



Slump och statistik med Scratch

Lektionen handlar om att simulera tärningskast och skapa en statistikapplikation genom att arbeta med modifiera algoritmer.

Lektionsförfattare: [Måns Jonasson](#)



Till läraren

1. Vad är risk och chans?
2. Inspektera ett program som simulerar tärningskast
3. Kasta tärning
4. Avslutande reflektion
5. Extrauppgift: Simulera tärningskast med Javascript
6. Nästa lektion

LÄRARINSTRUKTIONER

Lektionens syfte

Syftet med lektionen är att låta eleverna simulera tärningskast genom att modifiera en programmerad algoritm och göra bedömningar och statistik av chans.

Förberedelser

En förutsättning för att göra denna övning är att eleverna har grundläggande kunskaper i hur [blockprogrammeringsverktyget* Scratch](#) fungerar som programmeringsspråk. [Här hittar du lektionsserien "Scratch för nybörjare"](#) som passar både lärare och elever som är ovana vid programmering.

Den här lektion förutsätter även att eleverna är bekanta med begreppet algoritmer, gärna genom att först ha genomfört lektionen "[Undersök och diskutera algoritmer](#)".

Tänk på att det är viktigt att förmedla att programmering handlar om att prova, testa och göra om. Misstag och fel är nödvändiga delar för att kunna lösa problem. Prata gärna om begreppet bugg genom att använda material från lektionen "[Buggar eller fel vid programmering](#)".

*Blockprogrammeringsverktyg är ett verktyg där eleverna arbetar med programmering

genom att använda block som sammanfattar kodsnuttar i "vanlig" kod. Scratch är utvecklat av [MIT – Massachusetts Institute of Technology](https://www.mit.edu/).

Genomförande

Titta på produktionen [Tärningskast i Scratch](#) tillsammans med eleverna och låt dem sedan remixa produktionen för att skapa en algoritm som löser uppgiften.

LÄROPLANSKOPPLING

Skolans uppdrag

Skolan ska bidra till att eleverna utvecklar förståelse för hur digitaliseringen påverkar individen och samhällets utveckling. Alla elever ska ges möjlighet att utveckla sin förmåga att använda digital teknik. De ska även ges möjlighet att utveckla ett kritiskt och ansvarsfullt förhållningssätt till digital teknik, för att kunna se möjligheter och förstå risker samt kunna värdera information.

Centralt innehåll i ämnet matematik (årskurs 7-9)

- Hur mönster i talföljder och geometriska mönster kan konstrueras, beskrivas och uttryckas generellt.
- Hur algoritmer kan skapas och användas vid programmering.
- Programmering i olika programmeringsmiljöer.
- Matematisk problemlösning: Hur algoritmer kan skapas, testas och förbättras vid programmering för problemlösning
- Statistik. Bedömningar av risker och chanser utifrån datorsimuleringar och statistiskt material.

Lektionsdel 1:

Vad är risk och chans?

Instruktioner

Arbeta enskilt med att repetera begreppen risk och chans. Läs [avsnittet om sannolikhet på matteboken.se](#). Titta eventuellt på filmerna på samma sida.

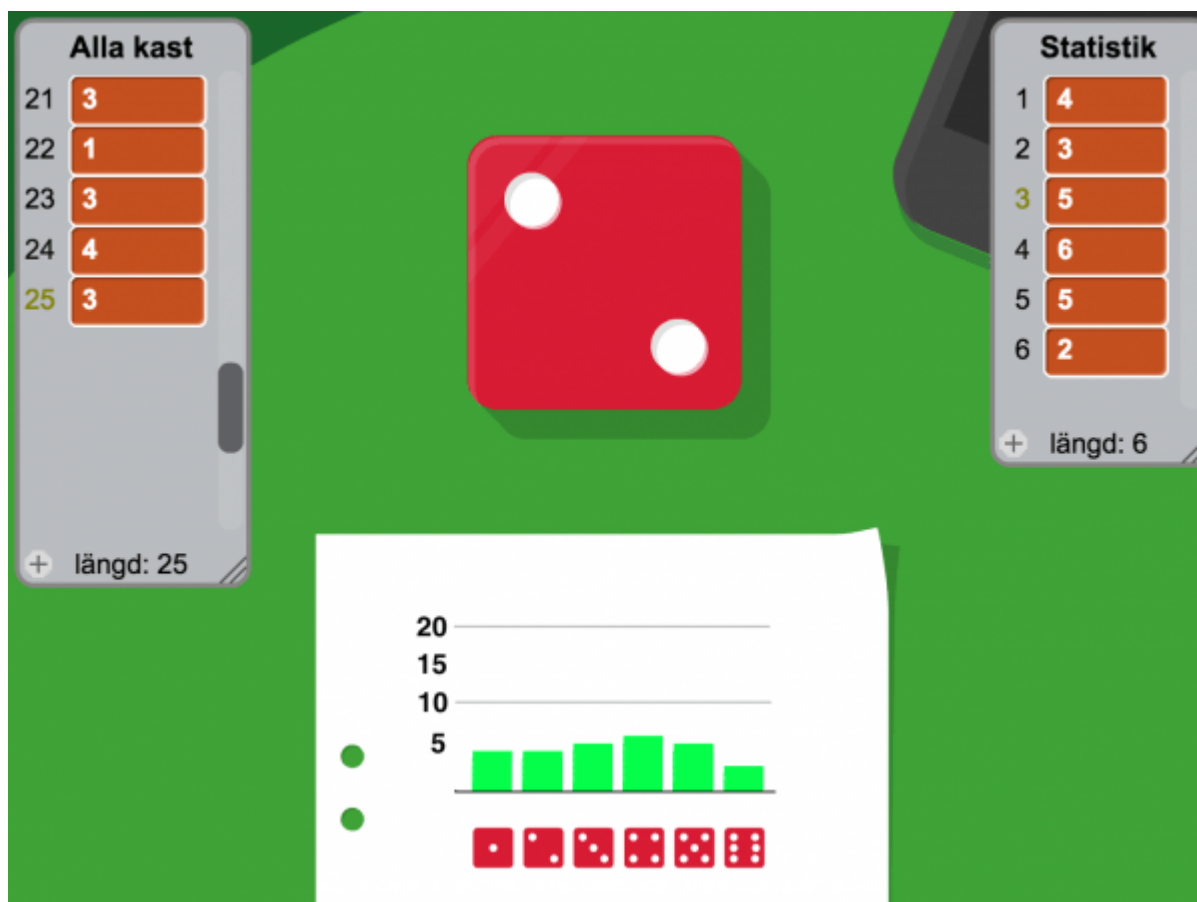
Lektionsdel 2:

Inspektera ett program som simulerar tärningskast

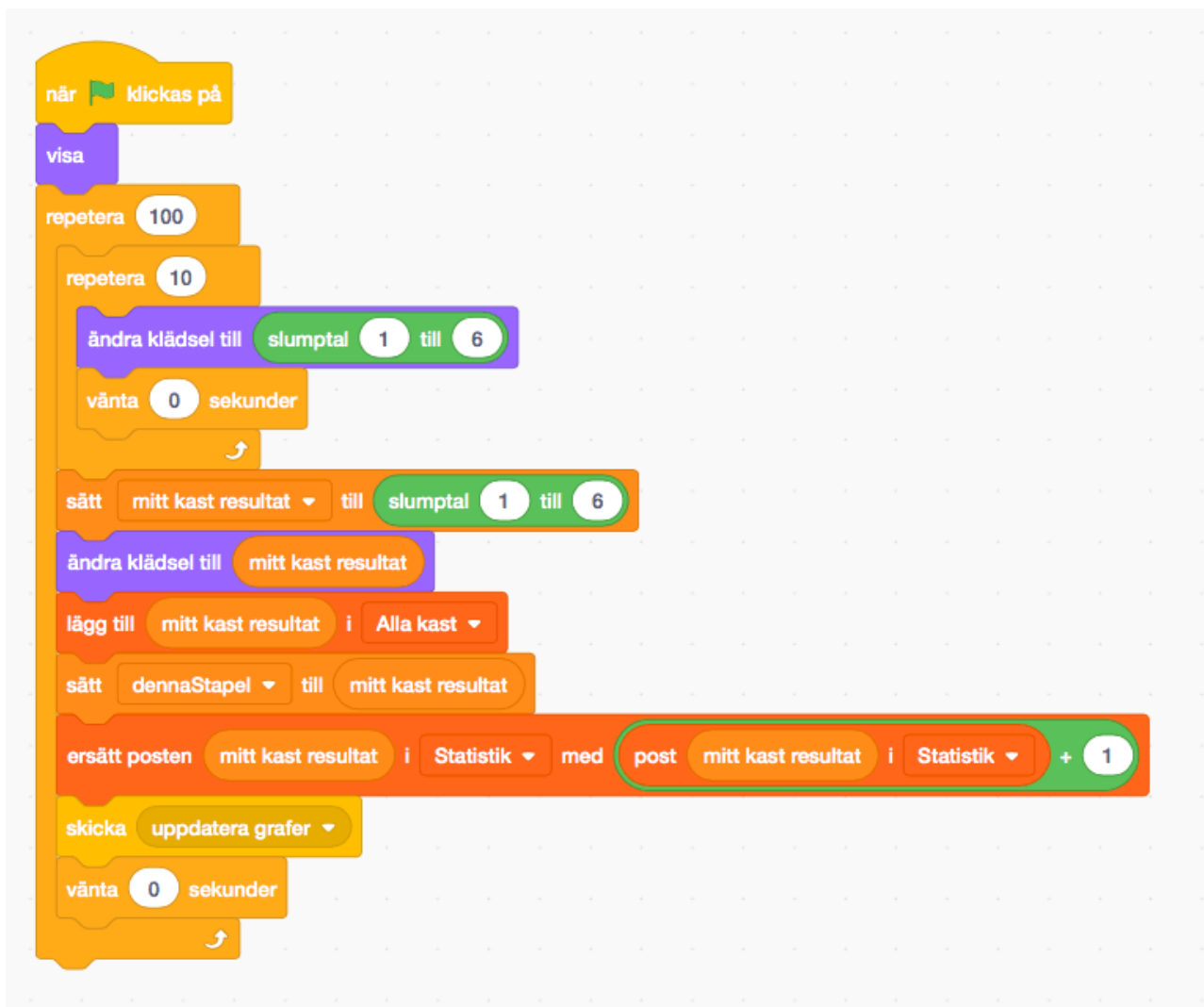
Instruktioner

Jobba i par eller grupper med övningarna.

1. Inspektera [Scratch-programmet Tärningskast](#).
2. Gå igenom koden på insidan och tolka de olika delarna. Hur fungerar koden som slumpar tärningskast? Hur lagras resultaten från tärningskast och hur genereras statistiken?
3. Simulera 100 tärningskast i programmet och notera utfallet.



Koden som är kopplad till tärningsprajten är den viktigaste, den ser ut så här:



Klicka på bilden för att se den större

Den här koden går att förklara, uppifrån och ned, på följande vis:

- När programmet startas
- Visa tärningsbilden
- Starta en loop med 100 repetitioner (för att göra 100 tärningskast)
 - Starta en loop med 10 repetitioner (för att animera tärningen)
 - Byt bild på tärningen till en slumpmässigt vald som visar en till sex prickar
 - Vänta 0 sekunder (experimentera med olika tal här för att se vad som händer)
 - Lagra resultatet av ett slumpantal mellan ett och sex i variabeln "mitt kast resultat"

- Visa samma antal prickar på tärningen genom att byta klädsel till motsvarande
- Lägg till resultatet av detta tärningskast i listan "Alla kast"
- Lagra resultatet av detta tärningskast i variabeln "dennaStapel"
- Uppdatera listan "Statistik" genom att addera 1 till motsvarande plats i listan
- Skicka meddelandet "uppdatera grafer" som sätter igång ett annat script på annan plats
- Vänta 0 sekunder (experimentera med olika tal här för att se vad som händer)

Lektionsdel 3:

Kasta tärning

Instruktioner

Arbeta i par med övningen.

1. Kasta en vanlig tärning 100 gånger och för statistik över resultatet - antingen på dator eller med papper och penna. Jämför utfallet mot de simulerade kasten. Analysera om resultaten skiljer sig, och varför.
2. Modifiera koden så att programmet bara utför 50 tärningskast. Jämför utfallet mot de första 100 som programmet kastade och de 100 som kastades med en riktig tärning. Hur skiljer sig utfallen? Analysera.
3. Ändra nu koden så att programmet bara utför 10 tärningskast. Jämför utfallet mot de första 100 som programmet kastade och de 100 som kastades med en riktig tärning. Hur skiljer sig utfallen? Analysera.
4. Till sist, ändra koden så att programmet utför 200 tärningskast. Jämför utfallet mot alla andra utfall du fått hittills. Hur skiljer sig utfallen? Analysera.

Lektionsdel 4:

Avslutande reflektion

Instruktioner

Jämför resultat och utfall med en annan grupp.

- Skiljer sig utfallen åt? Varför då?
- Diskutera statistik och chans i sammanhanget.
- Går det att förbättra programmet?

Lektionsdel 5:

Extrauppgift: Simulera tärningskast med Javascript

Instruktioner

Scratch är en utmärkt miljö för att lära sig programmeringslogik. För en mer avancerad nivå kan den här övningen vara ett bra alternativ.

Javascript-övningen går att köra i en dator med webbläsare. Övningen är framtagen av Mikael Tylmad, författare av läromedel i Javascript.

[Se video](#)

Lektionsdel 6:

Nästa lektion

Instruktioner

Utveckla dina kunskaper i programmering genom lektionen: [Programmera ett övergångsställe](#)



Klar? Klicka här