

# 一筆書きで遊ぼう

黒木玄

2018-09-03

- Copyright 2018 Gen Kuroki
- License: MIT <https://opensource.org/licenses/MIT> (<https://opensource.org/licenses/MIT>)
- Repository: <https://github.com/genkuroki/RecreationalMath/tree/master/MF2018>  
(<https://github.com/genkuroki/RecreationalMath/tree/master/MF2018>)

このファイルは次の場所でよりきれいに閲覧できる:

- [nbviewer](http://nbviewer.jupyter.org/github/genkuroki/RecreationalMath/blob/master/MF2018/onestroke.ipynb?flush_cache=true) ([http://nbviewer.jupyter.org/github/genkuroki/RecreationalMath/blob/master/MF2018/onestroke.ipynb?flush\\_cache=true](http://nbviewer.jupyter.org/github/genkuroki/RecreationalMath/blob/master/MF2018/onestroke.ipynb?flush_cache=true))

関連ウェブサイト:

- [https://mathtrain.jp/euler\\_graph](https://mathtrain.jp/euler_graph) ([https://mathtrain.jp/euler\\_graph](https://mathtrain.jp/euler_graph))
- <https://www3.cs.stonybrook.edu/~skiena/combinatorica/animations/euler.html>  
(<https://www3.cs.stonybrook.edu/~skiena/combinatorica/animations/euler.html>)

一筆書きの数学の解説については

- <http://jlmartin.faculty.ku.edu/~jlmartin/courses/math105-F11/Lectures/chapter5-part2.pdf>  
(<http://jlmartin.faculty.ku.edu/~jlmartin/courses/math105-F11/Lectures/chapter5-part2.pdf>)

が非常に詳しい. 日本語による解説を読みたいならば

- <http://math.cs.kitami-it.ac.jp/~fuchino/chubu/method-math-WS06.pdf> (<http://math.cs.kitami-it.ac.jp/~fuchino/chubu/method-math-WS06.pdf>)

という一筆書きを例に数学の考え方を解説した講義録もある.

一筆書きはすべての線を一回だけ通る経路のことである. そのような経路は**Euler(オイラー)経路**とも呼ばれる. すべての線ではなく, すべての頂点を1回だけ通る経路は**Hamilton(ハミルトン)経路**と呼ばれており, 一筆書きよりもずっと難しい数学になっている. 一筆書きをマスターしたら, Hamilton経路についても調べてみると面白いかもしれない.

- <http://www.sci.brooklyn.cuny.edu/~amotz/BC-ALGORITHMS/PRESENTATIONS/tours.pdf>  
(<http://www.sci.brooklyn.cuny.edu/~amotz/BC-ALGORITHMS/PRESENTATIONS/tours.pdf>)

には一筆書きとハミルトン経路の両方に関する解説がある.

グラフ理論の日本語による**易しい**解説は

- <https://www.kobepharmaceutical.ac.jp/knot/> (<https://www.kobepharmaceutical.ac.jp/knot/>)

の**配布資料** (<https://www.kobepharmaceutical.ac.jp/knot/document/index.html>)の2014年のグラフ理論の項目に見付かる.

**Julia言語:** [Julia言語](https://www.google.co.jp/search?q=Julialang) (<https://www.google.co.jp/search?q=Julialang>)のインストールについては

- [WindowsへのJulia言語のインストール](http://nbviewer.jupyter.org/gist/genkuroki/81de23edcae631a995e19a2ecf946a4f) (<http://nbviewer.jupyter.org/gist/genkuroki/81de23edcae631a995e19a2ecf946a4f>)

を参照. このノートブックでは v0.6.4 を使用している. その理由は v1.0.0 はこれを書いている時点で Windows 上の Jupyter で利用できないから.

## 目次

[1 Julia言語によるグラフのプロットの例](#)

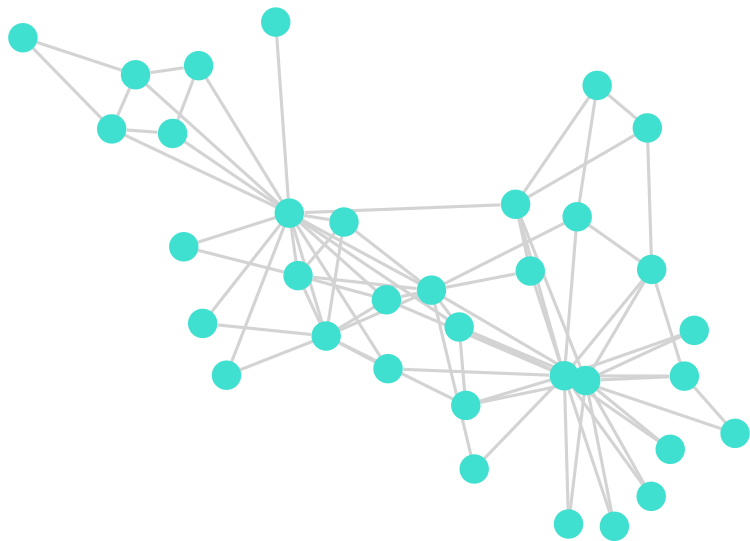
## Julia言語によるグラフのプロットの例

- <https://github.com/JuliaGraphs/GraphPlot.jl> (<https://github.com/JuliaGraphs/GraphPlot.jl>)
- <https://github.com/JuliaGraphs/LightGraphs.jl> (<https://github.com/JuliaGraphs/LightGraphs.jl>)

```
In [1]: 1 using GraphPlot # これを書いている時点で GraphPlot は Julia v1.0 にまだ対応していない
        2 using LightGraphs
```

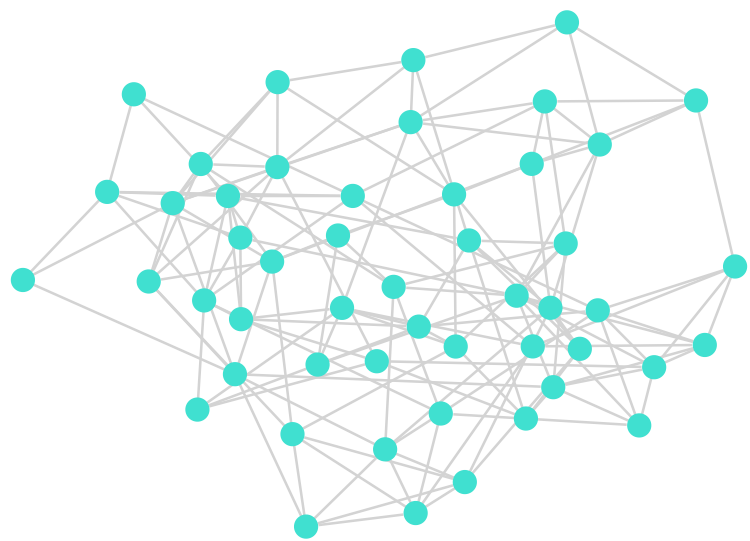
```
In [2]: 1 g = graphfamous("karate")
        2 gplot(g)
```

Out[2]:



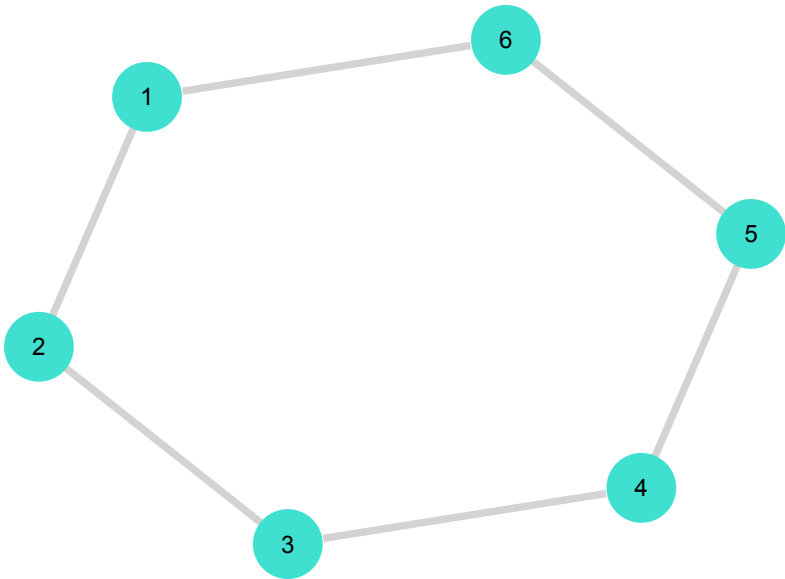
```
In [3]: 1 h = watts_strogatz(50, 6, 0.3)
        2 gplot(h)
```

Out[3]:



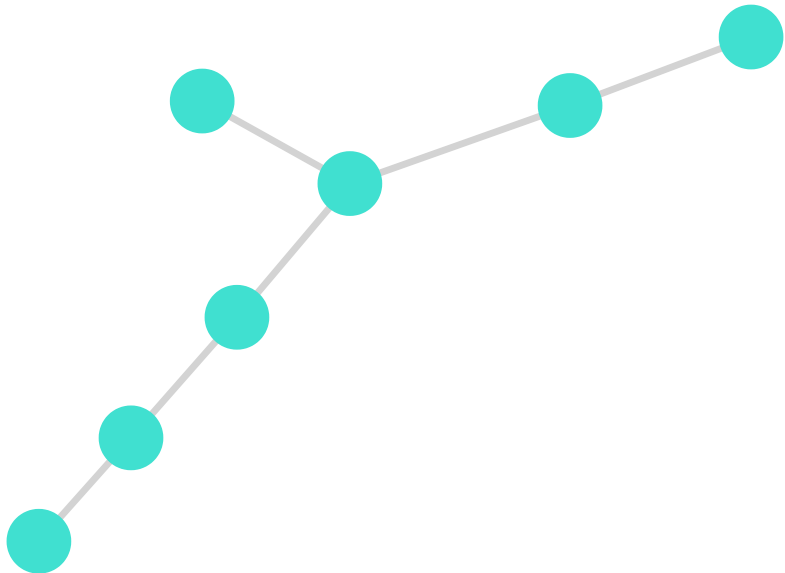
```
In [4]: 1 g = PathGraph(6)
2 nl = 1:nv(g)
3 add_edge!(g, 1, 6)
4 gplot(g, nodeLabel=nl)
```

Out[4]:

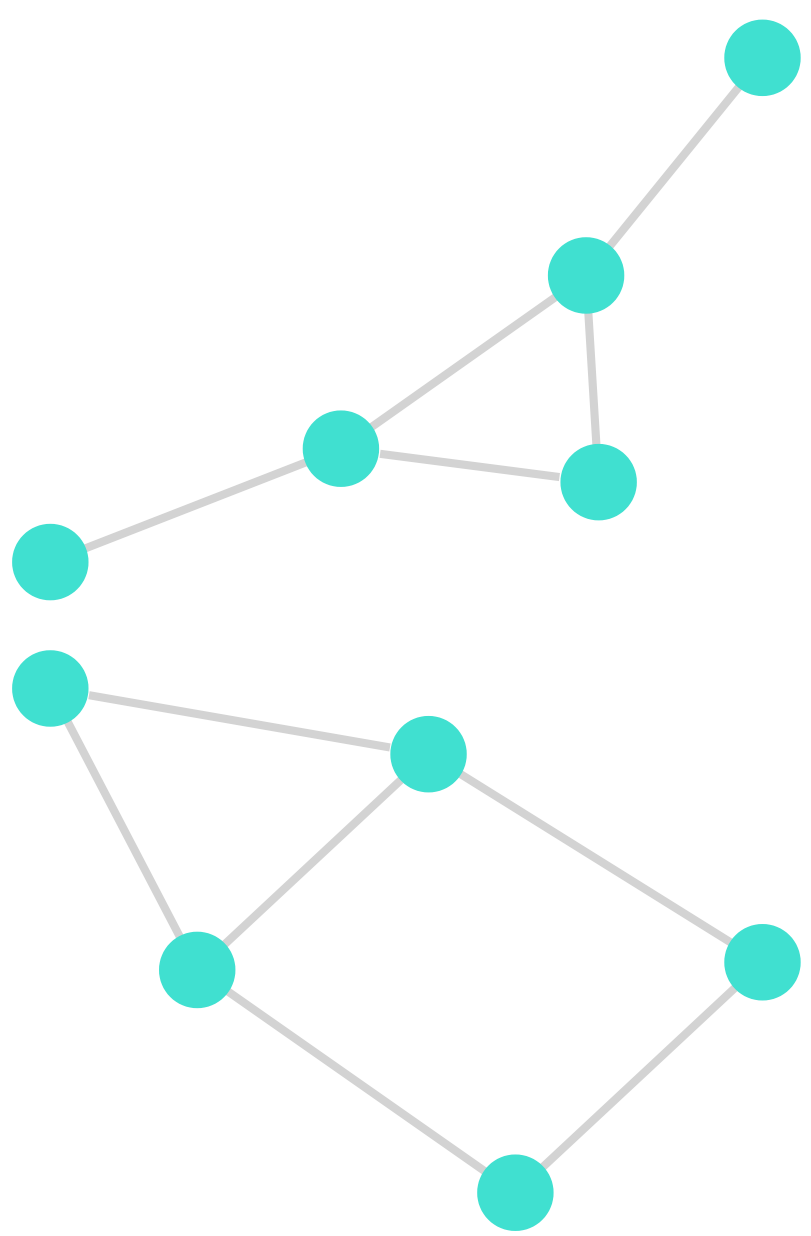


```
In [5]: 1 A = [
2     0 1 0 0 0 0 0 0
3     1 0 1 0 0 0 0 0
4     0 1 0 1 0 0 0 0
5     0 0 1 0 1 1 0 0
6     0 0 0 1 0 0 0 0
7     0 0 0 1 0 0 1 0
8     0 0 0 0 0 1 0 0
9     ]
10 g = Graph(A)
11 gplot(g)
```

Out[5]:



```
In [6]: 1 ▼ for shape in ["bull", "house"]
2         G = smallgraph(shape)
3         display(gplot(G))
4     end
```



```
In [7]: 1 h = WheelGraph(10)  
2 gplot(h)
```

Out[7]:

