$, genom att låta h succesivt anta mindre värden, som till exempel $\frac{\pi}{180}$, $\frac{\pi}{2\cdot 180}$, $\frac{\pi}{4\cdot 180}$, …

Att tänka på innan lektionen:

- Vilka problem kan eventuellt uppstå för elever som förväntas bygga algoritmen där h antar allt mindre värden och summan närmar sig ett gränsvärde?
- Måste man visa eleverna var sekanten är och hur dess längd beräknas?
- Exemplet med en kurvas gränsvärde illustrerar hur elever handgripligen upptäcker kurvans längd som ett konvergerande gränsvärde. Går det att praktiskt legitimera begreppet gränsvärde med andra metoder, utan kunskaper i algoritmer och programmering?
- Vilka fördelar/nackdelar ser ni med algoritmer som byggs i ett programmeringsspråk eller med ett kalkylschema i ett kalkylprogram?
- Finns det eventuella problem med antalet värdesiffror? När kan denna typ av problem uppstå?

Buffons nålproblem

Morten Misfeldt, Aalborg universitet

Buffons nålproblem är ett klassiskt problem inom geometrisk sannolikhetsteori formulerat av G.L.L. Buffon. Problemet innebär att vi släpper en nål med längden l på ett på ett papper med parallella linjer med det givna avståndet x. Hur stor är sannolikheten p för att nålen kommer att korsa en linje på pappret? Avståndet mellan raderna ska vara större än längden på nålen.

Denna planering är tänkt för 90 min.

- 20 min introduktion,
- 50 minuter programmering och undersökning (tre uppgifter)

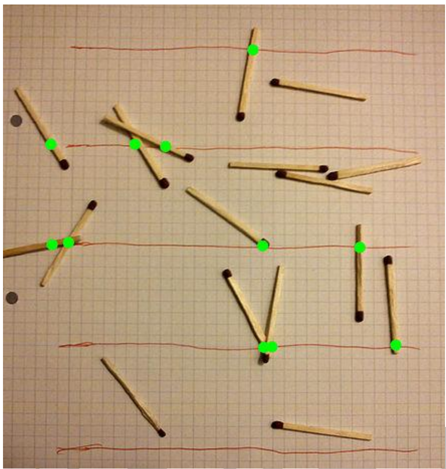
- 20 min summering

Introduktion

Presentera problemet. Berätta eventuellt lite om Buffon. Fokusera i den första uppgiften på det praktiska problemet att bestämma sannolikheten att nålen ska korsa en linje.

Uppgift 1 (Kan hoppas över eller bara beskrivas)

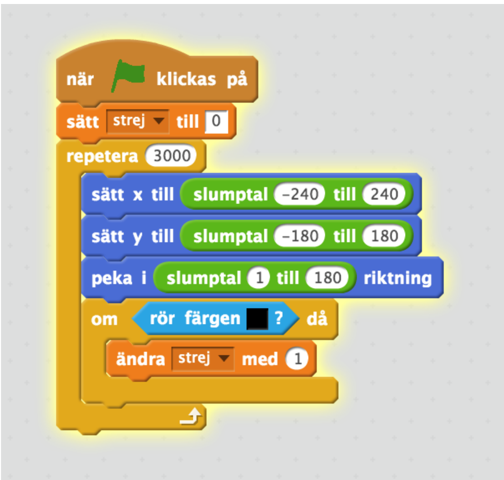
Genomför konkreta försök med till exempel 20 kast.



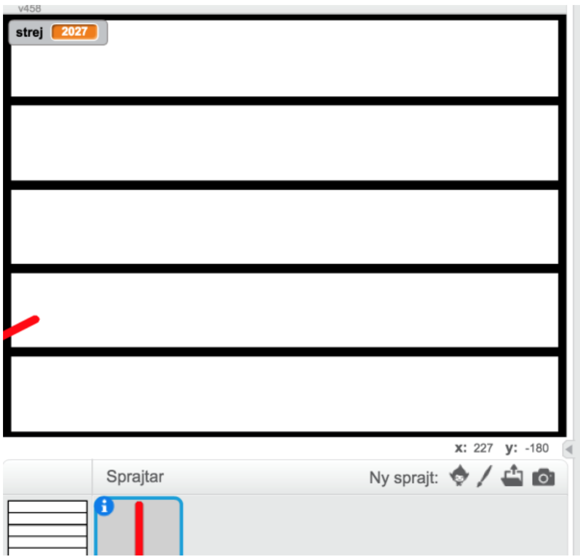
Figur 1. Konkret arbete med Buffons problem.

Uppgift 2

Skriv ett program som simulerer sådana kast. I figur 2 finns ett exempel på hur det kan se ut.



Figur 2. Kod til Buffons nålproblem.



Figur 3. Bakgrund och Sprajt till Buffons nålproblem.

Uppgift 3

Man kan visa att sannolikheten i Buffons nålproblem kan beräknas med formeln $p = \frac{2l}{\pi a}$, där p är sannolikheten, l är längden på nålen, och a är avståndet mellan raderna. Med hjälp av denna formel kan man även bestämma π empirisk.

a) Förklara varför formeln kan hjälpa oss att uppskatta π genom experiment.
b) Använd er modell i Scratch för att bestämma π.
c) Är Scratchprogrammet bättre eller sämre än experiment, om man vill bestämma π?

Diskutera gemensamt i helklass

- Diskutera nålproblemet igen – förstår eleverna problemet?
- Förklara den analytiska lösningen av problemet. Varför innebär den att vi kan finna π empiriskt?
- Diskutera vad exemplet säger om vilken betydelse för matematikämnet som experiment med hjälp av datorer har.