

HTML5 Canvas でいろいろいろいろ

JOI 2016-2017 夏季セミナー 発表

米田 寛峻 (square1001 / @square10011)

自己紹介

- 中学3年 米田 寛峻 (square1001) です
 - 競技プログラミングをやったりしています

 - Atcoder : レーティング **2468**
 - Topcoder : レーティング **1995**
 - CS Academy : レーティング **1682**
- ※これは2017.8.29時点でのレーティングです



目次

1. My Website
2. JavaScript / Canvas についての説明
3. Webpage 1 – Falling Game
4. Webpage 2 – Maze Solver

1. My Website

HTML5 Canvas でいろいろ

My Website

<https://square1001.github.io> を見てください

ここに JOI 2017 夏季セミナー で作ったページ
があります

2. JavaScript / Canvas についての説明

HTML5 Canvas でいろいろ

HTML5 Canvas では何をやったのか？

基本的に、次のような感じですよ

- JavaScript を書く方法を理解する
- Canvas を使う方法を理解する
- HTML5 Canvas を使ってウェブページを作る！

JavaScript について

JavaScript には、次の2個の特徴があります

1. HTML と JavaScript は連携できる！
2. JavaScript を使うことによって Canvas を使うことができる！

ここでは、時間の関係上、文法については触れません。

文法を知りたい人は <https://www.w3schools.com/js/> などのサイトで学習してみてください。

JavaScript について

1. JavaScript は HTML と連携できる！

- HTMLのコードの中で `<script>` タグを使うことによってJavaScriptを実行することができる
- HTMLのコードの `onclick` などを使うと、JavaScriptのコードや関数が呼び出せる
- JavaScriptのコードの中で `document.getElementById` という関数を使うことによって表示されるものを変えることができる
- 基本的にこの3つのことでHTMLとJavaScriptは連携している

JavaScript について

2. JavaScript を使うことによって Canvas を使うことができる！

- JavaScript には、Canvasを操る関数が定義されている
- このコードをHTMLから読み込むことによって、Canvasが動く

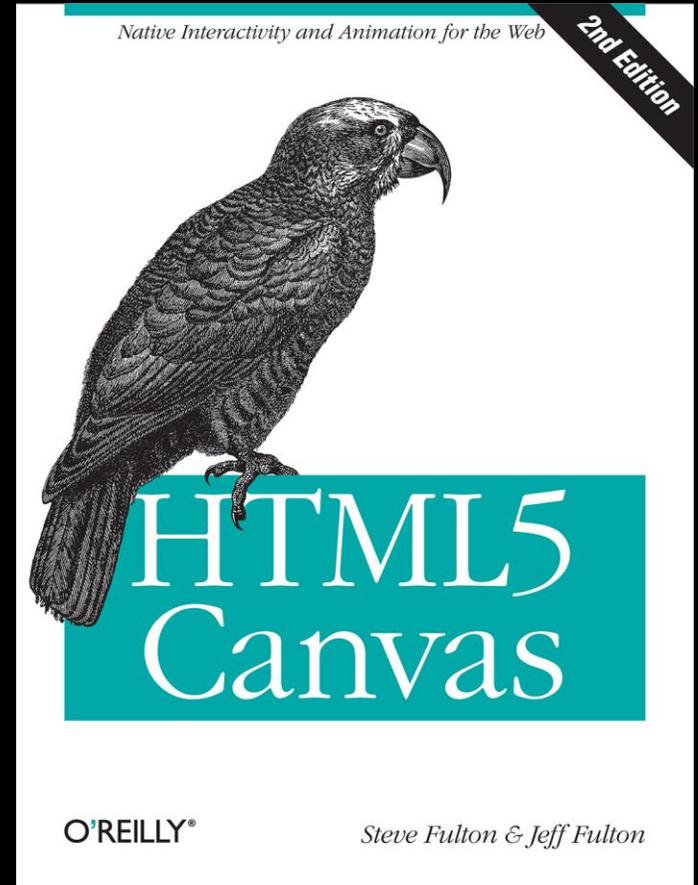


Canvas について

- Canvas は、HTML上 (要するにWebページ上) でグラフィックスを表現することができるもの
- Canvas を使うと、動くものも作れて、JavaScriptではキー入力やマウスなどの情報を受け取れることを考えると、いろいろな作品を作ることができそう！
- これはHTML5の新機能であり、これが作られる前はAdobe Flashが使われていた

Canvasについて

- HTML の `<canvas>` タグで、canvasの画面を作ることができる
- JavaScript上でcanvasの画面を動かすことができる、当然ほかのこともできる
- Canvasの描画関数は「HTML5 Canvas」という本を読むと分かる

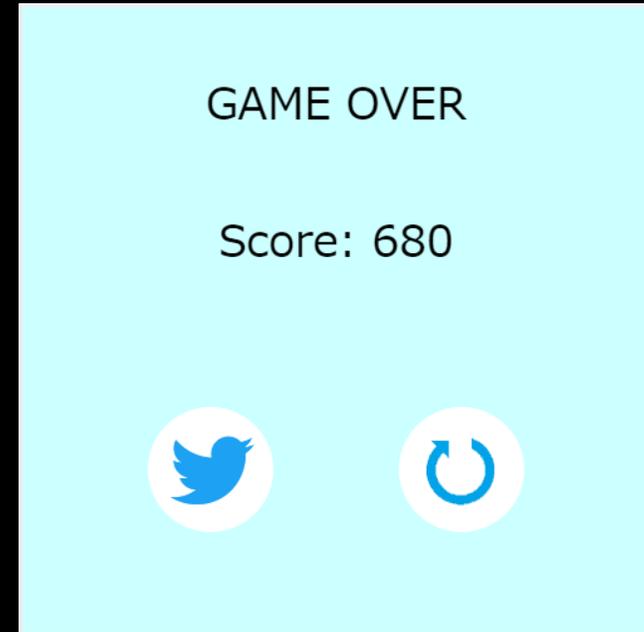
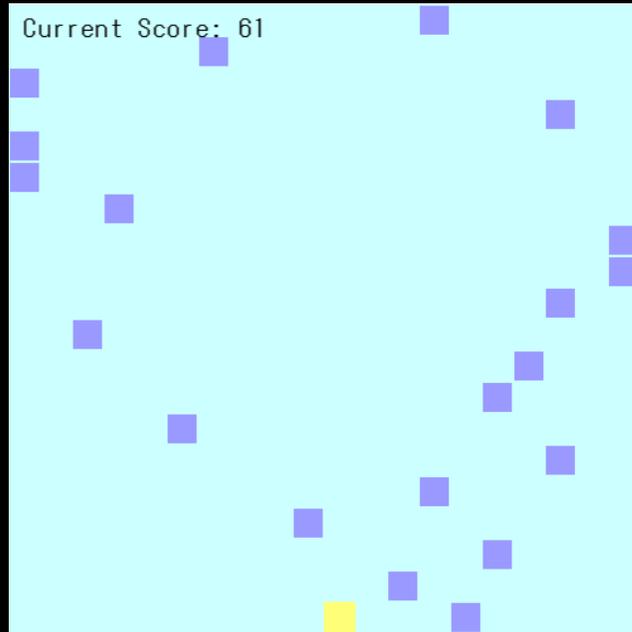


3. Webpage 1 – Falling Game

HTML5 Canvas でいろいろ

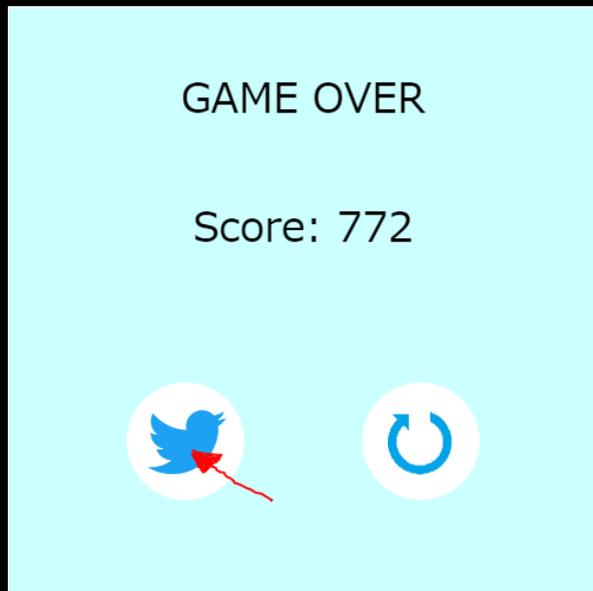
Falling Game

- リンクは <https://square1001.github.io/joiss2017/falling-game.html>
- 青い正方形の物体が下に落ちていくので、できるだけオレンジの物体が当たらないように左右に動かすゲーム



Falling Game

- ちなみに僕の最高スコアは1123
- ツイートボタンも付けたので、みなさんもこのゲームをプレイしてツイートをしましょう (ハッシュタグは #joissfallinggame)



4. Webpage 2 – Maze Solver

HTML5 Canvas

Maze Solver

- リンクは <https://square1001.github.io/joiss2017/maze-solver.html>
- 迷路を入力すると最短経路で解く方法を表示するページ

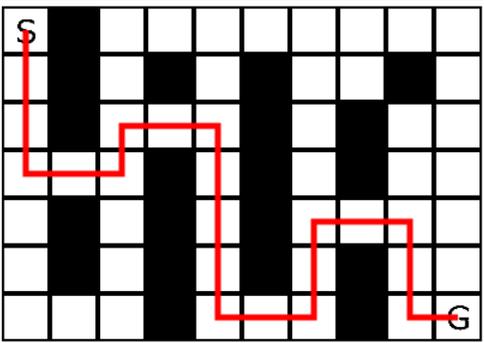
Maze Solver

Input

Number of Row ($2 \leq r \leq 30$):

Number of Column ($2 \leq c \leq 30$):

Editor / Solver



The maze is a 7x10 grid. The start cell 'S' is at (1,1) and the goal cell 'G' is at (7,10). A red path is shown starting from 'S' and ending at 'G'. The path consists of the following cells: (1,1), (1,2), (2,2), (2,3), (3,3), (3,4), (4,4), (4,5), (5,5), (5,6), (6,6), (6,7), (6,8), (6,9), (6,10), (7,10).

How to use this editor

You can choose four species of block from the selector which is on right of the editor.
Currently it is **Running mode**.
If you want to edit, click "Edit" button.

Results / Notifications

The distance of shortest path is **21**.

Maze Solver

どのようにして最短経路を求めたか？
次のような問題を考えてみよう

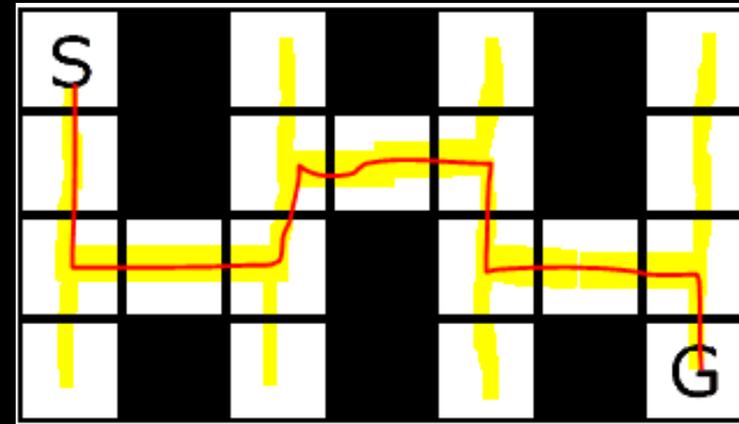
問題

$H \times W$ のグリッドがある。各マスは白か黒に塗られている。スタート地点からゴール地点に最短経路で行く方法のうち1個を求めなさい。

Maze Solver

隣接したマスが両方白いとき、これらのマスの上に辺を張ってグラフを作る

そのとき、グラフの頂点 **S** (スタート地点に対応) から **G** (ゴール地点に対応) まで最短経路で行く方法を求める、という問題に帰着できる



この問題は、次のような方法で解くことができる

Maze Solver

1. まず頂点 S から頂点 X ($1 \leq X \leq |V|$) までの最短経路 $d[X]$ を求める
2. 頂点 G から「 X が G に隣接し、 $\text{dist}[X] + 1 = \text{dist}[G]$ となる頂点 X ($\text{dist}[G] \geq 1$ の場合存在することが保証される)」を探し、 G に X を代入する、を $\text{dist}[G] = 0$ になるまで続ける
3. 2. で通った頂点が最短経路に使われた頂点である

Maze Solver

これを使うと、 $O(HW)$ で問題を解くことができます
JavaScript を使って実装するとページを作れます

Maze Solver

実は、これだけだと**毎回同じ解が出力される**ので、ランダムに一様な確率で解を出力するためには、どのようにすればいいか？



Maze Solver

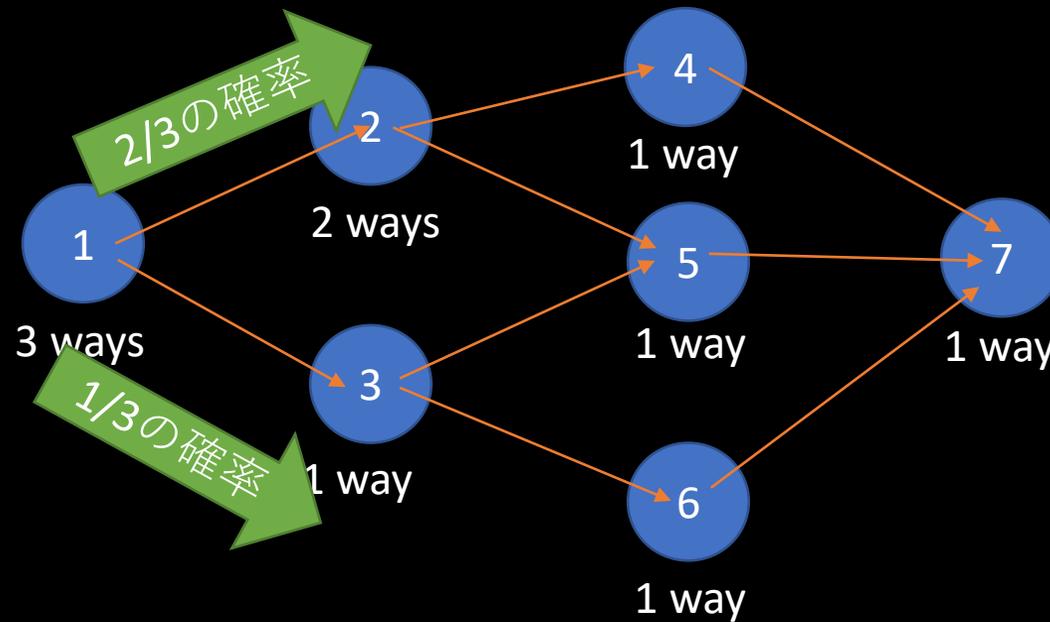
まず、「SからTの最短経路に使われる可能性がある頂点」の集合Vを求めます。これは $\text{dist}(S, X)$ をすべてのXに対して作った後GからDFSをすれば求められます

辺の両端がVに属し、両端のSからの距離が1違うような辺だけを使ったDAGを作り、これはSの入次数が0、Gの出次数が0となります

このグラフはどの辺を選んでも最短経路となることがわかるので、動的計画法を使ってXからGまでの最短経路数を求めることができる

Maze Solver

- あとは、スタートからランダムに移動するだけです



終わりに

これでスライドを終わります

みなさんは僕のウェブサイトを楽しんでください！

<https://square1001.github.io>

