Simple Coding of Statistical Simulations

T³ Europe, Brussels – March 2017 – Nevil Hopley

"We will show you how to construct simulations of experiments of chance. We shall use the minimum amount of code, aiming to use just the Calculator and Data & Statistics applications as much as possible. This is intended to help teach hypothesis testing to statisticians who have no coding experience. There will be ample extension material for the more code-curious!"

Simulations can be run in two main ways
1. Generate results from repetitions of experiments
2. First create a theoretical sample space and then randomly sample from it

This document showcases both methods.

The simulations are listed in the order of increasing complexity and sophistication.
You are advised to work through them in order to gradually develop and understand the techniques that are used in the later simulations.

TI-Nspire Skills Required/Developed

- Accessing the Catalogue of all commands, by pressing and 1
- On a calculator page, pressing ▲ repeatedly to highlight a previous calculation and then pressing enter to paste it into the current command line.
- Insert a new Data & Statistics page by pressing ctrl doc▾ then 5: Add Data & Statistics

TI-Nspire Commands Used

- randInt(lowerbound, upperbound)
- randInt(lowerbound, upperbound, repetitions)
- randSamp(list, sample_size)
- randSamp(list, sample_size, 1)
- seq(expression, variable, lowerbound, upperbound)
- countIf(list, condition)
- count(list)
- constructMat(expression, row variable, column variable, number rows, number cols)
- mat▶list(matrix)

Statistical Skills Required/Developed

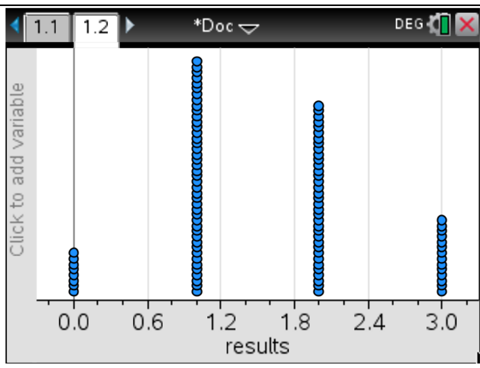
- Knowledge that a D6 is a six sided fair die numbered 1 to 6. Similarly, D4 is fair die numbered 1 to 4, D8 is a fair die numbered 1 to 8.
- Plotting the results from simulations to see their distribution
- The idea of comparing a 'test statistic' to simulated results
- Estimating a p-value (the measure of how 'extreme' a test statistic was)

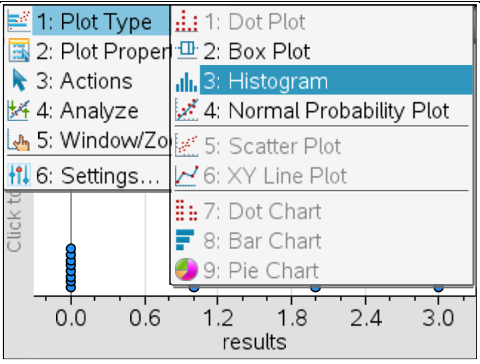
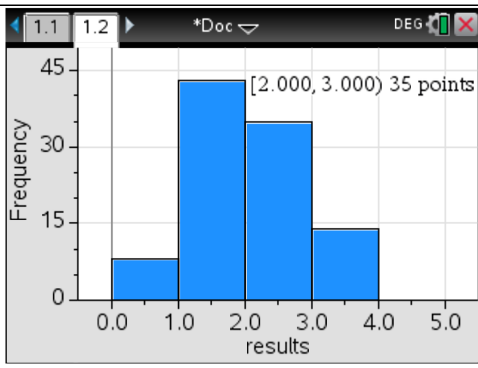
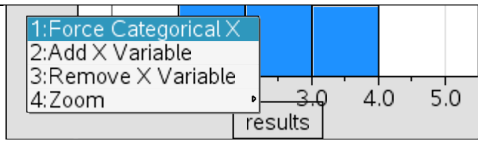
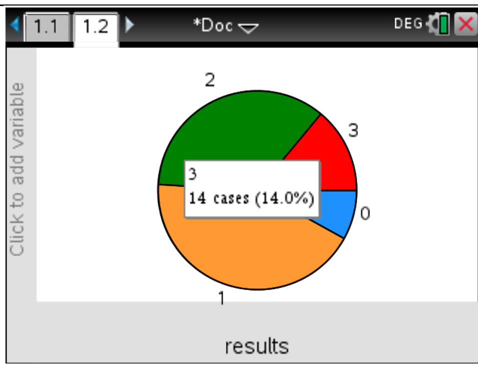
All screenshots from TI-Nspire OS 4.2

Authored by Nevil Hopley
Version 1: February 2017
Version 2: April 2017
www.CalculatorSoftware.co.uk

Challenge Problem 1 – Tossing Three Coins

I tossed a coin 3 times and noted the number of heads.
How likely is it to get 3 heads?

Task	Keypad Help	Screenshot
First, set the pseudo random number seed. __Any__ number can be used – we suggest a mobile phone number to ensure uniqueness!	menu 5: Probability 4: Random 6: Seed	1.1 *Doc DEG RandSeed 441314466000 *Done*
Simulate just one toss of a coin by using the **randInt(…)** command	menu 5: Probability 4: Random 2: Integer	1.1 *Doc DEG RandSeed 441314466000 *Done* randInt()
Specify integers from 0 to 1, where 1=Head, 0=Tail		randInt(0,1)
Add three random integers together to obtain the results of three coin tosses	Use ▲ to highlight and enter to paste in previous lines.	randInt(0,1) 0 randInt(0,1)+randInt(0,1)+randInt(0,1)
Check it works!	Use enter several times	randInt(0,1)+randInt(0,1)+randInt(0,1) 1 randInt(0,1)+randInt(0,1)+randInt(0,1) 3 randInt(0,1)+randInt(0,1)+randInt(0,1) 0 randInt(0,1)+randInt(0,1)+randInt(0,1) 2
Use the sequence command, **seq(…)** to run it 100 times	Use 📖 S and ▼ to get **seq( )** Use ▲ to highlight and enter to paste in previous lines.	seq(randInt(0,1)+randInt(0,1)+randInt(0,1),*n*,1,100)
Store this simulation in a variable called ***results***	Use ctrl [templates key] to get :=	seq(randInt(0,1)+randInt(0,1)+randInt(0,1),*n*,▸ {1,1,2,1,2,1,3,1,2,1,3,0,1,1,2,1,1,3,2,1,2,2,1▸ *results*:=seq(randInt(0,1)+randInt(0,1)+randIı▸
Plot the distribution of the ***results***.	ctrl doc▾ 5: Add Data & Statistics	

Task	Keypad Help	**Screenshot**
Change the plot type from Dot Plot to Histogram	menu 1: Plot Type 3: Histogram or ctrl Menu 2: Histogram	
View the frequencies of the number of heads obtained.	Move the cursor over any bar to see its count frequency.	
Change ***results*** to categorical data	Move the cursor over the x-axis label ***results***, then ctrl menu 1: Force Categorical X	
Change the plot type from Histogram to Pie Chart	menu 1: Plot Type 9: Pie Chart or ctrl menu 2: Pie Chart then Move the cursor over any sector to see its % frequency.	

From this simulation result, we obtained 3 heads 14% of the time.

Probability Theory Information

The distribution plots generated are simulations of a Binomial distribution, where n = number of coins, and p = 0.5

X = number of heads obtained when tossing a coin 3 times
X ~ Bin(3,0.5)
The theoretical probability of scoring 3 heads is given by P(X=3)=0.125

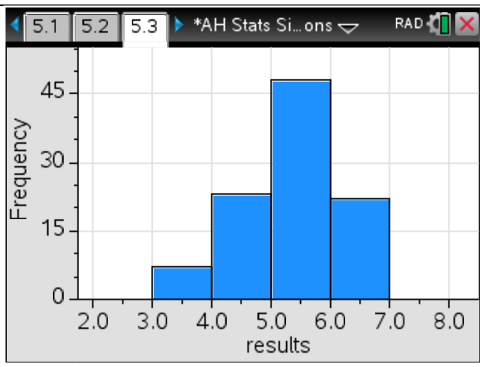
```
binomPdf(3,0.5,3)                0.125
```

Extension/Variants

Change the commands to run 1000 simulations of tossing 4 coins.

Challenge Problem 2 – Rolling Two Dice with Non-Standard Numbering

**A fair six-sided die has 1 on one face, 2 on two of its faces and 3 on the remaining three faces.
The die is thrown twice.
What is the distribution of the total score?**

Task	Keypad Help	Screenshot
First, define the ***die*** with the required numbers	Use ctrl to get := Use ctrl) to get { }	1.1 *Doc DEG *die*:={1,2,2,3,3,3} {1,2,2,3,3,3}
Roll two of these special dice independently. This requires sampling <u>with</u> replacement.	Not including the third argument of **randSamp( )** gives sampling <u>with</u> replacement.	randSamp(*die*,2) {3,1}
Sum the sample to obtain the total score. Repeat this process 100 times. Store the simulation ***results***	Use S and ▼ to get **seq( )**	sum(randSamp(*die*,2)) 5 seq(sum(randSamp(*die*,2)),*n*,1,100) {6,6,5,4,3,6,4,6,5,4,6,6,3,3,6,6,5,4,6,6,2,5,6, *results*:=seq(sum(randSamp(*die*,2)),*n*,1,100) {6,5,4,5,6,4,4,6,4,4,5,4,4,4,4,5,5,6,6,5,4,4,6,
Plot the distribution as a histogram.	ctrl doc▾ 5: Add Data & Statistics	

Probability Theory Information

You can generate the sample space of all the outcomes (below) and compare its frequencies to the simulation's results.

	1	2	2	3	3	3
1	2	3	3	4	4	4
2	3	4	4	5	5	5
2	3	4	4	5	5	5
3	4	5	5	6	6	6
3	4	5	5	6	6	6
3	4	5	5	6	6	6

This table also be created on the TI-Nspire – see the technique in Challenge Problem 8 and adapt it for this problem.

Extension/Variants

Find the distribution of the <u>product</u> of three rolls of the same number die.

Challenge Problem 3 – Picking a Team from a Mixed Group Without Replacement

A team of three is to be randomly chosen from 4 boys and 5 girls.
How likely is it to obtain an all girl team?

Task	Keypad Help	Screenshot
First, define the **group** of 9 with 0=boy and 1=girl	Use [ctrl] [:=] to get := Use [ctrl] [)] to get **{ }**	1.1 *Doc DEG **group**:={0,0,0,0,1,1,1,1,1}
Select a team of size 3 from the **group**, <u>without</u> replacement. This requires a third argument of '1' in the **randsamp( )** command.	[menu] 5: Probability 4: Random 5: Sample Use [var] to get **group**	randsamp(**group**,3,1)
Count the number of girls in the team, using **sum(…)**	[menu] 6: Statistics 3: List Maths 5: Sum of elements	randSamp(*group*,3,1) {1,0,0} sum(randSamp(*group*,3,1)) 1
Repeat simulation 100 times	Use [book] [S] and ▼ to get **seq( )**	seq(sum(randSamp(*group*,3,1)),*n*,1,100) {1,0,3,1,1,2,2,1,2,2,0,2,2,2,3,2,0,1,3,1,1,2,2▸
Store the simulation ***results*** in a variable and count the frequency of all girl teams.	Use [book] [C] and ▼ to get **countIf( )**	*results*:=seq(sum(randSamp(*group*,3,1)),*n*,1,1▸ {1,1,1,3,1,3,2,2,2,2,3,3,1,1,2,2,2,2,2,1,2,1,1▸ countIf(*results*,3) 14

Our simulation suggests a 14% chance of creating an all-girl team of three.

Theoretical Notes

As this problem features withdrawal <u>without</u> replacement, it is <u>not</u> a Binomial distribution. It is actually a Hypergeometric Distribution with a population size of 9, with 5 success states in the population and 3 draws.

The probability of 'g' girls in the team of 3 is given by:

$$\frac{\mathrm{nCr}(5,g)\cdot \mathrm{nCr}(4,3-g)}{\mathrm{nCr}(9,3)}$$

And so in our original situation, we have g=3, giving a theoretical result of:

$$\left.\frac{\mathrm{nCr}(5,g)\cdot \mathrm{nCr}(4,3-g)}{\mathrm{nCr}(9,3)}\right|g=3 \qquad 0.119048$$

Extension/Variants

Consider a family of 7 people: 2 adults, 2 boys and 3 girls, and you randomly pick three people.
How likely is it that you have exactly one adult, one boy and one girl?

<u>Hint</u>: let 100=adult, 10=boy, 1=girl and set ***group***:={100,100,10,10,1,1,1}

Challenge Problem 4 – Rolling a Single Die & Hypothesis Testing

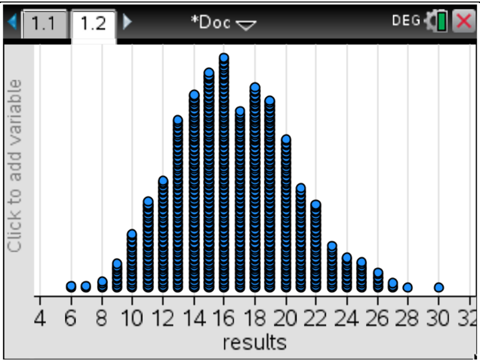
I rolled a D6 die 100 times and obtained the number four 28 times.
Is my die biased?

Task	Keypad Help	Screenshot
First, simulate just one roll of the dice by using the **randInt(…)** command	menu 5: Probability 4: Random 2: Integer	1.1 *Doc DEG randInt()
Specify integers from 1 to 6		1.1 *Doc DEG randInt(1,6)
Check it works as expected	Use enter several times	1.1 *Doc DEG randInt(1,6) 6 randInt(1,6) 1 randInt(1,6) 4
Now run it 100 times	Insert ,100 into the previous command	randInt(1,6,100) {3,5,1,3,6,2,5,6,2,3,1,6,1,1,4,6,6,2,2,1,1,5,1
Define a variable called **results** to store the simulated values.	Use ctrl to get := Use ▲ to highlight and enter to paste in previous lines.	randInt(1,6,100) {3,5,1,3,6,2,5,6,2,3,1,6,1,1,4,6,6,2,2,1,1,5,1 *results*:=randInt(1,6,100) {4,1,4,1,4,2,1,6,6,2,5,2,1,5,2,4,5,6,3,1,3,4,3
Count how many 4's happened	Use C and ▼ to get **countIf( )** Use var to get **results**	countIf(**results**,4)
In this case, we had 15 occurrences of the number 4 in 100 roles		*results*:=randInt(1,6,100) {4,1,4,1,4,2,1,6,6,2,5,2,1,5,2,4,5,6,3,1,3,4,3 countIf(*results*,4) 15
Combine these two processes into a single step	Use ▲ to highlight and enter to paste in previous lines.	countIf(randInt(1,6,100),4)
Check it works as expected!	Use enter several times	countIf(randInt(1,6,100),4) 17 countIf(randInt(1,6,100),4) 16 countIf(randInt(1,6,100),4) 18

We want to repeat this whole experiment of a hundred rolls, a 1000 times, counting the number of 4's each time.

We therefore need a way of automating the process to create a sequence of 1000 repetitions.

Use the sequence command, **seq(…)** to run it 1000 times	Use S and ▼ to get **seq( )** Use ▲ to highlight and enter to paste in previous lines.	seq(countIf(randInt(1,6,100),4),*n*,1,1000)
Store simulated **results**	Use ctrl to get :=	seq(countIf(randInt(1,6,100),4),*n*,1,1000) {17,23,17,22,23,21,19,19,19,12,23,24,14,19 **results**:=seq(countIf(randInt(1,6,100),4),*n*,1,

Task	Keypad Help	Screenshot
Plot the distribution of these **results**. We now look to see how likely it was to have got 28 rolls of a 4 (which was our experimental result)	ctrl doc▾ 5: Add Data & Statistics	
Count how many of the simulated results were 28 or more	Use 📖 C and ▼ to get **countIf( )** Use ?!▸ to get ? Use ctrl = to get ≥	countIf(*results*,?≥28) 2
Calculate the p-value	The **count(…)** command returns the total number of numerical results.	$\frac{\text{countIf}(results,?\geq 28)}{\text{count}(results)}$ 0.002

We conclude that the likelihood of obtaining 28 fours when rolling a die 100 times is in the most extreme 0.2% of the distribution of typical results. This appears very unlikely for a fair die, so we conclude from this simulation that it's likely that the real die that was used is biased towards the number 4.

Probability Theory Information

The distribution plots generated are simulations of a Binomial distribution where n = number of rolls and p = probability of rolling the specified number on the die.

X = number of 4's obtained with 100 rolls
$X \sim B(100,1/6)$
The theoretical chance of experiencing 28 occurrences of 4 from 100 rolls is given by
$P(X\geq 28) = 0.003101$

$$\text{binomCdf}\left(100,\frac{1}{6},28,100\right) \qquad 0.003101$$

Extension/Variants

Change the commands to run 2000 simulations of obtaining a score of 2 on a 8 sided die that's rolled 75 times.

Challenge Problem 5 – Rolling Two Fair Dice & Hypothesis Testing

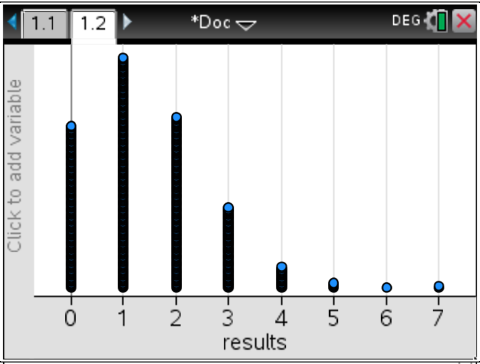
I rolled two D6 dice 50 times and noted their total score each time.
I obtained a total score of 12 only four times – is that to be expected?

Task	Keypad Help	Screenshot
First, simulate just one roll of a die by using the **randInt(…)** command	menu 5: Probability 4: Random 2: Integer	randInt()
Specify integers from 1 to 6		randInt(1,6)
Add two random integers together to simulate the total of two dice rolled	Use ▲ to highlight and enter to paste in previous lines.	randInt(1,6) 3 randInt(1,6)+randInt(1,6)
Use the sequence command, **seq(…)** to run it 50 times	Use 📖 S and ▼ to get **seq( )**	seq(randInt(1,6)+randInt(1,6),n,1,50)
Store simulated **results**	Use ctrl ⊞ to get :=	seq(randInt(1,6)+randInt(1,6),n,1,50) {12,6,8,6,10,8,3,10,7,2,10,11,7,10,11,9,6,3, results:=seq(randInt(1,6)+randInt(1,6),n,1,50)
Count how many 12's happened	Use 📖 C and ▼ to get **countIf( )** Use var to get **results**	countif(**results**,12)
In this case, we only had 1 occurrence of 12 in 50 roles of two dice.		results:=seq(randInt(1,6)+randInt(1,6),n,1,50) {9,7,6,8,7,6,9,5,7,7,8,9,5,5,5,9,8,6,10,7,7,5, countIf(results,12) 1
Combine these two processes into a single step	Use ▲ to highlight and enter to paste in previous lines.	countIf(seq(randInt(1,6)+randInt(1,6),n,1,50),12)
Check it works as expected!	Use enter several times	countIf(seq(randInt(1,6)+randInt(1,6),n,1,50),12) 1 countIf(seq(randInt(1,6)+randInt(1,6),n,1,50),12) 2 countIf(seq(randInt(1,6)+randInt(1,6),n,1,50),12) 0

We now repeat this whole experiment of fifty rolls 1000 times, counting the number of 12's each time.

We need a way of automating the process to create a sequence of 1000 repetitions.

Use the sequence command, **seq(…)** to run it 1000 times		seq(countIf(seq(randInt(1,6)+randInt(1,6),n,1,50),12),n,1,1000)
Store simulated **results**	Use ctrl ⊞ to get := Use ▲ to highlight and enter to paste in previous lines.	seq(countIf(seq(randInt(1,6)+randInt(1,6),n,1 {4,2,1,3,2,2,0,1,1,3,3,5,0,2,1,1,1,0,3,1,2,1,1 **results**:=seq(countIf(seq(randInt(1,6)+randIn

Task	Keypad Help	Screenshot
Plot the distribution of the **results**. We now look to see how likely it was to have recorded 4 rolls of 12 (which was our experimental result)	ctrl doc▾ 5: Add Data & Statistics	
Count how many of the simulated results were 4 or more	Use ?!▸ to get ? Use ctrl = to get ≥	$\text{countIf}(results,?\geq 4)$ 44
Calculate the p-value	The **count( )** function returns the total number of numerical results.	$\frac{\text{countIf}(results,?\geq 4)}{\text{count}(results)}$ 0.044

We conclude that the likelihood of obtaining 4 total scores of twelve when rolling two die 50 times is in the most extreme 4.4% of the distribution of typical results. If we consider 5% as a standard cutoff, this simulation suggests that the throws of the real dice that were used are not fair in some regard.

Probability Theory Information

This simulation requires summing of two Uniform Distributions (each U[1,6]) which gives rise to a Binomial Distribibution with n=50 and p=1/36 as we were focussing on a total score of 12. Had we been interested in another total score, p would have changed.

X = number of double sixes obtained with 50 rolls of two D6
X ~ B(50,1/36)
The theoretical chance of experiencing at least 4 occurrences of a double six from 50 rolls is given by P(X≥4) = 0.049947

$$\text{binomCdf}\left(50,\frac{1}{36},4,50\right) \quad 0.049947$$

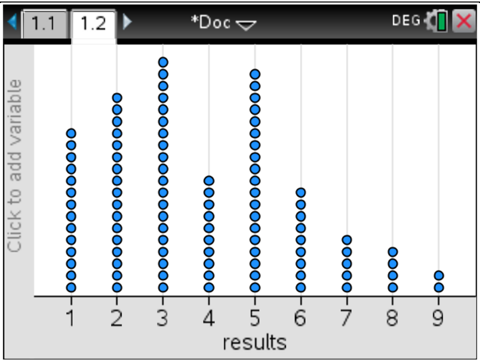
Extension/Variants

Change the commands to run 2000 simulations of obtaining a total score of 16 from rolling three six sided dice 80 times.

Challenge Problem 6 – Removing Discs from a Bag Without Replacement

Ten identically shaped discs are in a bag; two of them are black, the rest white.
Discs are drawn at random from the bag in turn and not replaced.
How many discs are expected to be drawn up to and including the first black one?

Task	Keypad Help	**Screenshot**
First, define a ***bag*** with the required discs where 0=white, 1=black	Use [ctrl] [templates] to get := Use [ctrl] [)] to get { }	1.1 *Doc DEG *bag*:={1,1,0,0,0,0,0,0,0,0} {1,1,0,0,0,0,0,0,0,0}
Create a list that knows the number of each draw.		*positions*:=seq(*n*,*n*,1,10) {1,2,3,4,5,6,7,8,9,10}
Define a function called ***draw(x)*** that picks all ten discs from the bag - <u>without</u> replacement - and returns the draw number of the black discs.	Note the variable ***x*** is a dummy input, that has no use. ***draw(x)*** returns the <u>**product**</u> of the random sample of 0's and 1's with the list of positions.	*draw*(*x*):=randSamp(*bag*,10,1)·*positions* *Done*
Check that it works! Here, our first test had the black discs on draws 7 and 8; the next on draws 6 and 10, and the last on draws 2 and 9.	Note the input of '**1**' in the ***draw(..)*** function is irrelevant. It could have been any number.	*draw*(1) {0,0,0,0,0,0,7,8,0,0} *draw*(1) {0,0,0,0,0,6,0,0,0,10} *draw*(1) {0,2,0,0,0,0,0,0,9,0}
Define a function that returns the ***first_non_zero*** element of the output of the ***draw(x)*** function. We can't use the minimum function here, as it returns the value of zero.	Use [ctrl] [space] to get underscore _ Use [menu] 6: Statistics 3: List Maths to get 5: Sum of Elements and 2: Maximum	*first_non_zero*(*x*):=sum(*x*)−max(*x*) *Done*
Check that it works!	Use [enter] several times	*first_non_zero*(*draw*(1)) 3 *first_non_zero*(*draw*(1)) 1 *first_non_zero*(*draw*(1)) 6
Repeat for 100 experiments Store the results	Use [catalog] [S] and ▼ to get **seq( )**	seq(*first_non_zero*(*draw*(1)),*n*,1,100) {8,6,2,6,2,2,3,1,8,2,1,3,5,7,1,5,2,2,7,5,2,4,3▸ *results*:=seq(*first_non_zero*(*draw*(1)),*n*,1,100) {2,6,4,6,6,5,8,5,3,5,9,5,3,2,6,1,2,2,4,2,2,3,7▸
Calculate the expected number of draws until the first black disc	[menu] 6: Statistics 3: List Maths 3: Mean	mean(*results*) 3.82

Task	Keypad Help	Screenshot
And you can view the distribution of number of discs until the first black disc.	ctrl doc▾ 5: Add Data & Statistics	

Probability Theory Information

This is a variant on a Geometric Distribution, but different in that the process is finite as there are only 10 discs in the bag. Also, the chance of a success changes after each trial as it is withdrawal without replacement. So, it's not like the Geometric Distribution at all……

X = number of discs drawn until first black disc

x	1	2	3	4	…	n	…	…	9
P(X=x)	$\frac{2}{10}$	$\frac{8}{10}\cdot\frac{2}{9}$	$\frac{8}{10}\cdot\frac{7}{9}\cdot\frac{2}{8}$	$\frac{8}{10}\cdot\frac{7}{9}\cdot\frac{6}{8}\cdot\frac{2}{7}$		$\frac{\frac{8!}{(9-n)!}\cdot 2}{\frac{10!}{(10-n)!}}$			$\frac{8!\cdot 2}{10!}$

Giving:

$$E(X) = 3\tfrac{2}{3}$$

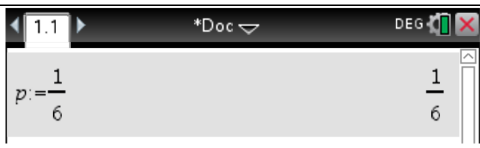
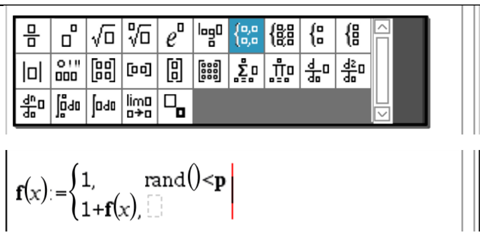
Extension/Variants

i. Adapt the simulation so that you record the number of discs drawn until <u>both</u> black discs are found.

ii. Adapt it again for a bag of 3 black and 7 white discs, and record when you've drawn out all <u>three</u> black discs.

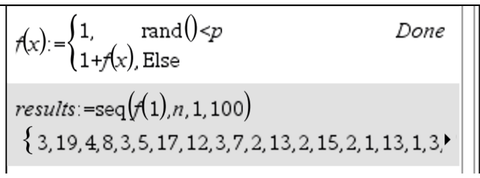
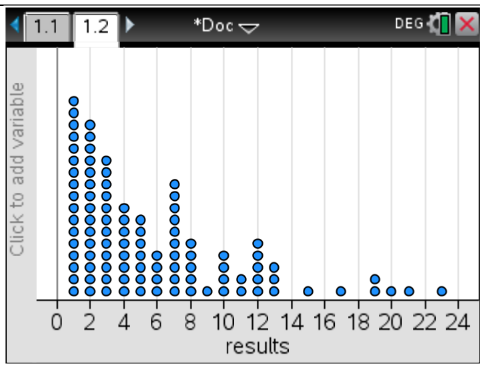
Challenge Problem 7 – Simulating a Geometric Distribution

How many rolls of a dice do you expect to roll before the first 6?

Task	Keypad Help	Screenshot
Define ***p***, the probability of rolling a 6 on a standard die.	Use ctrl to get :=	
Define ***f(x)*** to be a piecewise recursive function. Note the variable '***x***' is a dummy input, that has no use.	Use to get piecewise function template Use menu 5: Probability 4: Random 1: Number to get **rand( )**	

How it works....

The function ***f(x)*** first generates a random number between 0 and 1.
If it is less than p, then a 'success' has happened and the function returns a '1'
If the random number is not less than p, it calls itself adding 1 to its value.
Hence calling this function with any dummy input value will return the number of times it had to generate a random number until the first success.

Store the results of 100 experiments	Use S and ▼ to get seq()	
Calculate the expected number of rolls until the first six.	menu 6: Statistics 3: List Maths 3: Mean	
And you can view the distribution of number of rolls until the first six.	ctrl doc▾ 5: Add Data & Statistics	

Probability Theory Information

This problem is an example of a Geometric Distribution with parameter $p = 1/6$.
It differs from a Binomial Distribution as there is no fixed number of trials as it only terminates after the first success. Theory predicts that the expected number of trials until the first success is $1/p$, which is 6 in this case. You can also verify whether the standard deviation of the simulated sample matches the theoretical value of $(1-p)/p$

Extension/Variants

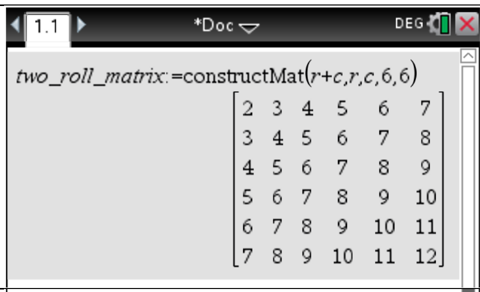
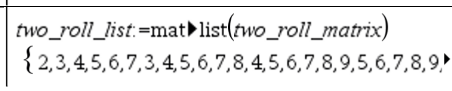
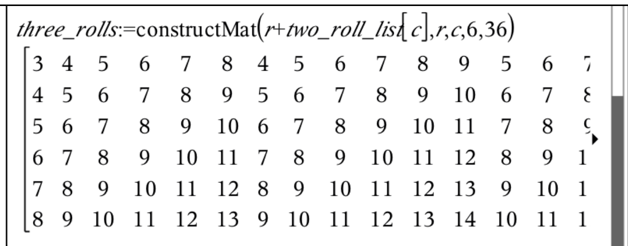
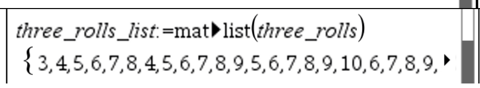
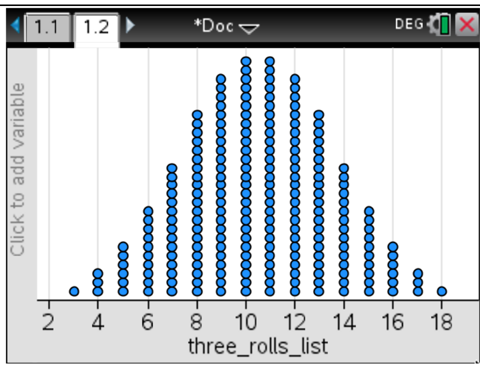
The Negative Binomial distribution counts the number of trials until the k^{th} success, not just the first success.

i. Adapt the simulation's code so that it counts the number of trials until the second 6.
ii. Then have it count the nmber of trials until the third 6.

Challenge Problem 8 – Generating Outcome Tables

How to create the complete outcome table for the sum of three D6 dice.

Instead of simulating an experiment many times to obtain accurate results, you can generate the full theoretical outcome tables, and then take random samples from it.

Task	Keypad Help	Screenshot
First, define the outcome table for the sum of two D6. Call it ***two_roll_matrix***	Use ctrl ␣ to get underscore _ menu 7: Matrix & Vector 1: Create A: Construct Matrix	
Define a list called ***two_roll_list*** that's the ***two_roll_matrix*** in one long list, row by row.	menu 6: Statistics 4: List Operations 9: Convert Matrix to List	
Define a new matrix called ***three_rolls*** that combines each element of the ***two_roll_list*** with a roll of a single third D6 die.		
Define a list called ***three_rolls_list*** that's the ***three_rolls*** matrix in one long list, row by row.	menu 6: Statistics 4: List Operations 9: Convert Matrix to List	
View the theoretical distribution of the sum of scores on three D6. You could now perform **randSamp(…)** commands on the ***three_rolls_list*** to simulate multiple rolls of three D6.	ctrl doc▾ 5: Add Data & Statistics	

Probability Theory Information

You will notice that ***two_roll_list*** should match the simulation results from Problem 9. If this problem only had two dice, it would have been an outcome table with rows and columns – a 2D table. As it had three dice, we would view it as a 3D table. This is hard to draw!

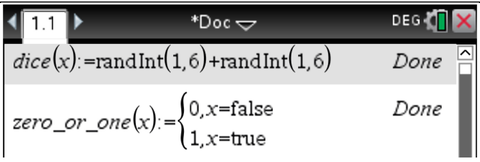
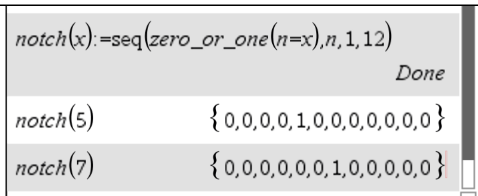
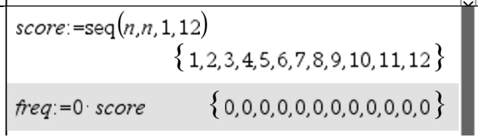
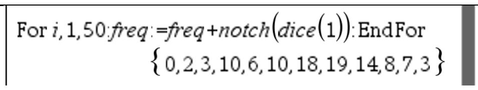
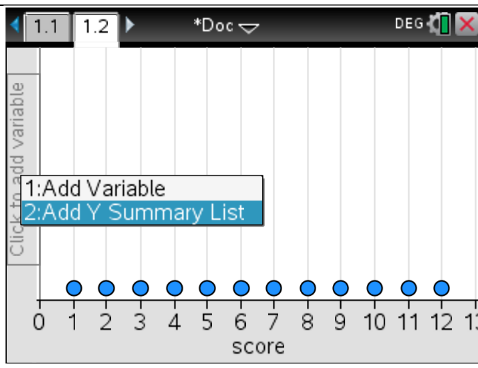
Extension/Variants

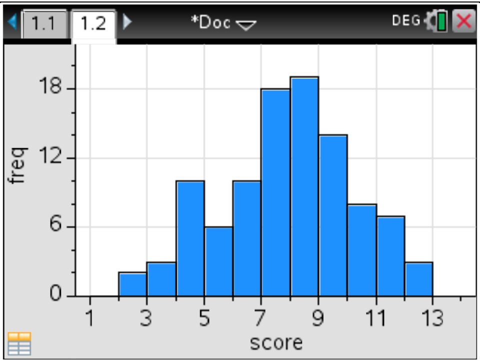
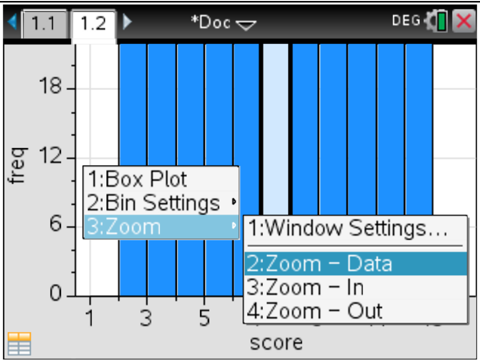
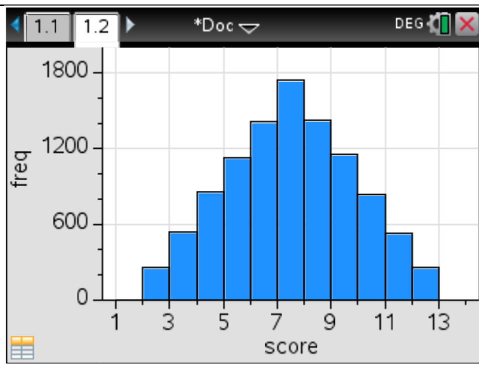
Consider the distribution of the sum of a D4, a D6 and a D8.
This will have a minimum score of 3 and a maximum score of 18, just like rolling three D6.
However, is it a symmetrical distribution like the one for the total score of three D6?

Challenge Problem 9 – Capturing Results in a Frequency Table

How to store the results of rolling two dice 10,000 times and creating a summary plot

The maximum number of elements per list is typically 2500 elements, so if you wish to run larger simulations you need a technique to capture the results in a frequency table as they are generated, and not simply store the raw results. This is demonstrated in the following example.

Task	Keypad Help	Screenshot
Define a function called ***dice(x)*** that simulates the sum of two fair D6 dice. Define a piecewise function called ***zero_or_one(x)*** that returns numerical values if the input is either 'true' or 'false'.	Use [ctrl] [␣] to get underscore _ Use [template key] to get piecewise function template	
Define a function called ***notch(x)*** that returns a list with '1' in the x^{th} position.	Use [var] to get ***zero_or_one()***	
Define ***score*** to be the list of outcome totals, (including the not-possible total score of 1). Define the ***freq*** list ready to keep a record of the cumulative frequencies of each score.		
Before we run 10,000 simulations, we shall make sure that it works for just 50. Once that is done, we shall update the code to run it for 10,000 times.		
Construct a **For** loop to run 50 times. Note that ***i*** is the loop variable, that could be any letter.	[menu] 9: Functions & Programs 5: Control 5: For…EndFor Use [?!▸] to get colon :	
The output are the frequencies of each ***score*** obtained, stored in ***freq*** Note the 0 in the first element, showing that it's not possible to obtain a total score of 1 from rolling two D6.		
View the summary data	[ctrl] [doc▾] 5: Add Data & Statistics Add ***score*** variable then Press [ctrl] [menu] when over vertical axis, and select Add Y Summary List or [menu] 2: Plot Properties 9: Add Y Summary List	

Task	Keypad Help	Screenshot
Add the ***freq*** variable as the Y Summary List		
Now that the simulation is constructed, edit the **For…EndFor** loop to run from 1 to 10000. Then, sit back and wait….	To rescale the axes, press ctrl menu and select 3: Zoom 2: Zoom - Data	
Lovely and symmetrical! Sample size saves the day.		

Probability Theory Information

You can generate the sample space of all the outcomes (below) and compare its frequencies to the simulation's results.

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

This table can also be created on the TI-Nspire – see ***two_roll_matrix*** in Challenge Problem 8.

Extension/Variants

Change the simulation to model 10,000 repeats of the product of two D6 die.

Challenge Problem 10 – Comparing Two Independent Distributions

Anne rolls three D4 dice and Bob rolls two D6 dice.
What is the Probability that Anne's total is greater than Bob's total?

Task | **Screenshot**

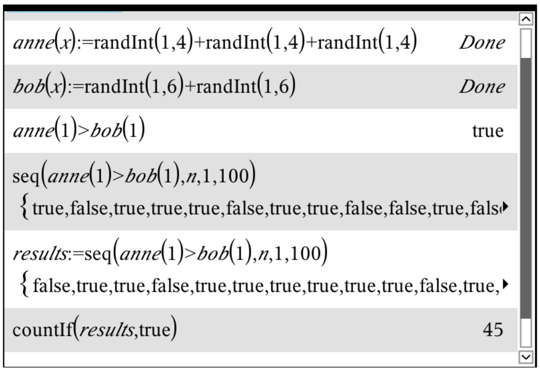
Method 1 – Using true & false

Define functions called ***anne(x)*** and ***bob(x)*** that simulate each of their dice rolls.

Simulate one roll of Anne's and Bob's die.
Note that the input value of 1 has no relevance.

Store the results of 100 experiments.
Count how many times Anne score more than Bob.

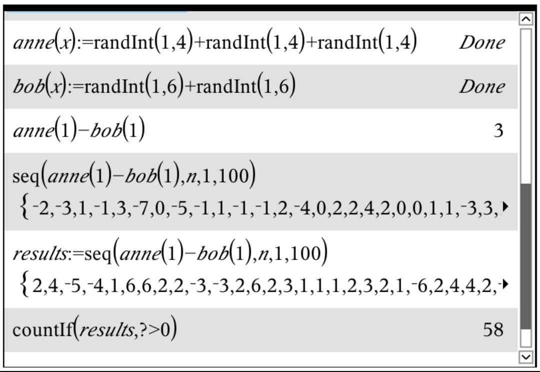
So this simulation suggests that the probability of Anne scoring more than Bob is 0.45



Method 2 – Using positive & negative

This is very similar to Method 1, but instead of recording a true/false flag, we record how much more Anne's score was than Bob's. We then look for how many times it was a positive score.
Method 2 gives access to the distribution of score 'differences', whereas Method 1 does not.

This simulation suggests that the probability of Anne scoring more than Bob is 0.58



Probability Theory Information

This problem is very challenging to analyse theoretically, as it requires all of the combinations from Anne and Bob's dice to be tabulated and their respective probabilities taken into account.
A more accurate simulation result would come from increasing the number of repetitions from 100.
Task: Design a simulation to run this experiment 10,000 times, storing the results in a frequency table (see Problem 9)

Extension/Variants

i. What if Anne sums a D4 and a D8, whilst Bob remains rolling two D6?
ii. What if Anne rolls one D4 and trebles its score, whilst Bob remains rolling two D6?
iii. What are the chances of them obtaining the same score?