Fixed-point free permutations

"Run out of money at the bar ... let your friend pay!" writes Greg Gutfeld in Men's Health magazine.
Have him shuffle two decks of cards and lay them side by side. Explain that you always take one card from each deck from the top at the same time and bet that a pair of identical cards eventually appear.
We encounter this paradoxical process in a variety of guises (secretary problem, mixed-up letters problem...).

In class, after a well-chosen introductory example, different approaches are possible:

Each students of a pair of students writes down the numbers from 1-10 in a randomly chosen way and checks if there is at least one "fixed point". If there are enough pairs of students per class, a tendency should already be recognizable. Subsequently, an investigation for small “n” offers itself. If you note the fixed-point free permutations, you could perhaps consider the cases of:

n = 2 21,

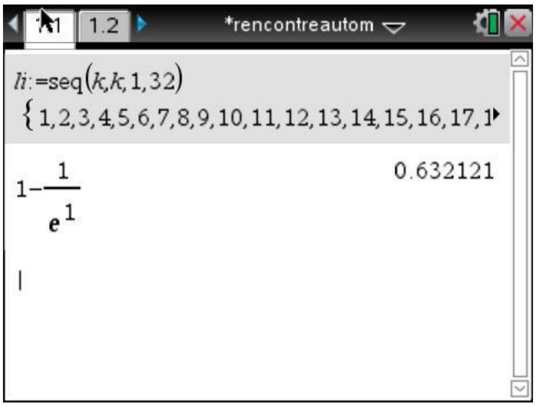
n= 3 231 312,

n = 4 2143 2341 2413 3142 3412 3421 4123 4312 4321.

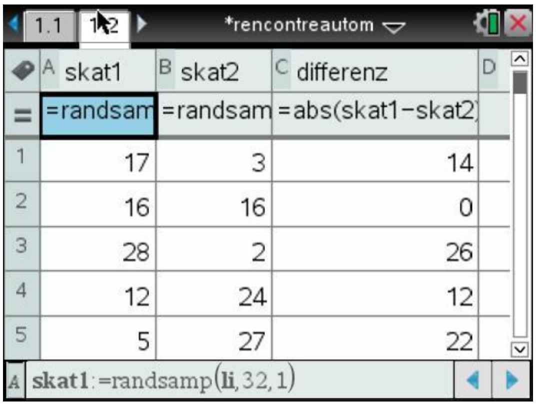
To what extent you derive the formula $P(n)=1-\frac{1}{2!}+\frac{1}{3!}-\frac{1}{4!}+\frac{1}{5!}...$ should be left to the class situation. It is at least interesting that from n = 4 the probability for at least one fixed point remains stable $\approx 1-\frac{1}{e}=0,6321$.
The actual paradox is not that the probability for at least one fixed point seems to be relatively high, but that these probabilities do not differ any more from n = 6 up to four decimal places.
Before using a program, you can first run various "program-free" simulations. The command randsamp(list, selection, condition) is very useful for this.

| | |
|---|---|
| $l:=\text{seq}(t,t,1,10)$ | $\{1,2,3,4,5,6,7,8,9,10\}$ |
| $\text{randSamp}(l,10,1)$ | $\{5,10,9,3,7,8,6,2,4,1\}$ |
| $\text{randSamp}(l,10,1)$ | $\{3,4,6,10,5,7,8,9,2,1\}$ |
| $\text{randSamp}(l,10,1)$ | $\{2,4,7,6,8,9,5,3,1,10\}$ |

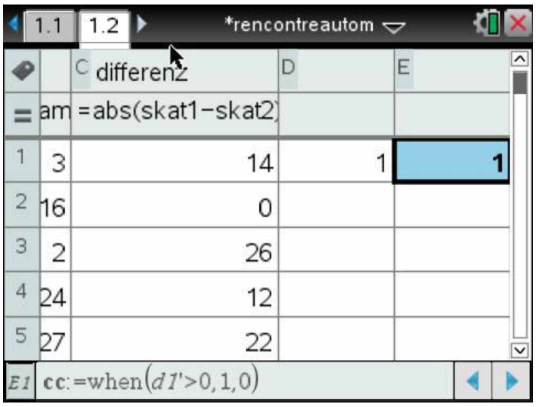
For larger “n” the use of lists and the spreadsheet is recommended.




Definition of a list with 32 elements, ascending from 1 to 32. The list thus corresponds to a deck of cards.




Model drawing from both decks of cards 32 times (without putting them back) and find the difference.

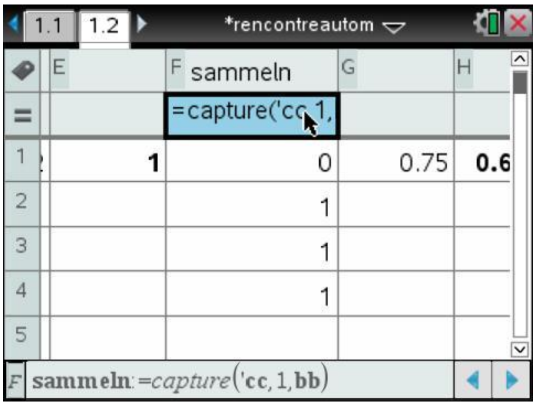



Count the matches in cell d1 with

= countif(difference,?=0)

In cell e1 a variable cc is defined, that writes a "1" into the cell, if there was at least one match in one of the performed experiments (= 32 times drawing without putting back).

This then makes it possible to evaluate several such experiments via the capture command.




In cell g1, the formula

= sum(collect)/dim(collect) determines the proportion of experiments that are not fixed-point free. Multiplication by 1. yields a decimal number.

However, a small trick is necessary for the capture command - with the variable bb in cell h1 a random number is generated by = rand(), which overwrites the "memory" of the capture command and also counts successive equal measured values.

Another Approach

| | |
|---|---|
| $grundmenge(n):=\text{seq}(k,k,1,n)$ | *Fertig* |
| $perm(n):=\text{randSamp}(grundmenge(n),n,1)$ | *Fertig* |
| $perm(5)$ | $\{5,4,2,1,3\}$ |
| $perm(5)$ | $\{3,2,5,1,4\}$ |
| $\{5,4,2,1,3\}-\{3,2,5,1,4\}$ | $\{2,2,^{-}3,0,^{-}1\}$ |
| $\text{countIf}(\{2,2,^{-}3,0,^{-}1\},0)$ | 1 |
| $\text{seq}(\text{countIf}(perm(5)-perm(5),0),k,1,100)$ | |
| $\{3,1,1,1,2,1,1,1,0,0,0,1,2,0,0,0,1,0,0,2,1,0,0,0,1,1,1,1,0,1,0,0,1,0\ldots$ | |
| $\dfrac{\text{countIf}(\text{seq}(\text{countIf}(perm(5)-perm(5),0),k,1,1000),?>0)}{1000}$ | 0.637 |

Yet another one – using a basic or python program

```
Define LibPub durche(n,liste)=
Func
Local i,l,t,z,l1,l2
z:=0
l:=seq(t,t,1,liste)
For i,1,n
l1:=randSamp(l,liste,1)
l2:=randSamp(l,liste,1)
If countIf(l2-l1,0)>0 Then
z:=z+1
EndIf
EndFor
1.·z
 n
EndFunc
```

```
1.1  1.2  rencontre…on1  RAD
rencontre_python1.py  1/30
from random import *
from math import exp

def perm(n):
  p = []
  while len(p) < n:
    a =randint(1, n)
    if not(a in p): p.append(a)
  return(p)

def permdiff(n):
```