

Project Euler 79. Passcode Derivation

hiragn

2025 年 1 月 12 日

1. 問題の概要

オンラインバンキングではあらかじめ渡されているパスコードからランダムに選んだ 3 文字の入力を要求されることがある。たとえばパスコードが 531278 で 2, 3, 5 の数字を要求された場合、期待される答えは 317 である。

テキストファイル `keylog.txt`^a にはログインに成功した 50 回の試行が記録されている。このファイルを分析して考えられるパスコードのうち最も短いものを探せ。

<https://projecteuler.net/problem=79>

^a https://projecteuler.net/project/resources/0079_keylog.txt

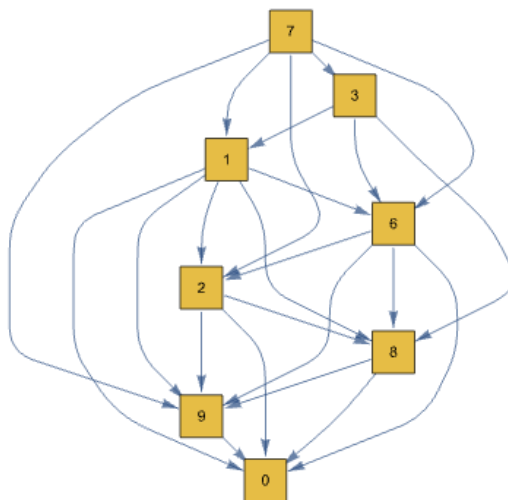
2. 解法

たとえば 123 でログインに成功した場合、パスコード中で「1 は 2 の前」「2 は 3 の前」にあることがわかります。この情報を有向辺 $1 \rightarrow 2$, $2 \rightarrow 3$ に変換するとグラフができます。

```

1 Clear["Global`*"];
2 passcodes =
3   Flatten@DeleteDuplicates@Import["0079_keylog.txt", "CSV"];
4 makeEdge[n_] := Module[{a, b, c},
5   {a, b, c} = IntegerDigits@n;
6   {DirectedEdge[a, b], DirectedEdge[b, c]};
7 gr = SimpleGraph[DeleteDuplicates@Flatten[makeEdge /@ passcodes, 2],
8   VertexLabels -> Placed[Automatic, Center],
9   VertexSize -> .75, VertexShapeFunction -> "Square",
10  VertexStyle -> Hue[0.125, 0.7, 0.9]]

```



答えは「73162890」です。グラフから 2 回以上含まれる数字がないこともわかるので、パスコードはこのグラフ上のハミルトン路（すべての頂点を一回ずつ通る路）です。

```

1 In[] := Clear["Global`*"];
2 passcodes =
3   Flatten@DeleteDuplicates@Import["0079_keylog.txt", "CSV"];
4 RepeatedTiming[
5   makeEdge[n_] := Module[{a, b, c},
6     {a, b, c} = IntegerDigits@n;
7     {DirectedEdge[a, b], DirectedEdge[b, c]}];
8   gr = SimpleGraph[DeleteDuplicates@Flatten[makeEdge /@ passcodes]];
9   ans = FromDigits@FindHamiltonianPath@gr]
10
11 Out[] = {0.000529978, 73162890}

```

トポロジカルソートでも解けます。

```

1 In[] := Clear["Global`*"];
2 passcodes =
3   Flatten@DeleteDuplicates@Import["0079_keylog.txt", "CSV"];
4 RepeatedTiming[
5   makeEdge[n_] := Rule @@@ Partition[IntegerDigits@n, 2, 1];
6   gr = Graph[DeleteDuplicates@Flatten[makeEdge /@ passcodes]];
7   ans = FromDigits@TopologicalSort@gr]
8
9 Out[] = {0.000117821, 73162890}

```
