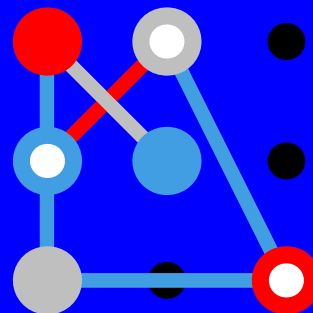


**Lua,
ConTeXt,
MetaPost**



**and bash (and its colleagues) can
help someone conquer their own phones**

A short etude for 'Composed thoughts' conference

Tomáš Hála (and Jakub Hála)

BachoTeX, 2024

0.03448275862069

What has happened

0.068965517241379

What has happened

new phone

new gesture after a few month

not fully charged off

forgotten the new gesture

crisis...

A bit of math

0.13793103448276

A bit of math – permutations

Assume a gesture consisting just 9 points

$$N = n!$$

for nine values ($n = 9$)

$$N = 9! = 362\,880$$

A bit of math – permutations

If we spent
5.235987755983...
seconds for one attempt only,
we need exactly
 π
weeks (assuming 24/7 mode).

A bit of math – the worst situation

n	$n!$	$v = \frac{n!}{(n-k)!}$	$\sum_1^n v$
1	1	9	9
2	2	72	81
3	6	504	585
4	24	3024	3609
5	120	15120	18729
6	720	60480	79209
7	5040	181440	260649
8	40320	362880	623529
9	362880	362880	986409

0.24137931034483

A bit of math – the worst situation

If we assume

$$\sum_1^n v$$

attempts, we need

$$\approx e \cdot \pi$$

weeks (assuming 24/7 mode).

A bit of Lua – generating permutations

```
function permgen (a, n)
  if n == 0 then coroutine.yield(a)
  else
    for i = 1, n, 1 do
      -- put i-th element as the last one
      a[n], a[i] = a[i], a[n]
      -- generate all permutations of the other elements
      permgen(a, n - 1)
      -- restore i-th element
      a[n], a[i] = a[i], a[n]
    end
  end end

function printResult (a)
  for i,v in ipairs(a) do io.write(v, " ") end
  io.write("\n")
end
```

A bit of Lua – generating permutations

```
function perm (a)
  local n = #a
  local co = coroutine.create(
    function () permgen(a, n) end )
  return function () -- iterator
    local status, res = coroutine.resume(co)
    return res
  end
end

for p in perm({"1", "2", "3", "4", "5", "6",
              "7", "8", "9"}) do
  printResult(p)
end
```

(From <https://www.lua.org/pil/9.3.html>)

Reducing the problem...

0.37931034482759

A bit of bash et al. – reducing the problem

(1) partial permutations (variations) 4 of 9

```
lua perm2.lua | cut -d' ' -f1-4 | sort -u
```

3 024

0.41379310344828

A bit of bash et al. – reducing the problem

(1) partial permutations (variations) – 4 of 9

(2) not started at the corner

```
lua perm2.lua | cut -d' ' -f1-4 | sort -u |  
egrep -v "(1 [379])|(3 [179])|(7 [139])|(9 [137])"
```

1800

0.44827586206897

A bit of bash et al. – reducing the problem

- (1) partial permutations (variations) – 4 of 9
- (2) not started at the corner
- (3) must be from the corner to the opposite-side middle

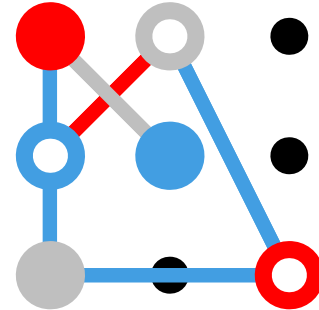
```
lua perm2.lua | cut -d' ' -f1-4 | sort -u |  
egrep -v "(1 [379])|(3 [179])|(7 [139])|(9 [137])" |  
egrep "(1 [68])|(3 [48])|(7 [26])|(9 [24])"
```

Output and How to present the output

1 2 3 4	1 2 6 3	1 2 8 9
1 2 3 5	1 2 6 4	1 2 9 3
1 2 3 6	1 2 6 5	1 2 9 4
1 2 3 7	1 2 6 7	1 2 9 5
1 2 3 8	1 2 6 8	1 2 9 6
1 2 3 9	1 2 6 9	1 2 9 7
1 2 4 3	1 2 7 3	1 2 9 8
1 2 4 5	1 2 7 4	1 3 2 4
1 2 4 6	1 2 7 5	1 3 2 5
1 2 4 7	1 2 7 6	1 3 2 6
1 2 4 8	1 2 7 8	1 3 2 7
1 2 4 9	1 2 7 9	1 3 2 8
1 2 5 3	1 2 8 3	1 3 2 9
1 2 5 4	1 2 8 4	1 3 4 2
1 2 5 6	1 2 8 5	1 3 4 5
1 2 5 8	1 2 8 6	1 3 4 6
1 2 5 9	1 2 8 7	1 3 4 7 ...

Output and How to present the output

1 2 3 4	1 2 6 3	1 2 8 9
1 2 3 5	1 2 6 4	1 2 9 3
1 2 3 6	1 2 6 5	1 2 9 4
1 2 3 7	1 2 6 7	1 2 9 5
1 2 3 8	1 2 6 8	1 2 9 6
1 2 3 9	1 2 6 9	1 2 9 7
1 2 4 3	1 2 7 3	1 2 9 8
1 2 4 5	1 2 7 4	1 3 2 4
1 2 4 6	1 2 7 5	1 3 2 5
1 2 4 7	1 2 7 6	1 3 2 6
1 2 4 8	1 2 7 8	1 3 2 7
1 2 4 9	1 2 7 9	1 3 2 8
1 2 5 3	1 2 8 3	1 3 2 9
1 2 5 4	1 2 8 4	1 3 4 2
1 2 5 6	1 2 8 5	1 3 4 5
1 2 5 8	1 2 8 6	1 3 4 6
1 2 5 9	1 2 8 7	1 3 4 7 ...



0.55172413793103

A bit of MetaPost

```
pickup pencircle scaled 3dd;
for i=0 upto 2:
  for j=0 upto 2: drawdot (i*u,j*u); endfor
endfor
pickup pencircle scaled 1dd;

z0=(0,2.75u);

z1=(0,2u);          z2=(1u,2u);          z3=(2u,2u);
z4=(0,1u)           z5=(1u,1u);          z6=(2u,1u);
z7=(0,0);           z8=(1u,0);           z9=(2u,0);

draw z\za--z\zb--z\zc--z\zd withcolor \mycolor;
pickup pencircle scaled 5dd;
drawdot (z\za) withcolor red;
```

A bit of ConT_EXt

```
\setuplayout[.....]
```

```
\def\mycolor{blue}
```

```
\def\mystartcolor{red}
```

```
\ssx % font size
```

```
\starttext
```

```
\startMPcode    u=.875cc;    \stopMPcode
```

```
\startuseMPgraphic{gesto}
```

```
% the code has been already seen
```

```
\stopuseMPgraphic
```

```
\startcolumns[n=10]
```

```
\startluacode
```

A bit of Lua in ConTeXt

```
ctx = context
```

```
for ni=1,632 do
```

```
  local a,b,c,d = io.read("*n","*n","*n","*n")
```

```
  ctx("\\def\\za{".a.."}")
```

```
  ctx("\\def\\zb{".b.."}")
```

```
  ctx("\\def\\zc{".c.."}")
```

```
  ctx("\\def\\zd{".d.."}")
```

```
  ctx("\\def\\ni{".ni.."}")
```

```
  ctx("\\def\\popis{".ni..  
    "\\crlf ".a..",".b..",".c..",".d.."}")
```

```
  ctx("\\framedtext[width=3.25cc,offset=2dd]{".  
    "\\popis\\crlf"}")
```

```
  ctx("\\useMPgraphic{gesto}")
```

```
  ctx("}")
```

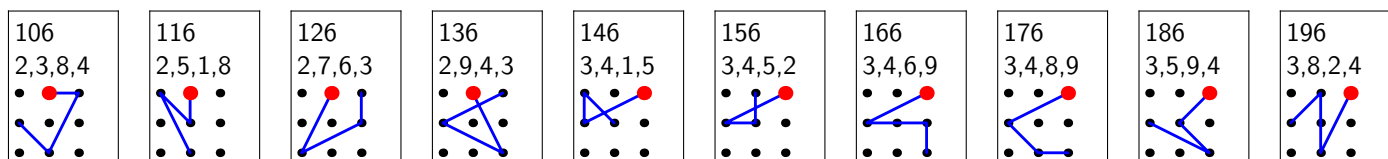
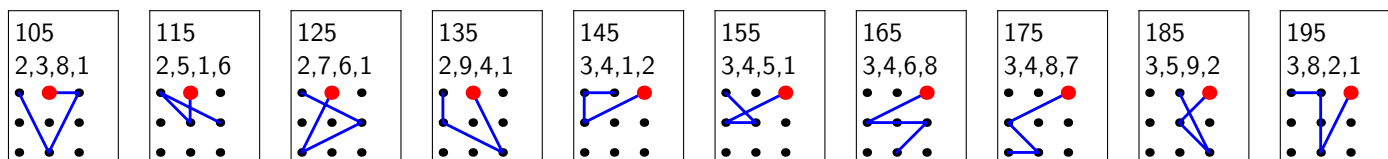
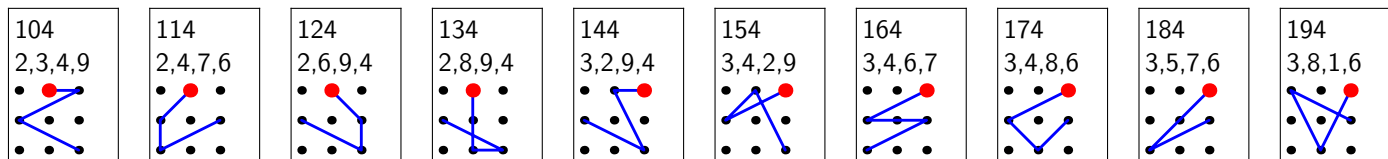
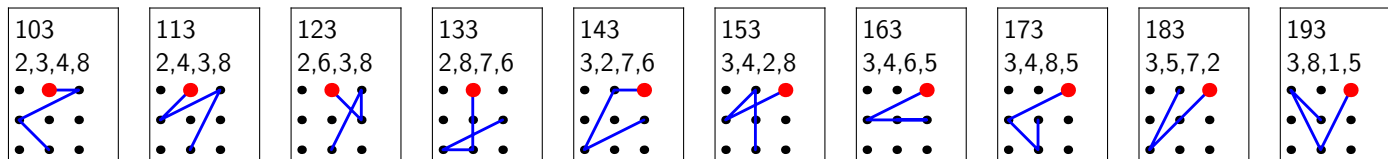
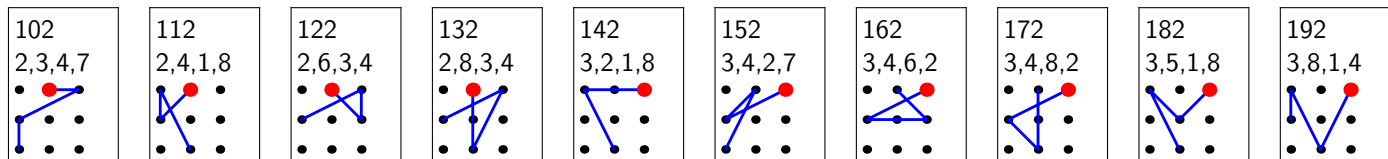
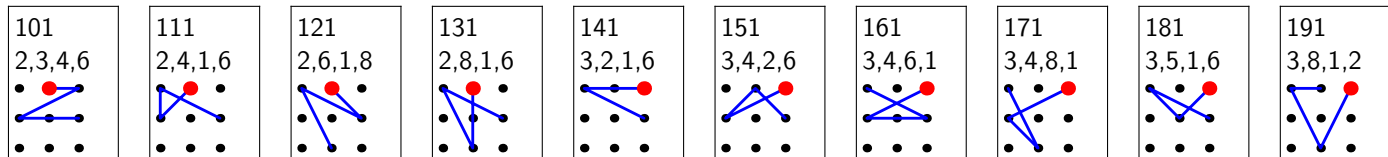
```
end
```

A bit of ConT_EXt (ctd.)

`\stopluacode`

`\stopcolumns`

`\stoptext`



Reducing the problem... (Part #2)

0.75862068965517

Reducing the problem... (Part #2)

(4) The gesture must contain a loop

Not implemented, solved manually.

Epilogue

0.82758620689655

A bit of math once more

After each
five attempts
the
'antihacker penalty pause'
increases
twice.

A bit of math once more

$$N_A = 205$$

$$N_G = \frac{205}{5} = 41$$

$$t_0 = 2^{-2}$$

$$Fort_{\text{AHP}_{G_A}} = t_0 \cdot 2^{G_A} = 2^{39}$$

R.I.P.

DATA

Instead of conclusion...

Using the Math:

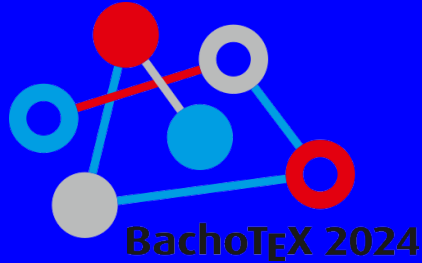
Kernings in Schola (Regular) Math?

Writing the presentation:

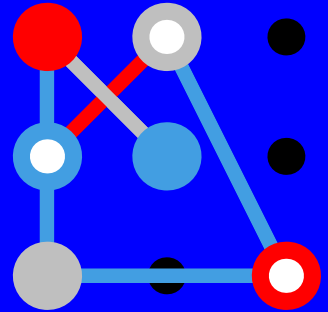
No manual for writing math for (Czech) students...

The main finding:

Never forget your secure gesture!



**Lua,
ConTeXt,
MetaPost**



**and bash (and its colleagues) can
help someone conquer their own phones**

A short etude for ‘Composed thoughts’ conference

Tomáš Hála (and Jakub Hála)

BachTeX, 2024