

# 情報メディア専門ユニットI(セミナー) (2018 年度版)

## 1 全般的な注意

この授業の進め方と注意はつぎのとおりである。

1. セミナー開始時にくじを引いて3人のグループを3つ作成しその中で議論を行う。
2. グループ内で検討した結果を1グループあたり10分を目安に最後に発表する。
3. セミナーでは与えられた課題をグループごとに行った議論については詳細なメモをノートにまとめ、その日の発表に生かすこと。
4. ノートは授業終了時に提出するかは今のところ未定である。

この授業で使用するブラウザは Google Chrome とする。他のブラウザでもよいが、ブラウザによっては機能が対応していないこともある。各種ブラウザの表示の違いなどを確認しておくのもよいであろう。

## 2 第1回 — SVGによる基本図形の描画

### 2.1 作業内容とレポート

次の事柄について確認しながら作業し、レポートとして報告すること。今回の内容はテキストの第2章を参考にすること。

1. [www.hilano.org/hilano-lab/](http://www.hilano.org/hilano-lab/)の情報メディア演習 I の資料の中からいくつかの SVG の画像を表示できることを確認する。ブラウザは Chrome で確認すること。
2. 長方形をいくつか色や形を変えて描く。
3. 円や楕円をいくつか色や形を変えて描く。
4. 内部の塗りをグラデーションにする。
5. テキストにある錯視図形の問題をいくつか作成すること。また、それらの一部の属性を変えて見え方がどのように変化するか報告すること。

**レポート課題 1** 次の事柄について報告すること。

1. 各種ブラウザで表示が異なる点があれば報告すること。
2. 本日の演習で利用した要素とその属性についてまとめる。
3. BOM とはなにか。SVG ファイルで BOM があるとなぜいけないのか。
4. 図形が描かれる順序と表示の形について説明する。
5. 作成した図形のついて解説をつける。

**レポート課題 2 (余力問題)** スマートフォンなどの携帯端末のブラウザで SVG の画像が表示できるか調査すること。

## 3 第2回 — SVGにおけるアニメーション

### 3.1 アニメーションに関する注意

アニメーションに関する基本的な事項を確認する。

- アニメーションの種類
  - `animateTransform` 位置を移動する
  - `animateColor` 色を変化させる
  - `animateMotion` 道のりに沿ったアニメーション  
道のりについては次回行うので今回は範囲外とする。
  - `animate` 上記以外の属性を変化させる
  - `set` 値を瞬時に変化させる
- アニメーションの要素はその親要素につくことを忘れないように

```
<rect x="0" y="0" width="100" height="200" fill="red"/>
```

の長方形の `fill` にアニメーションをつけるには`<rect` の最後にある`/>`を`>`に変える必要がある。

```
<rect x="0" y="0" width="100" height="200" fill="red">  
  <animateColor attributeName="fill" attributeType="CSS"  
    from="red" to="green" dur="10s" fill="freeze"/>  
</rect>
```

- いくつかの画像をまとめて移動させたいときは親要素として`<g>`を追加するとよい。
- 不透明度 (配布資料 35 ページ) についても理解すること

### 3.2 課題

次の事柄からいくつか選択して作業すること。

1. 情報メディア演習 (セミナー) の資料からいくつかを選び、各種アニメーションが実行できることを確認する。

アニメーションのキャプチャ画像は開始時 (に近いとき)、途中、終了時の3つを添えること

2. いろいろな図形が移動するアニメーションを作成する。

3. 色のアニメーションで次のことを行う。

- アニメーションの色を変える。
- 複数の図形に別の色のアニメーションをつける
- 図形がある範囲で左右に動き、動きの向きが変わるときに一瞬、色が変わる
- 不透明度にアニメーションをつけて、重なった図形の見え方を変える図形

4. グラデーションのアニメーションで次のことを行う。
  - アニメーションが左から右に流れる。
  - 属性 `x1` と属性 `x2` のアニメーションのスピードを変える
  - 属性 `stop-color` や属性 `offset` にアニメーションをつける
5. 前回作成した図形に、位置の変化、色の変化のアニメーションをつけたものを作成すること。  
1 年の時の Processing で作成したものを SVG で作り直すのもよい。そのときは作成の手間などの比較もするとよい。
6. テキストにある錯視図形のいくつかにアニメーションをつけ、効果の変化を調べること。たとえば次のようなものが挙げられる。
  - Fick の錯視で垂直線を水平線の左端から右端に移動させる (問題 4.1)。
  - Judd の錯視 (問題 4.3) を作成し、両端の線分の回転させて開く角度を変化させるアニメーションをつける。
  - カフェウォール錯視 (23 ページ) の細い線に色のアニメーションをつける
  - 問題 4.11 の図を作成する。
7. (ちょっと面倒) グラデーションにアニメーションをつける図形を<stop>要素のアニメーションだけで作成する

## 4 第3回 — 複雑な図形の描き方、文字列

### 4.1 課題

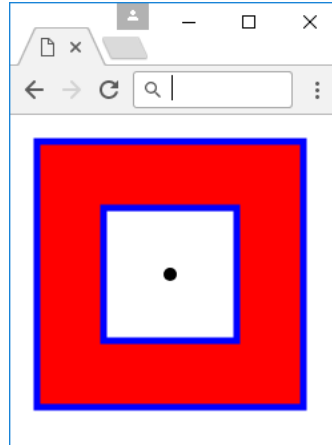
次の中からいくつか選んで報告する。

1. `polyline` または `polygon` を用いた図形を描くこと。たとえば次のようなことを考えてください。

- Excel など他のプログラムを用いて正 6 角形の頂点の座標を計算し、その結果を用いて正 6 角形を描く
- 一つの`<path>`要素で二つの長方形を描く



- 穴が開いた正方形を一つの`<path>`要素で描く



- `<polygon>`要素、`<polyline>`要素と`<path>`要素での描き方の比較をする
- `<path>`要素を用いて曲線を含む図形を作成する
- `path` を用いた図形を描くこと

なお、これらの図形には必要であればグラデーションやアニメーションをつけることが望ましい。

2. 今まで作成した図形の繰り返しがある画像を選んで（なければ今回新たに作成して）`pattern`を用いて塗りつぶすこと。`pattern`を用いなかった場合と比較をして考察に加えること。

- 今までに作成した SVG 画像で繰り返しの部分があるものをパターンで書き直し、コードの長さ、変更の手間などの比較、考察をする。
- ザビニの錯視 (配布資料 46 ページ) や輝くヘルマン格子、モーガンのねじれのひも (配布資料 48 ページ) をパターンを用いて作成する。
- パターンを構成する要素の一部の属性にアニメーションをつける。錯視図形の場合にはその見え方の変化を報告する。

### 3. 文字列を表示すること

- 自分の名前を表示
- それを姓と名前の部分に分け、移動したり、色を変えたアニメーションをつける
- 文字列表示の属性 (配布資料 101 ページ) の属性 **text-anchor** や属性 **dominant-baseline** (表 5.1 を参照) の違いを例とともに報告する。ブラウザによって対応が異なるかもしれない。