letterjongg!


bok laca

Not logged in Talk Contributions Create account Log in

Article Talk Read Edit View history Search Wikipedia

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Current events
Random article
About Wikipedia
Contact us
Donate

Contribute
Help
Learn to edit
Community portal
Recent changes
Upload file

Tools
What links here
Related changes
Special pages
Permanent link
Page information
Cite this page
Wikidata item

Print/export
Download as PDF
Printable version

In other projects
Wikimedia Commons

Languages
한국어
Italiano
Magyar
Polski
Русский
中文
Edit links

Mahjong solitaire

From Wikipedia, the free encyclopedia

This article is about the card-matching solitaire game. For the Chinese tile-based game, see Mahjong. For other uses, see Mahjong (disambiguation).

Mahjong solitaire (also known as **Shanghai solitaire**, **electronic** or **computerized mahjong**, **solitaire mahjong** or simply **mahjong**) is a single-player matching game that uses a set of mahjong tiles rather than cards. It is more commonly played on a computer than as a physical tabletop game.

Its name comes from the four-player game mahjong, but it is played entirely differently.

A Shanghai solitaire videogame arranged in "turtle formation"

Contents [hide]

Play [edit]

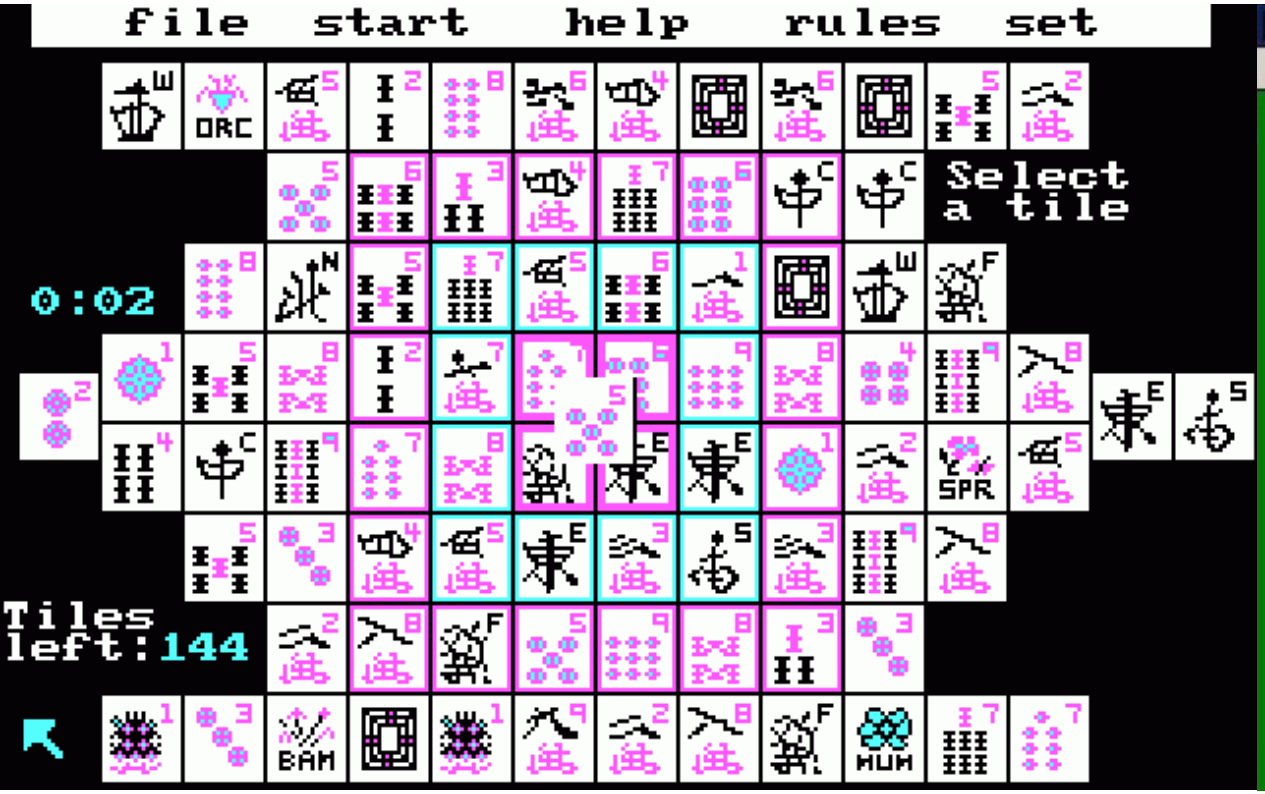
The 144 tiles are arranged in a special four-layer pattern with their faces upwards. A tile is said to be open or exposed if it can be moved either left or right without disturbing other tiles. The goal is to match open pairs of identical tiles and remove them from the board, exposing the tiles under them for play. The game is finished when all pairs of tiles have been removed from the board or when there are no exposed pairs remaining.

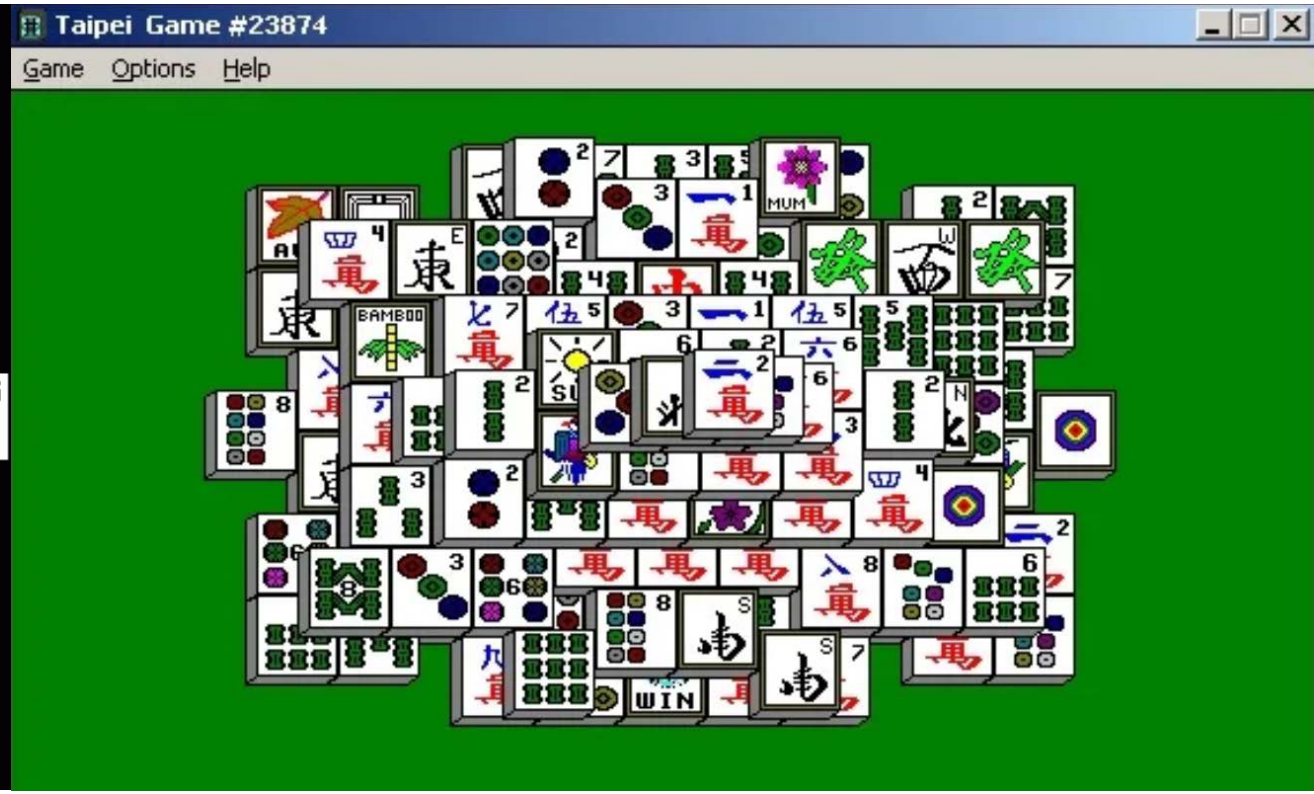
An untouched turtle formation, with exposed "free" tiles highlighted in blue

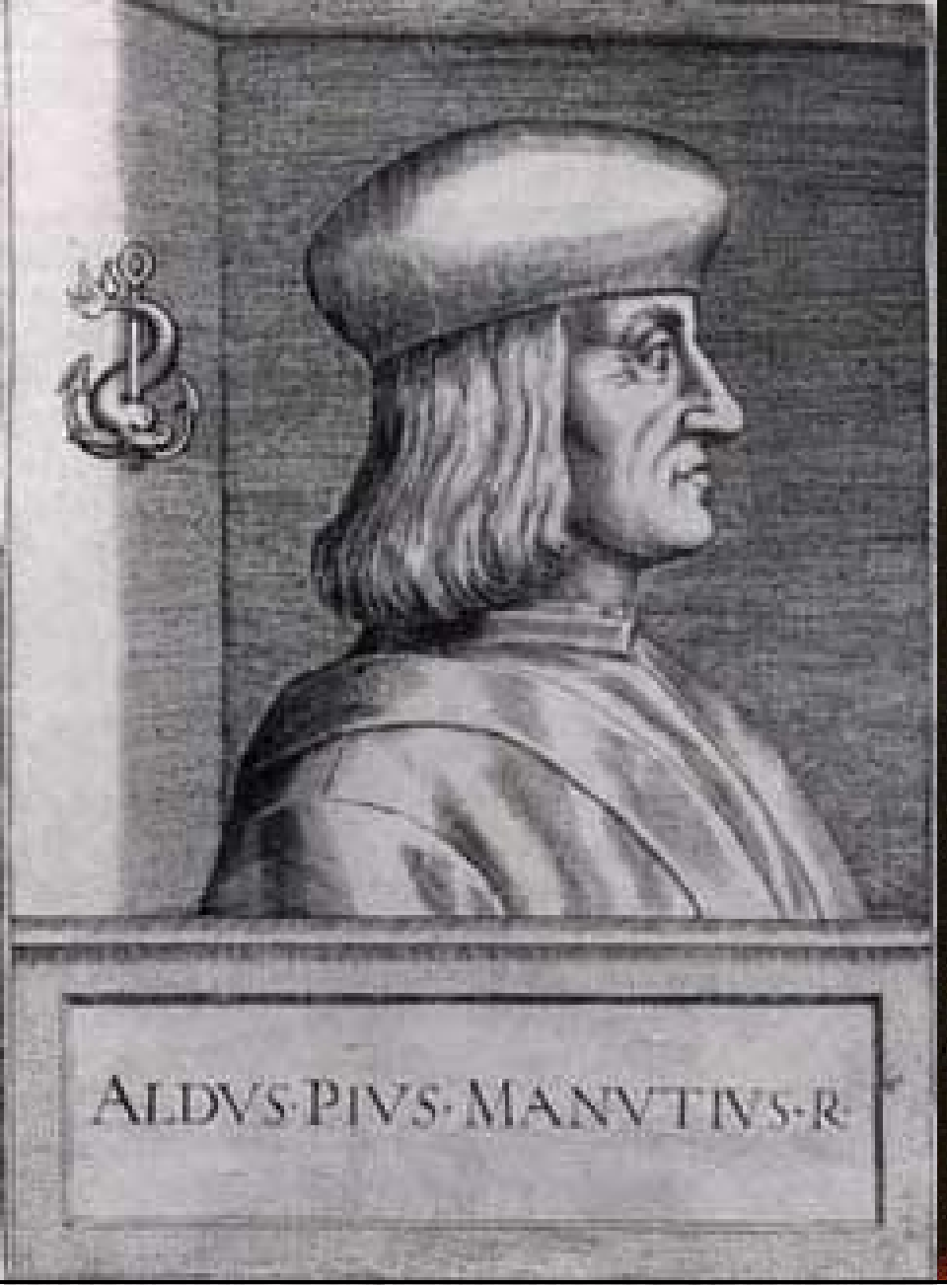
Tiles that are below other tiles cannot be seen. But by repeated undos or restarts which some programs offer, one gradually gets more and more information. Sometimes, tiles are only partially covered by other tiles, and the extent to which such tiles can be distinguished depends on the actual tile set.

Mathematical analysis [edit]

Playing Mahjong solitaire optimally in the sense to maximize the probability of removing all tiles is PSPACE-complete, and the game gets NP-complete when looking below tiles is allowed.[1] It has been proven that it is PSPACE-hard to approximate the maximum probability of removing all tiles within a factor of n^{ϵ}, assuming that there are arbitrarily many quadruples of matching tiles and that the hidden tiles are uniformly distributed.[2] The perfect-information version of


file start help rules set
Select a tile
0:02
Tiles left:144


Taipei Game #23874
Game Options Help

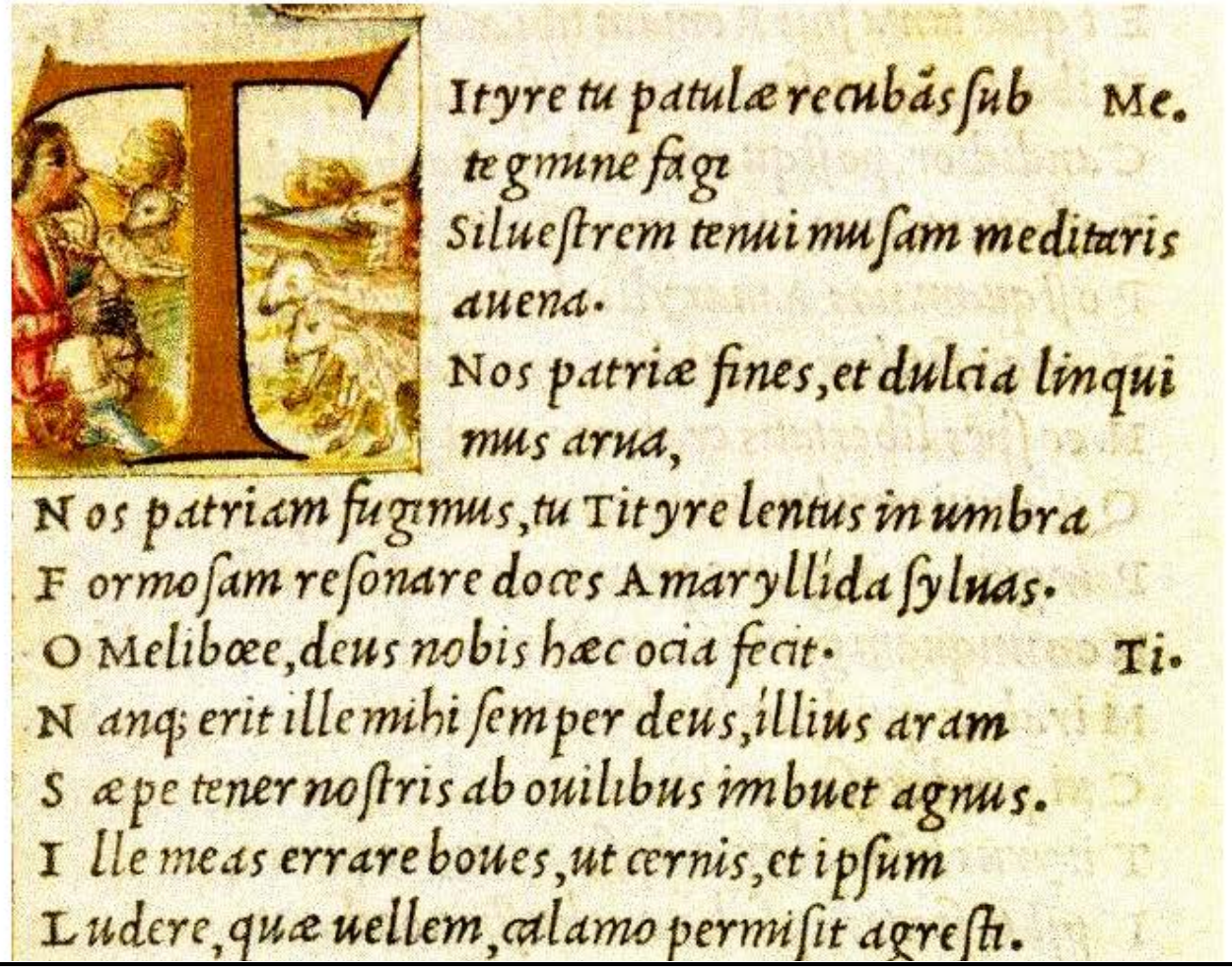


File:Virgil 1501 Aldus Manutius (Italika).jpg

Wikimedia Commons auf Deutsch

From Wikimedia Commons, the free media repository

File | File history | File usage on Commons | File usage on other wikis



Ityre tu patulæ recubãs ſub Me.
tegmine fagi
Silueſtrem tenui muſam meditaris
auena.
Nos patriæ fines, et dulcia linqui
mus arua,
Nos patriam fugimus, tu Tityre lentus in umbra
Formoſam reſonare doces Amaryllida ſylvas.
O Melibœe, deus nobis hæc ocia fecit. Ti.
Nanq; erit ille mihi ſemper deus, illius aram
Sæpe tener noſtris ab ouilibus imbuet agnus.
Ille meas errare boues, ut cernis, et ipſum
Ludere, quæ uellem, calamo permiſit agreſti.

Download all sizes

Use this file on the web

Use this file on a wiki

Email a link to this file

Information about reusing

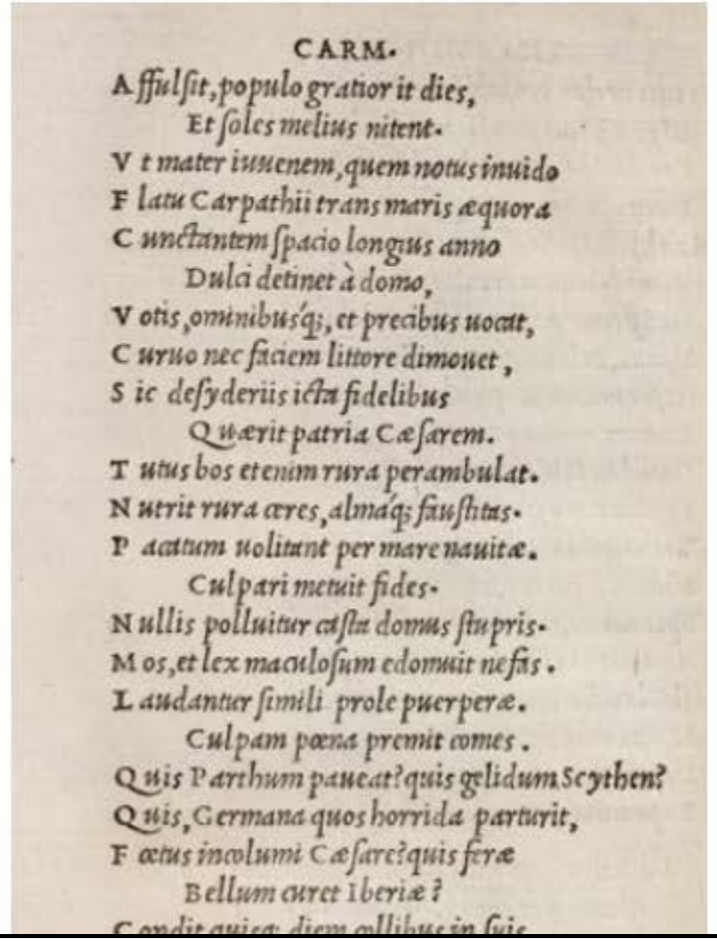
deus nobis hæc oda fecit.
lle mihi semper deus, illius ara
nostris ab ouilibus imbuet agr
errare boues, ut cernis, et ipsum
æ uellem, calamo permisit agre

File:Aldus Manutius, Horatius Flaccus, Opera, Venedig 1501.tif

Wikimedia Commons auf Deutsch

From Wikimedia Commons, the free media repository

File | File history | File usage on Commons | Metadata



CARM.

Affulſit, populo gratior it dies,
Et ſoles melius nitent.
Vt mater iuuenem, quem notus inuido
Flatu Carpathii trans maris æquora
Cunctantem ſpacio longius anno
Dulci detinet à domo,
Votis, omnibusq; et precibus uocat,
Curuo nec faciem littore dimouet,
Sic deſyderiis icta fidelibus
Quærit patria Cæſarem.
Tutus bos etenim rura perambulat.
Nutrit rura cæres, almáq; fauſtitas.
Pacatum uolitant per mare nauitæ.
Culpari metuit fides.
Nullis polluitur cæſta domus ſtupris.
Mos, et lex maculoſum edomuit nefas.
Laudantur ſimili prole puerperæ.
Culpam pœna premit comes.
Quis Parthum paueat? quis gelidum Scythen?
Quis, Germana quos horrida parturit,
Fœtus incolumi Cæſare? quis feræ
Bellum curet Iberiæ?
Condit quieta diem collibus in ſuis

Download
all sizes

Use this file
on the web

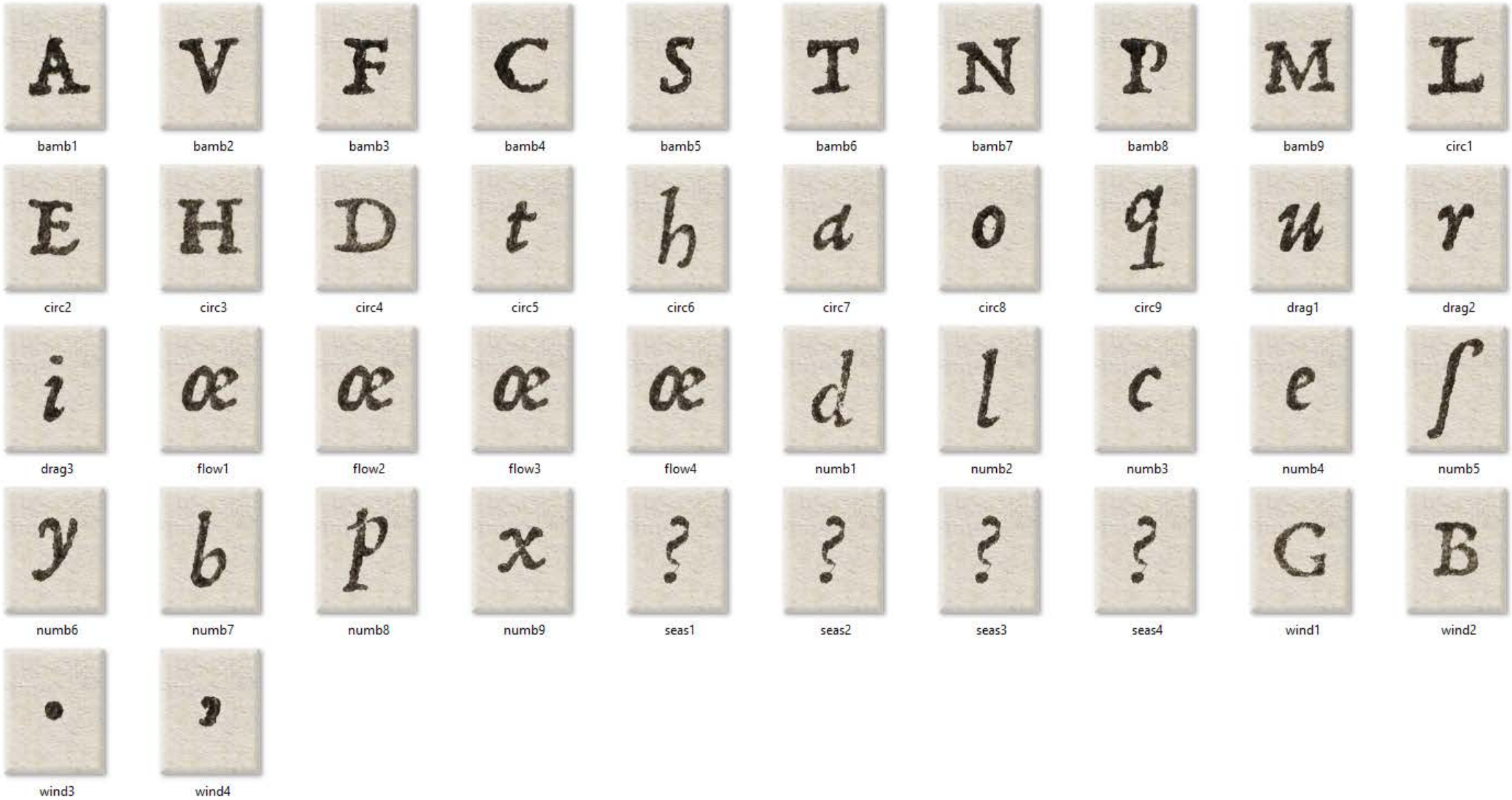
Use this file
on a wiki

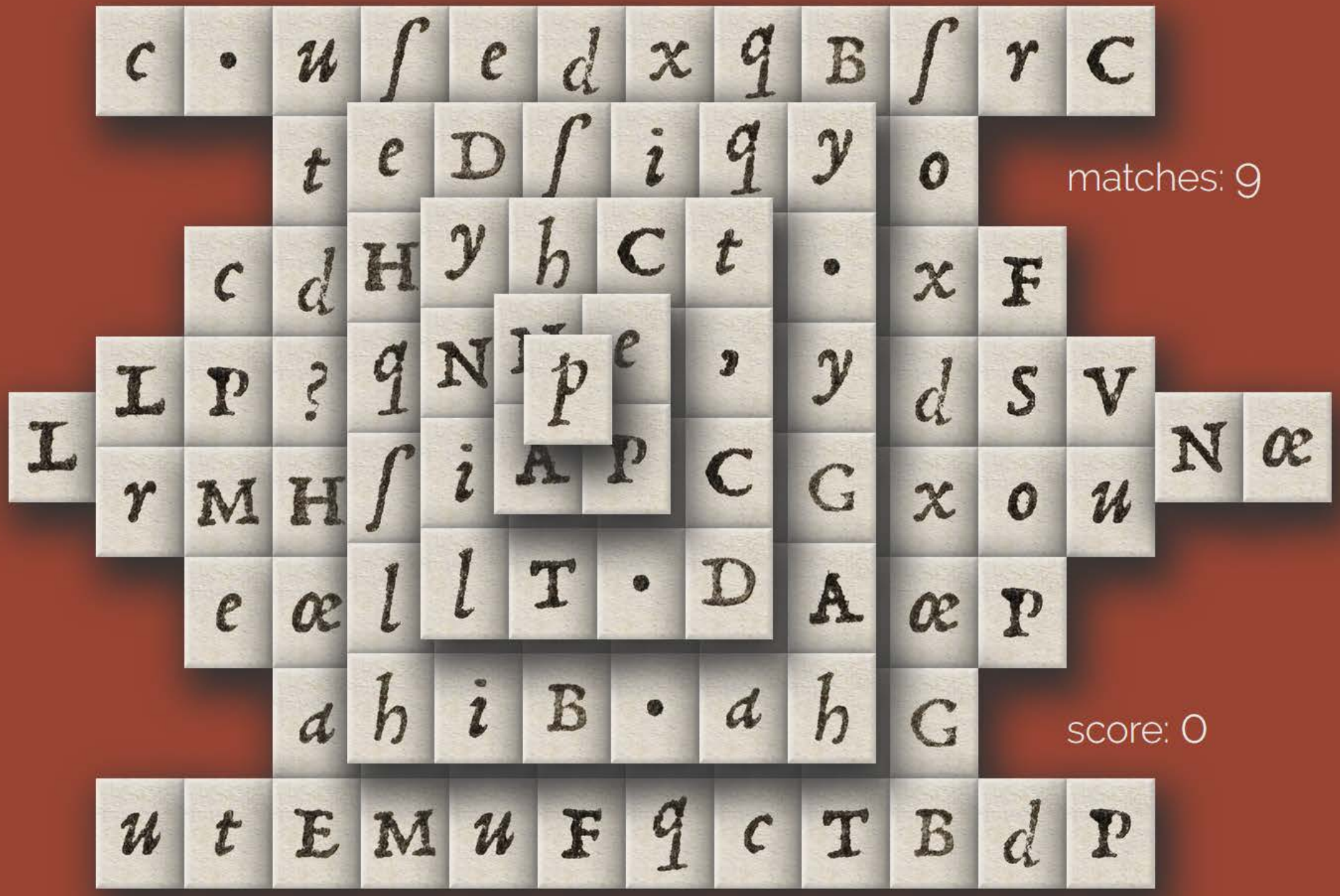
Email a link
to this file

Information
about reusing

Hinc ad uina redit lætus, et alteris
Temensis adhibet deum.
Te multa precæ, te prosequitur mero
Diffuso pateris, et laribus tuum
Miscet numem, uti græcia Castoris
Et magni memor Herculis.


bamb1
bamb2
bamb3
bamb4
bamb5
bamb6
bamb7
bamb8
bamb9
circ1
circ2
circ3
circ4
circ5
circ6
circ7
circ8
circ9
drag1
drag2
drag3
flow1
flow2
flow3
flow4
numb1
numb2
numb3
numb4
numb5
numb6
numb7
numb8
numb9
seas1
seas2
seas3
seas4
wind1
wind2
wind3
wind4


bamb1
bamb2
bamb3
bamb4
bamb5
bamb6
bamb7
bamb8
bamb9
circ1
circ2
circ3
circ4
circ5
circ6
circ7
circ8
circ9
drag1
drag2
drag3
flow1
flow2
flow3
flow4
numb1
numb2
numb3
numb4
numb5
numb6
numb7
numb8
numb9
seas1
seas2
seas3
seas4
wind1
wind2
wind3
wind4



matches: 9

score: 0

new

undo

reset

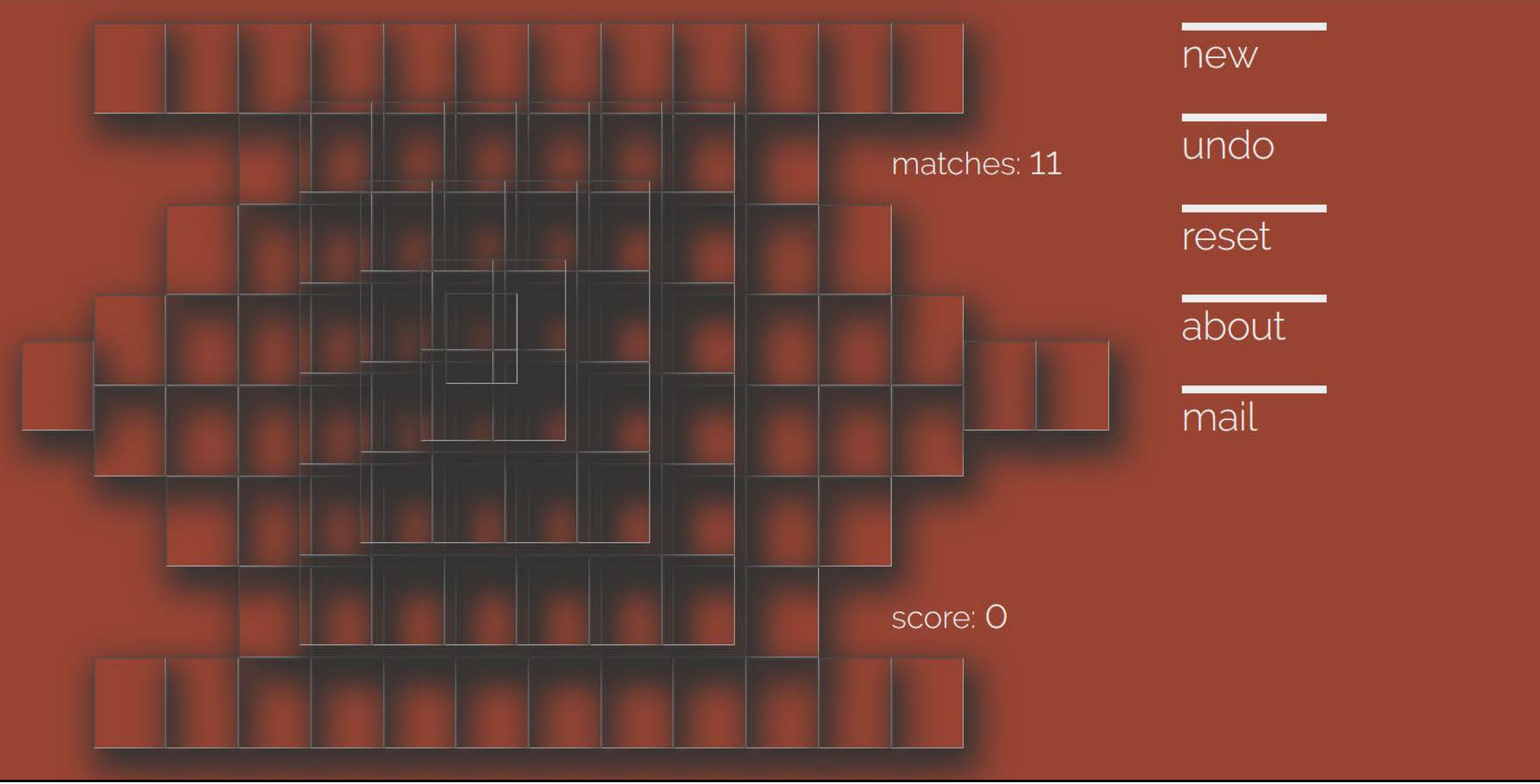
about

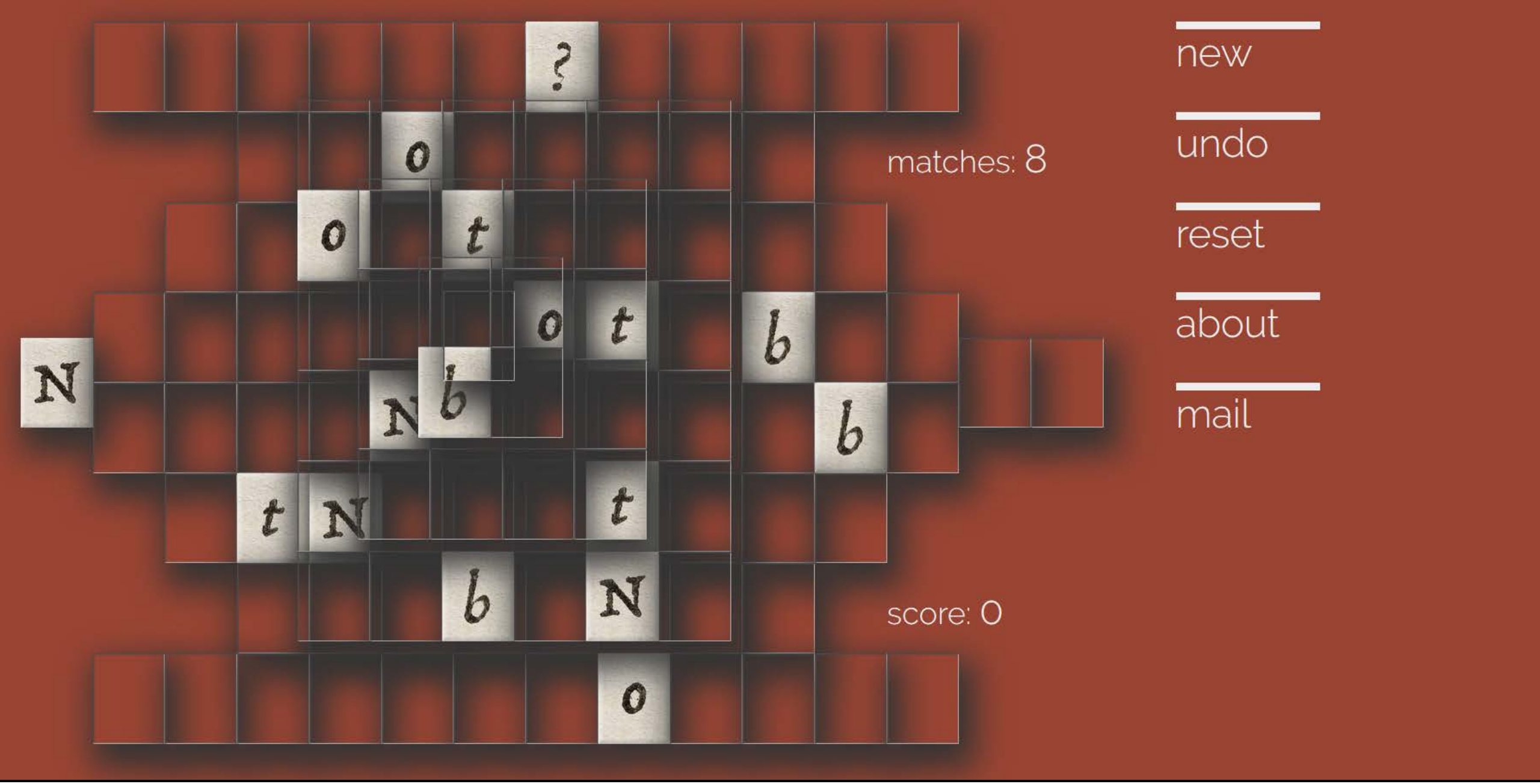
mail

thomasweibel.ch/letterjongg

HTML

```
<body>
  <div id="canvas"></div>
  <p id="new" class="button" onclick="newgame()"
  onmousedown="animon('new')"
  onmouseup="animoff('new')">new</p>
  ...
  <p id="moves">matches: <span id="movescounter">0</span></p>
  <p id="score">score: <span id="scorecounter">0</span></p>
  <div id="overlay"><p id="splash"><span
  id="gametitle">letterjongg!</span></p></div>
</body>
```





matches: 8

score: 0

new

undo

reset

about

mail

matches: 9

score: 0

new

undo

reset

about

mail

matches: 8

score: 0

new

undo

reset

about

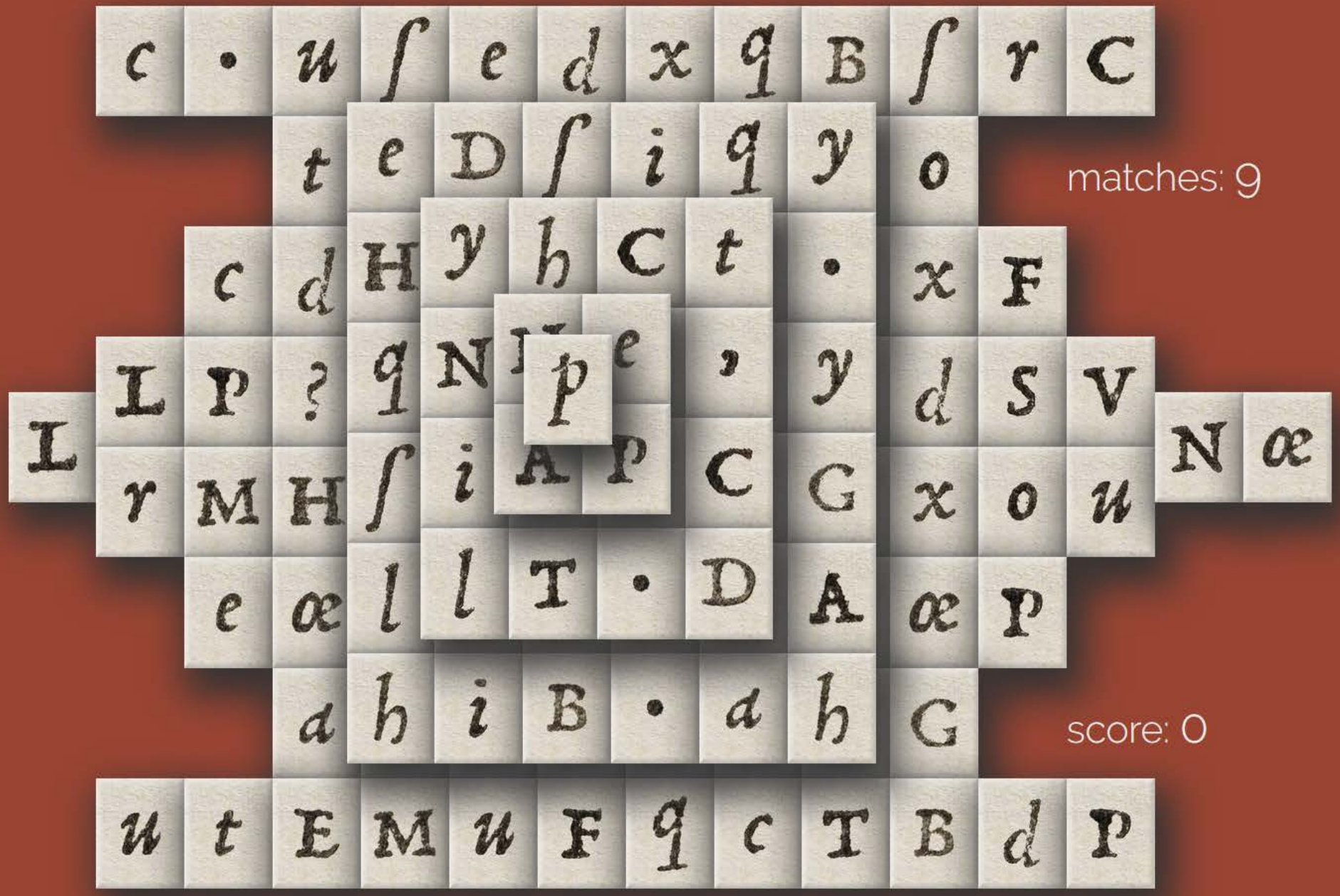
mail

Bilder vorladen

```
function preload() {
  tileface=["bamb1.jpg","bamb2.jpg","bamb3.jpg"...];

  for (i=0; i<tileface.length; i++) {
    images[i]=new Image();
    images[i].src="img\/"+tileface[i];
    images[i].onload=checkPreload;
  }
}
```

Bilder vorladen

```
function checkPreload() {
  img++;
  percent=img/tileface.length*100;

  document.getElementById("percentage").style.transform=
  "rotate("+(percent*3.6)+"deg)";

  if (img==images.length) {
    init();
  }
}
```



matches: 9

score: 0

new

undo

reset

about

mail

Shuffle-Algorithmus

```
Math.random() // → z.B. 0.264588171146922
```

Fisher-Yates-Verfahren [Bearbeiten | Quelltext bearbeiten]

Eine Verbesserung ist das Fisher-Yates-Verfahren (nach Ronald Fisher und Frank Yates). Dieses Verfahren arbeitet am Platz, das heißt, die Zahlen werden im Feld umsortiert und nicht in zusätzlichen Speicher kopiert. Sie werden schrittweise umsortiert, so dass am Ende eine zufällige Permutation entsteht. Um die aufwändige Suche nach einer noch nicht verwendeten Zahl zu vermeiden, wird in jedem Schritt die ausgewählte Zahl ans Ende des aktuell betrachteten Teilfelds gestellt, indem sie mit der letzten noch nicht ausgewählten Zahl vertauscht wird. Das Verfahren in Pseudocode:

```
function randperm(n)
  P = [1:n]              // Initialisierung mit der identischen Permutation
  for i = n downto 2     // Schleife über die Einträge von P (außer dem ersten)
    z = random(i)        // Gleichverteilte Zufallszahl 1 <= z <= i
    swap(P(i),P(z))      // Vertausche die Zahlen an den Stellen i und z
  end
  return P
end
```

Die Initialisierung kann dabei, wie im Codebeispiel, mit der identischen Permutation erfolgen, man kann aber auch von einer beliebigen Permutation ausgehen, die dann zufällig umsortiert wird. Die Permutationsroutine kann somit iterativ aufgerufen werden, wobei sie jeweils die vorherige Zufallspermutation umsortiert. Da die Vertauschung zweier Elemente eines Felds fester Größe mit einem konstanten Aufwand erfolgt, besitzt das Fisher-Yates-Verfahren eine Laufzeitkomplexität der Ordnung

$$O(n),$$

eine erhebliche Verbesserung gegenüber dem direkten Verfahren. In der nebenstehenden Tabelle wird das Verfahren anhand eines Beispiels illustriert, bei dem eine pseudozufällige Permutation der Länge sechs erzeugt wird. Die jeweils miteinander vertauschten Zahlen sind dabei unterstrichen. Es kann vorkommen, dass eine Zahl mit sich selbst vertauscht wird, was keinen Effekt hat. Es wurden die gleichen Zufallszahlen verwendet wie in dem Beispiel zum direkten Verfahren, allerdings ist die Ergebnispermutation hier eine andere.

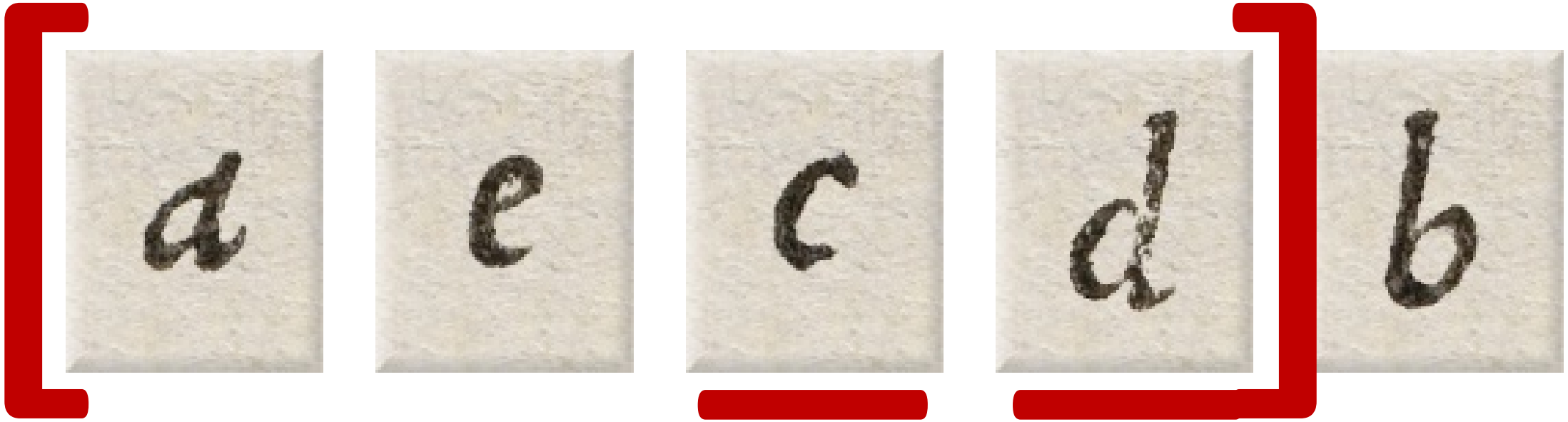
Bereich	Zufallszahl	Permutation
1–6	5	1 2 3 4 <u>5</u> <u>6</u>
1–5	3	1 2 <u>3</u> 4 <u>6</u> 5
1–4	4	1 2 6 <u>4</u> 3 5
1–3	1	<u>1</u> 2 <u>6</u> 4 3 5
1–2	2	6 <u>2</u> 1 4 3 5

Dieses Verfahren ist beispielsweise in dem numerischen Softwarepaket MATLAB als eingebaute Funktion `randperm` verfügbar.[6]

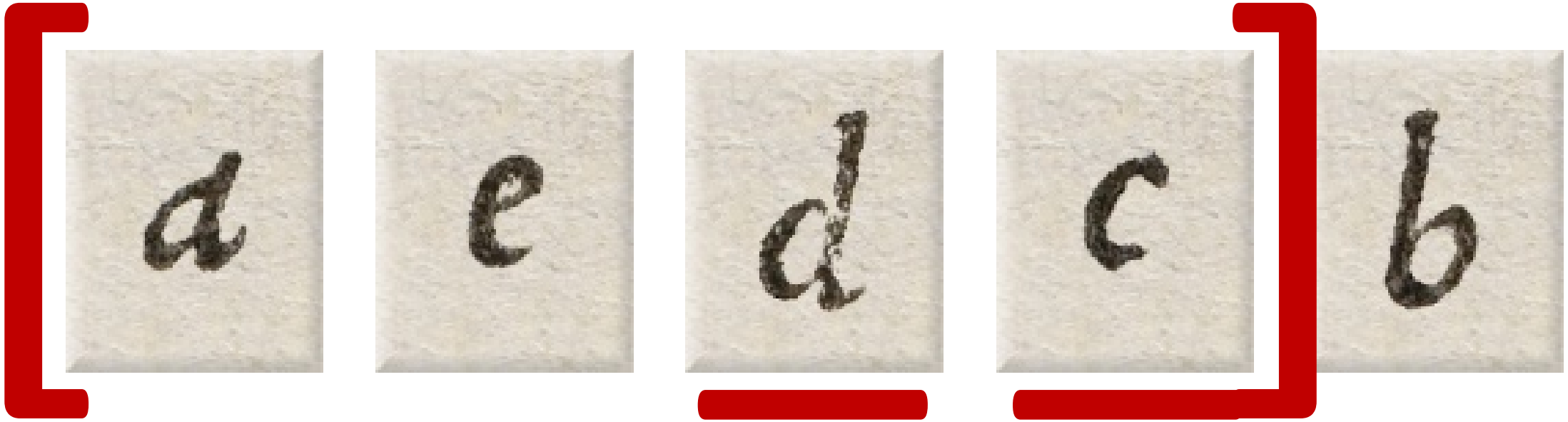
Shuffle-Algorithmus

Shuffle-Algorithmus

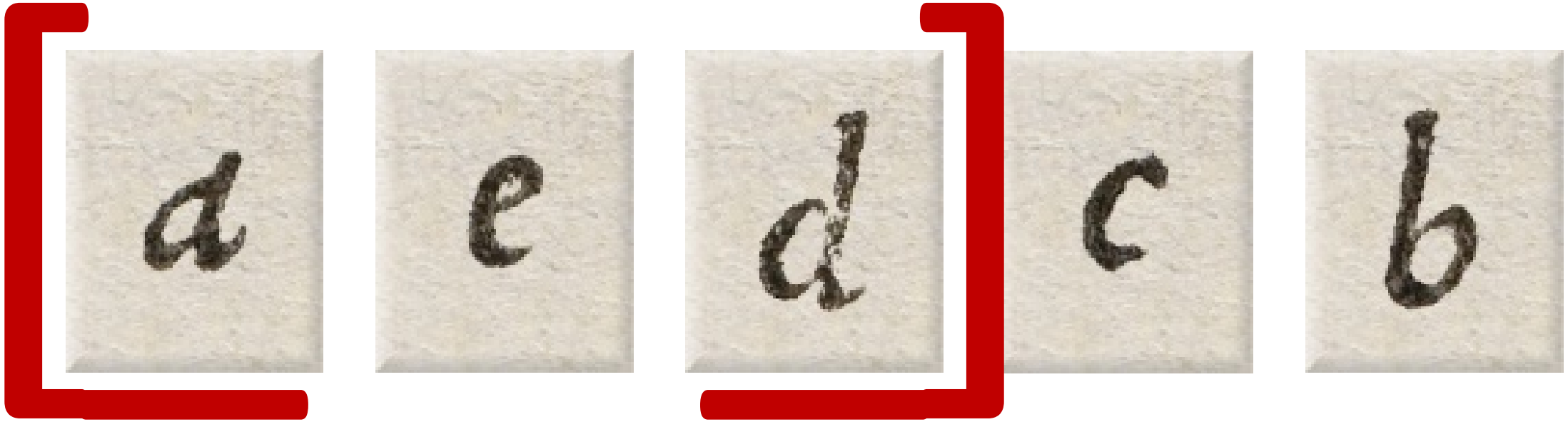
Shuffle-Algorithmus



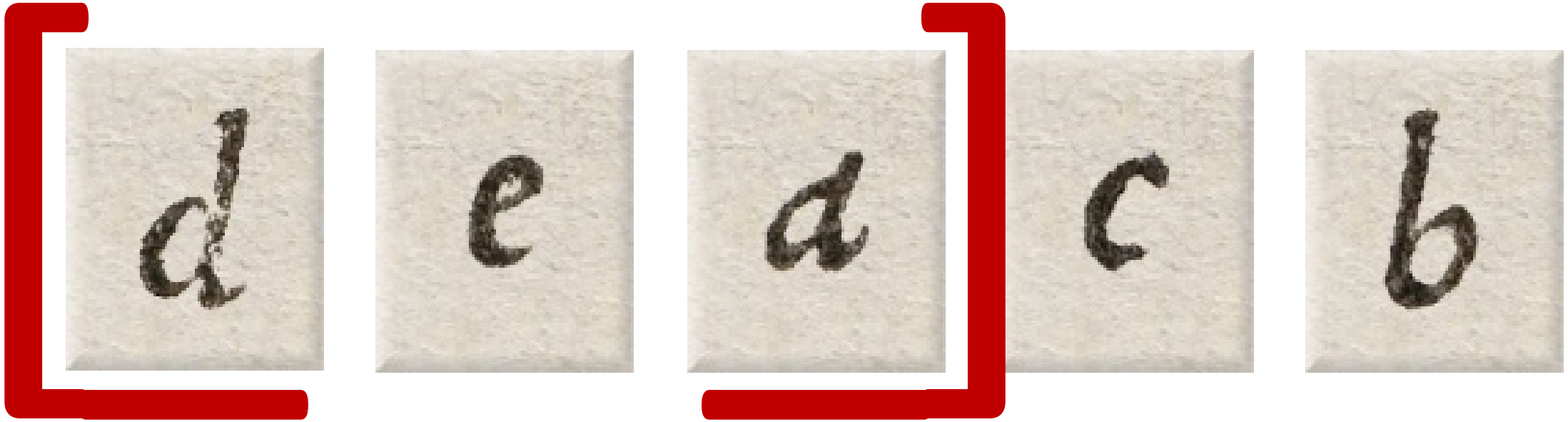
Shuffle-Algorithmus



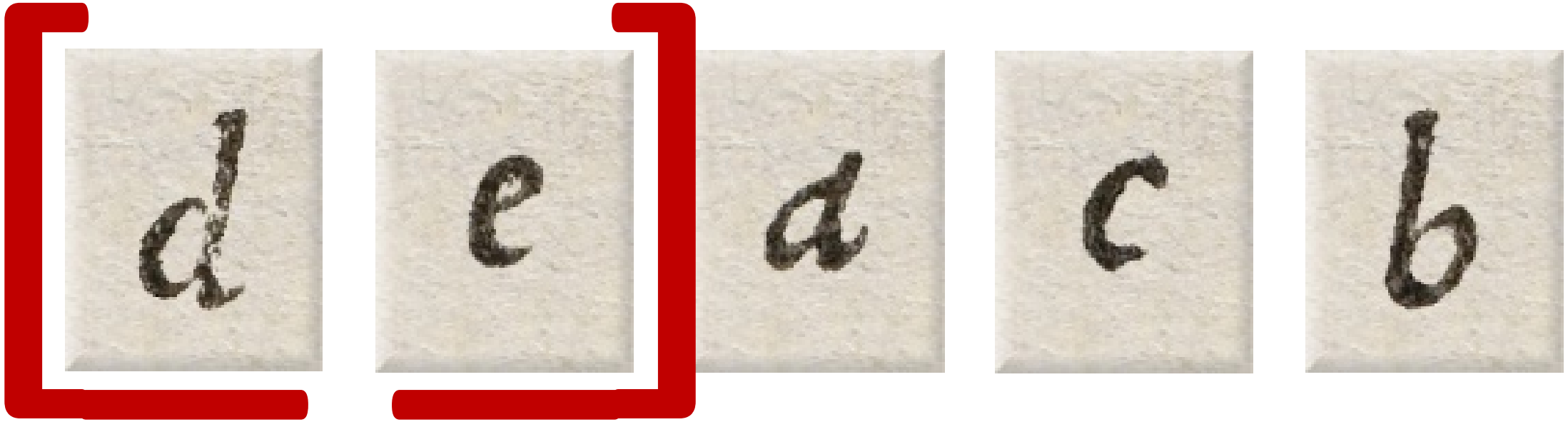
Shuffle-Algorithmus



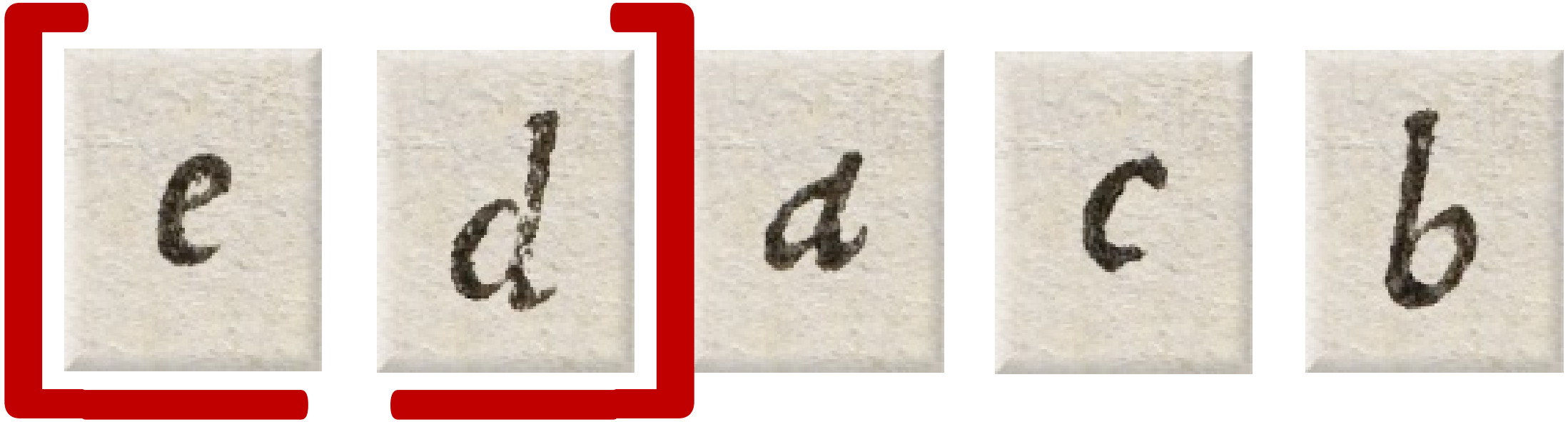
Shuffle-Algorithmus



Shuffle-Algorithmus



Shuffle-Algorithmus



Shuffle-Algorithmus (ES5)

```
arr=[0,1,2,3,4,5...n-1];

function shuffle() {
  for (i=n; i>0; i--) {
     var j=Math.floor(Math.random()*i);
     var k=arr[i-1];
     arr[i-1]=arr[j];
     arr[j]=k;
  }
}
```

Shuffle-Algorithmus (ES6)

```
arr=[0,1,2,3,4,5...n-1];

shuffle = () => {
  for (i=n; i>0; i--) {
     let j=Math.floor(Math.random()*i);
     [arr[j],arr[i-1]]=[arr[i-1],arr[j]];
  }
}
```

Dynamisches HTML

```
for (i=0; i<36; i++) {
  for (j=0; j<4; j++) {
    tile[arr[i*4+j]]="<img id=\""+(arr[i*4+j])+"\"
    onclick=\"check("+(arr[i*4+j])+")\" class=\"tile\"
    src=\"img\/"+tileface[i]+"\" name=\""+suitname[i]+"\">";
  }
}

html=tile.join("");
document.getElementById("canvas").innerHTML=html;
```

Dynamisches HTML

```
<img id="143" onclick="check(143)" class="tile"
src="img/bamb1.jpg" name="bamb1">
```

?

Dynamisches HTML

```
turtleX=[1,2,3,4,5,6,7,8,9,10,11,12,3,4,5,6,7,8,9,10...];
turtleY=[0,0,0,0,0,0,0,0,0, 0, 0, 0,1,1,1,1,1,1,1, 1...];
offsetX=9.2;
offsetY=11.5;

for (i=0; i<87; i++) { // layer 1
  el=document.getElementById(i);
  el.style.top=(turtleY[i]*offsetY)+"vh";
  el.style.left=(turtleX[i]*offsetX)+"vh";
}
```

Dynamisches HTML

```
<img id="143" onclick="check(143)" class="tile"
src="img/bamb1.jpg" name="bamb1" style="left: 63.8vh; top:
37.25vh;">
```

Dynamisches HTML

```
▼ <body>
  ▼ <div id="canvas">
      <img id="0" class="tile" onclick="check(0)" src="img/bamb1.jpg" name="bamb1" alt="" style="top: 3vh; left: 19.2vh; height: 11.5vh; width: 9.2vh; opacity: 1; cursor: pointer;">
       event
      <img id="1" class="tile" onclick="check(1)" src="img/circ9.jpg" name="circ9" alt="" style="top: 3vh; left: 28.4vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="2" class="tile" onclick="check(2)" src="img/bamb9.jpg" name="bamb9" alt="" style="top: 3vh; left: 37.6vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="3" class="tile" onclick="check(3)" src="img/bamb9.jpg" name="bamb9" alt="" style="top: 3vh; left: 46.8vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="4" class="tile" onclick="check(4)" src="img/circ6.jpg" name="circ6" alt="" style="top: 3vh; left: 56vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="5" class="tile" onclick="check(5)" src="img/numb9.jpg" name="numb9" alt="" style="top: 3vh; left: 65.2vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="6" class="tile" onclick="check(6)" src="img/bamb6.jpg" name="bamb6" alt="" style="top: 3vh; left: 74.4vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="7" class="tile" onclick="check(7)" src="img/circ8.jpg" name="circ8" alt="" style="top: 3vh; left: 83.6vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="8" class="tile" onclick="check(8)" src="img/bamb4.jpg" name="bamb4" alt="" style="top: 3vh; left: 92.8vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="9" class="tile" onclick="check(9)" src="img/numb7.jpg" name="numb7" alt="" style="top: 3vh; left: 102vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="10" class="tile" onclick="check(10)" src="img/wind4.jpg" name="wind4" alt="" style="top: 3vh; left: 111.2vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="11" class="tile" onclick="check(11)" src="img/bamb8.jpg" name="bamb8" alt="" style="top: 3vh; left: 120.4vh; height: 11.5vh; width: 9.2vh; opacity: 1; cursor: pointer;">
       event
      <img id="12" class="tile" onclick="check(12)" src="img/circ4.jpg" name="circ4" alt="" style="top: 14.5vh; left: 37.6vh; height: 11.5vh; width: 9.2vh; opacity: 1; cursor:
      pointer;"> event
      <img id="13" class="tile" onclick="check(13)" src="img/bamb7.jpg" name="bamb7" alt="" style="top: 14.5vh; left: 46.8vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="14" class="tile" onclick="check(14)" src="img/numb5.jpg" name="numb5" alt="" style="top: 14.5vh; left: 56vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="15" class="tile" onclick="check(15)" src="img/circ6.jpg" name="circ6" alt="" style="top: 14.5vh; left: 65.2vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="16" class="tile" onclick="check(16)" src="img/bamb3.jpg" name="bamb3" alt="" style="top: 14.5vh; left: 74.4vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="17" class="tile" onclick="check(17)" src="img/bamb7.jpg" name="bamb7" alt="" style="top: 14.5vh; left: 83.6vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="18" class="tile" onclick="check(18)" src="img/wind3.jpg" name="wind3" alt="" style="top: 14.5vh; left: 92.8vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="19" class="tile" onclick="check(19)" src="img/wind3.jpg" name="wind3" alt="" style="top: 14.5vh; left: 102vh; height: 11.5vh; width: 9.2vh; opacity: 1; cursor:
      pointer;"> event
      <img id="20" class="tile" onclick="check(20)" src="img/wind3.jpg" name="wind3" alt="" style="top: 26vh; left: 28.4vh; height: 11.5vh; width: 9.2vh; opacity: 1; cursor: pointer;">
       event
      <img id="21" class="tile" onclick="check(21)" src="img/circ7.jpg" name="circ7" alt="" style="top: 26vh; left: 37.6vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
      <img id="22" class="tile" onclick="check(22)" src="img/bamb5.jpg" name="bamb5" alt="" style="top: 26vh; left: 46.8vh; height: 11.5vh; width: 9.2vh; opacity: 1;"> event
```

Algorithmus

```
lr=((a-1==removed[a-1]) || (a+1==removed[a+1]));

cond=(
  ( // initially playable tiles
    a==0 || a==11 || a==12 || a==19 || a==20 || a==29 ||
    a==30 || a==56 || a==57 || a==66 || a==67 || a==74 ||
    a==75 || a==86 || a==87 || a==92 || a==93 || a==98 ||
    a==99 || a==104 || a==105 || a==110 || a==111 || a==116
    || a==117 || a==122 || a==123 || a==126 || a==127 ||
    a==130 || a==131 || a==134 || a==135 || a==138 || a==143
  ) ||
```

Algorithmus

```
( // level 1, row 1
  (a>0 && a<11) && lr
) ||

...

( // level 4
  (a>138 && a<143) && (143==removed[143])
)
```

Algorithmus

```
store=[];

if (cond) {
  if (store.length<2) {
    store.push(document.getElementById(a));
  }
}
```

Algorithmus

```
removed=[];

if (store.length==2 && store[0].name==store[1].name) {
  for (i=0; i<2; i++) {
    store[i].style.opacity="0";
    removed[store[i].id]=store[i].id;
  }

  delay1=setTimeout(remove, 400);
}
```

Unbeschränktes Undo

```
archive=[];

if (archive.length>0) {
  for (i=1; i<3; i++) {
    archive[archive.length-i].style.display="block";
  }

  delay1=setTimeout(backone, 200);
}
```

Unbeschränktes Undo

```
function backone() {
  for (i=1; i<3; i++) {
    archive[archive.length-i].style.opacity="1";
    removed[archive[archive.length-i].id]="";
  }
}
```

Reset (gleiche Partie)

```
function reset() {
  if (archive.length>0) {
    for (i=1; i<archive.length+1; i++) {
      archive[archive.length-i].style.display="block";
    }
  }
}
```

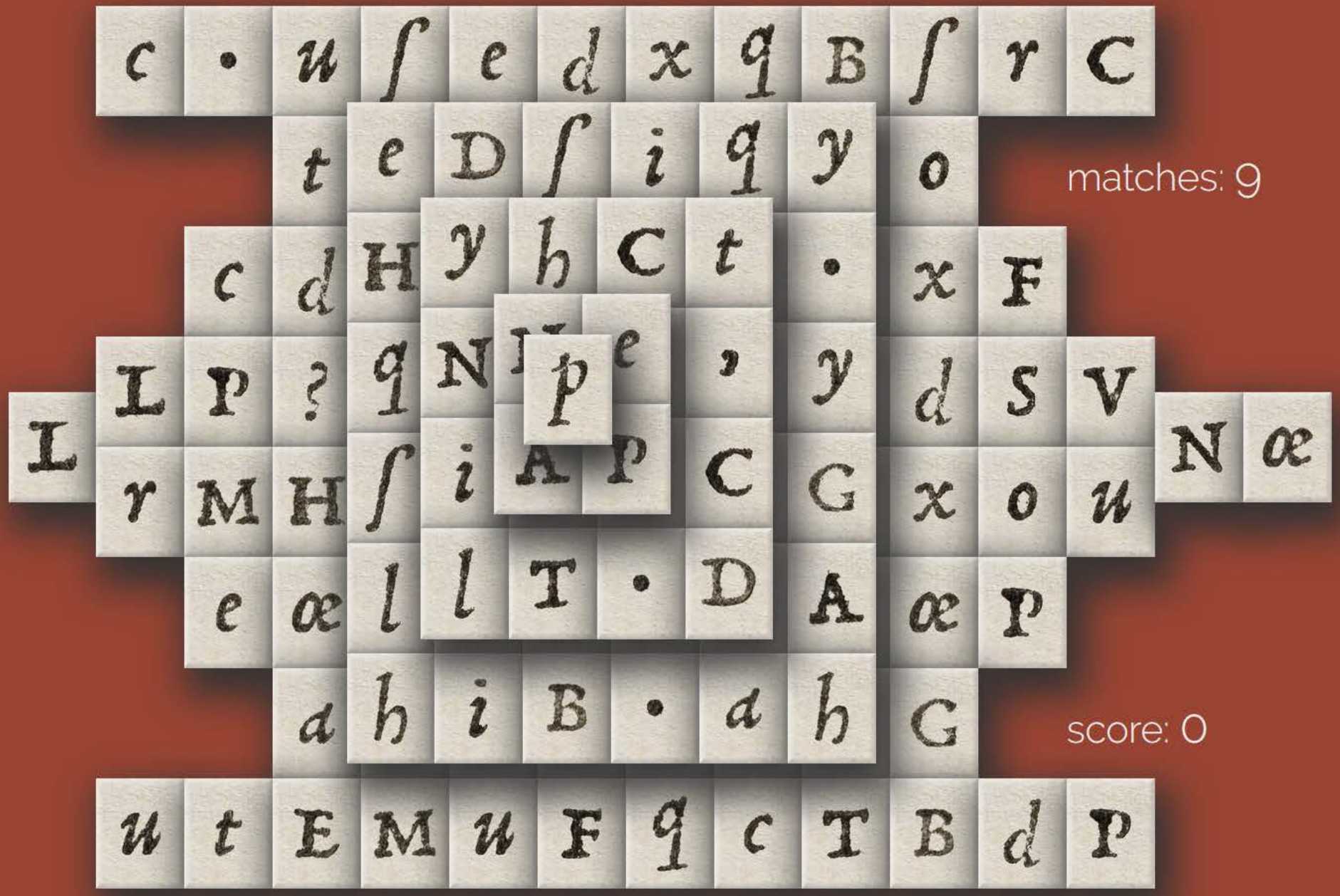
Zugzähler (Autoplay)

```
possible=[];
matches=0;

for (i=0; i<144; i++) {
  if (tile[i]!=tile[removed[i]] && cond) {
    possible.push(document.getElementById(i).name);
  }
}

possible.sort();
```

Zugzähler (Autoplay)

```
while (possible.length>1) {
  if (possible[possible.length-1]==possible[possible.length-
  2]) {
    matches++;
    possible.length-=2;
  }
}

document.getElementById("movescounter").innerText=matches;
```


lenovo



matches: 9

score: 0

new

undo

reset

about

mail