

十進BASIC (3) グラフィックス入門

かつらだ まさし
桂田 祐史

2012年5月16日

この授業用の WWW ページは <http://www.math.meiji.ac.jp/~mk/syori2-2012/>
数学のためにコンピューターに計算をさせる場合、

1. 数値計算 (Numerical computation)
2. 数式処理 (Symbolic computation)
3. グラフィックス (Graphics, 可視化 visualization)

が3本柱と言われます。今日は十進 BASIC で簡単なグラフィックスをやってみましょう。

1 連絡事項

- (前回言ったことですが) 前回時間が大幅に足りなくなったので、課題 4B¹ の提出締切りは、当初予定の 5月15日 (火曜) から1週間先に延ばして、5月22日 (火曜) 18:00 とします。
- 課題 4B のレポートを作成する手順 (verbatimfiles.sty をダウンロードして用いる) について、念のため、再度説明します。既に課題 4B を提出した人は、前回の研究課題に挑戦するか (これは必修でなく、気の向いた人のための課題です)、この後の資料を読み進めて下さい。

2 十進BASICでグラフィックス

2.1 イントロ

汎用のプログラミング言語で、その規格の中にグラフィックスを含んでいるもの²は実は珍しく (Java と BASIC くらいしか思い浮かばない…)、十進 BASIC は気軽にグラフィックスをするのにオススメです。

¹<http://www.math.meiji.ac.jp/~mk/syori2-2012/jouhousyori2-2012-04/node7.html>

²多くの場合、特定のグラフィックス・ライブラリ (例えば GLSC) を呼び出して描画することになり、使用する環境に依存したプログラムとなります。

まず十進 BASIC の説明書チュートリアル (印刷配布してある文書) (<http://www.koshigaya.bunkyo.ac.jp/shiraish/basic/tutorial/contents.htm>) の 2.6 節³ を読んでみましょう。

2.2 チュートリアルを忘れた人向け

チュートリアルのコピーを持って来るのを忘れた (持って来て下さい)、という人のために (コンピューターの画面上で見られますが、自分で打ち込む場合は紙を見ながらの方がやりやすいと思います)、「2.6 グラフィックス」の例題プログラムをここに引用しておきます。

例 14 (関数のグラフ)

```
10 DEF f(x)=x^3-3*x+1
20 SET WINDOW -4,4,-4,4
30 DRAW GRID
40 FOR x=-4 TO 4 STEP 0.1
50   PLOT LINES: x,f(x);
60 NEXT x
70 END
```

(要点: 1 変数実数値関数のグラフ $y = f(x)$ の描き方の定跡みたいなプログラム)

例 15 (OPTION ANGLE DEGREES, DRAW GRID())

```
10 OPTION ANGLE DEGREES
20 DEF f(x)=sin(x)
30 SET WINDOW -360,360,-4,4
40 DRAW GRID(90,1)
50 FOR x=-360 TO 360
60   PLOT LINES: x,f(x);
70 NEXT x
80 END
```

(要点: 三角関数の単位を度にする OPTION ANGLE DEGREES と DRAW GRID(,) の使い方)

例 16 (“PLOT LINES” で線を切る)

2つの関数のグラフを描くために、途中で「切る」必要がある。

```
100 DEF f(x)=x^2
110 DEF g(x)=x^3
120 SET WINDOW -4,4,-4,4
130 DRAW GRID
140 FOR x=-4 TO 4 STEP 0.1
150   PLOT LINES: x,f(x);
160 NEXT x
170 PLOT LINES
180 FOR x=-4 TO 4 STEP 0.1
190   PLOT LINES: x,g(x);
200 NEXT x
210 END
```

(要点: “PLOT LINES: x,y;” は線をつなげていくので、170 行目のセミコロンなしの PLOT LINES で線を切る。)

³<http://www.koshigaya.bunkyo.ac.jp/shiraish/basic/tutorial/section2.htm\#2.6>

例 17 (パラメーター曲線)

```
10 OPTION ANGLE DEGREES
20 DEF f(t)=3*COS(t)
30 DEF g(t)=2*SIN(t)
40 SET WINDOW -4,4,-4,4
50 DRAW grid
60 FOR t=0 TO 360
70   PLOT LINES: f(t),g(t);
80 NEXT t
90 END
```

(要点: パラメーター曲線 $x = f(t)$, $y = g(t)$ の描き方の定跡みたいなプログラム)

例 18 (極方程式表示の曲線)

```
10 DEF f(t)=SIN(2*t)
20 SET WINDOW -1,1,-1,1
30 DRAW grid
40 FOR t=0 TO 2*PI STEP PI/360
50 PLOT LINES: f(t)*COS(t), f(t)*SIN(t);
60 NEXT t
70 END
```

(要点: 極方程式 $r = f(\theta)$ は、 $x = f(\theta) \cos \theta$, $y = f(\theta) \sin \theta$ として単なるパラメーター曲線)

例 19 (直交座標から極座標への変換)

```
100 SET WINDOW -4,4,-4,4
110 DRAW grid
120 FOR t=0 TO 2*pi STEP pi/180
130   LET x=cos(t)+1
140   LET y=sin(t)
150   LET r=x^2+y^2
160   LET a=ANGLE(x,y)*2
170   PLOT LINES: r*cos(a),r*sin(a);
180 NEXT t
190 END
```

余談 2.1 (直交座標 ↔ 極座標変換) 例 19 は率直に言って、お手本とするには不適切なプログラムだと思います。 $x = r \cos \theta$, $y = r \sin \theta$ の x, y (直交座標) から r, θ (極座標) を求めるには、

```
R=SQRT(X^2+Y^2)
THETA=ANGLE(X,Y)
```

のようにする、と覚えて下さい。

それはさておき、直交座標から極座標に変換するには、C 言語のプログラムだったら、

```
r=sqrt(x*x+y*y); (または r=hypot(x,y);) そして theta=atan2(y,x);
```

とするところです。 `atan2(,)` という得体の知れない名前の関数よりは、`ANGLE(,)` の方が覚えやすく良い (BASIC の規格の方が趣味が良い)、と私は思います。 ■

メモ: hypotenuse (直角三角形の) 斜辺

2.3 まとめ

- `SET WINDOW x_1, x_2, y_1, y_2` は座標系の設定
- `DRAW GRID` はグリッド (格子、網目という意味) を描く
`DRAW GRID(x, y)` というようにグリッドの幅も指定できる
- `PLOT LINES: x, y` ; 次に実行される PLOT LINES 命令で指定される点との間を線分で結ぶ。線を区切るには、単に `PLOT LINES` とする。
- 1変数関数のグラフ、2次元パラメーター曲線は描くのが簡単 (例題プログラム参照)。
- 他に
 - 点を描く `PLOT POINT: x, y` 命令
 - 文字を描く `PLOT TEXT ,AT x, y : "文字列"` 命令
 - 多角形領域を塗り潰す `PLOT AREA: $x_1, y_1; \dots; x_n, y_n$` 命令

などもある⁴。これらの命令の使い方についてはオンライン・ヘルプを参照。

- 色も指定できる。

`SET POINT COLOR c` , `SET LINE COLOR c` , `SET AREA COLOR c` , `SET TEXT COLOR c` 等々。ここで c は色番号で、以下のように定義されている。

0 白, 1 黒, 2 青, 3 緑, 4 赤, 5 水色, 6 黄色, 7 赤紫, 8 灰色, 9 濃い青, 10 濃い緑,
11 青緑, 12 えび茶, 13 オリーブ色, 14 濃い紫, 15 銀色, …

- グラフのウィンドウ (なんとか .BMP のような名前がついている) の「ファイル」メニューで、「名前をつけて保存 (A)」から描いたグラフィックスをファイルに保存できますが、「ファイルの種類 (T)」で、BitMap でなく、JPEG や GIF フォーマットを選ぶことを勧めます (画像ファイルはサイズが大きくなりがちで、特別の理由がない限り、何らかの方法で圧縮したフォーマットを選ぶべきです、TeX に取り込む場合は JPEG が良いと思います)。

圧縮する場合に、現在使用している Windows の「名前をつけて保存」メニューでは、圧縮率 (≡画像の品質) の設定が出来ないので、一度 BitMap (24ビット) で出力してから、適当な画像処理ソフトを用いて適当な圧縮率で JPEG や GIF に変換する、というやり方は考えられる。

3 レポート課題5A

十進 BASIC で円を描く⁵プログラムと描いた図を送って下さい (図の形式は JPEG にして下さい)。色はお好みで。

⁴area は数学語としては「面積」ですが、「範囲」、「地方」などの意味があるわけですね (知っていると思いますが)。text は「本文」、「教科書」、「文章」。

⁵昔、パソコンで BASIC 華やかなりし頃は「(楕)円を描く命令」が備わっていました。でも十進 BASIC にはありません。ないけれども“簡単に描ける”はず。とりあえずノーヒントにします。

提出は、SNS「第5回 (5/16)」⁶ への書き込みで行って下さい。写真1に円の画像ファイル (kadai5a.JPG), ファイルにプログラム kadai5a.BAS を添付します。

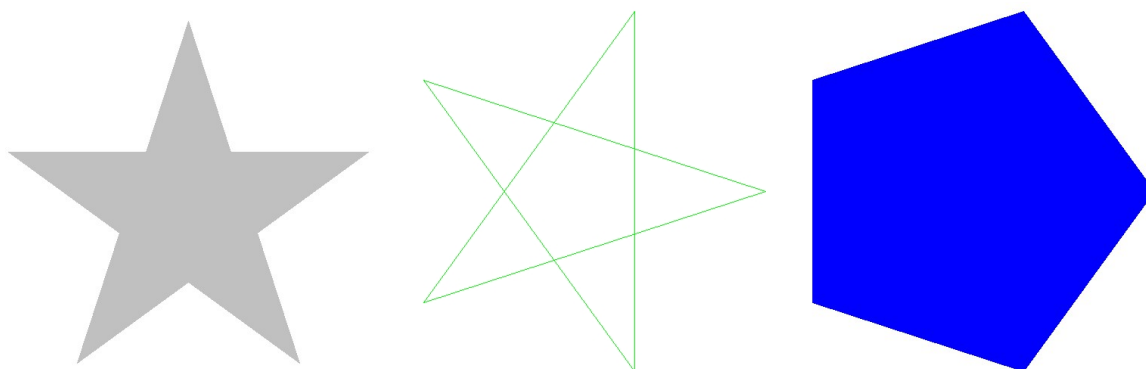
これは出席代りで、なるべく今日 (5月16日) 授業時間中に書き込むこと。

4 レポート課題5B

次の (1), (2) のいずれかを行う BASIC プログラムを作成し、そのレポートを送って下さい。

BASIC プログラムとその簡単な説明、そのプログラムを実行して描いた図を含んだ TeX 文書を作成し、PDF に変換したものを kadai5b.pdf を Oh-o! Meiji で送って下さい。締め切りは (一応) 5月29日 (火) 18:00 とします。

- (1) 塗り潰した星を描く (図の星形の中を銀色 — と指定したいところだけれど、光沢は難しいので、自分の好きな色に塗る)。

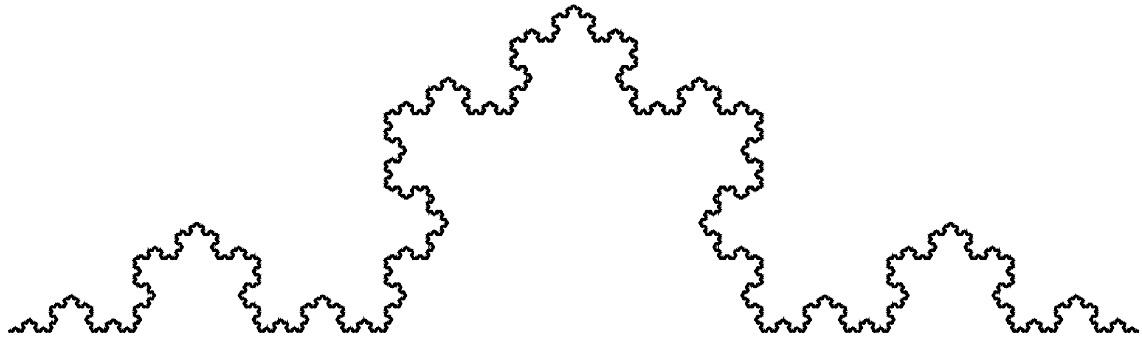


— 大変だったら塗り潰しできなくても構いません。また右図のように五角形を塗るのでも受け付けます (でもなるべく頑張りましょう)。PLOT AREA 以外に PAINT (A.1 参照) や MAT PLOT AREA などの命令 (A.2 参照) が利用できます (PAINT を使うのが簡単ですが、PAINT のような便利な命令は使えない場合も多いので、出来れば PLOT AREA や MAT PLOT AREA を使って描くように努力してみてください)。

この問題にどのように取り組めば良いか、固まってしまう人も多いかと思いますが、「頂点」の座標が分かれば後はちょっと頑張れば良く、それをするために **しばらく紙と鉛筆を使って考える** ことを勧めます (高校レベルの数学ですが、暗算でやるのは大変でしょう)。この講義はかなりの部分 “ペーパーレス” で乗り切れますが、ある程度込み入ったこと (特に図形の問題) を考えるのには、自分で書いて (or 描いて) みるのが一番です。

- (2) 自己相似図形として有名な Koch 曲線 (の近似) を描く。

⁶http://sns.math.meiji.ac.jp/?m=pc&a=page_c_topic_detail&target_c_commu_topic_id=1387



有名なので探せばそのものズバリのプログラムも見つかるでしょうが、自分で考えて見ませんか？サブルーチンの再帰呼び出しを使えば、意外に簡単に描けると思います。(Koch 曲線を知らないという人には…ウィキペディアで「コッホ曲線」を調べてみるとか)

画像を $\text{T}_\text{E}\text{X}$ 文書に取り込む方法については、「画像の取り込み」⁷ を参考にして下さい。今回の場合の手順は

1. 十進 BASIC のグラフィックスのウィンドウで [ファイル] メニューの [名前をつけて保存 (A)] で、JPEG 形式で保存する。ここでは `kadai5b.JPG` という名前のファイルになったとする。
2. 情報処理 2 用のフォルダに作った [コマンドプロンプト] を開いて、

```
Z:\¥.windows2000¥syori2>jpeg2ps kadai5b.JPG > kadai5b.eps
```

とコマンドを実行する。これで `kadai5b.eps` というファイルが出来るはず。

3. $\text{T}_\text{E}\text{X}$ 文書 (`kadai5b.tex`) のプリアンブル (`\begin{document}` の前) に `\usepackage[dvips]{graphics}` と書き、グラフを取り込みたいところで、例えば

```
\begin{center}
\includegraphics[width=8cm]{kadai5b.eps}
\end{center}
```

とする (図を取り込むような場合、本当は `figure` 環境を使うのが良いと思いますが、説明をサボりたいので、上の例では `center` 環境ですませました)。

もう一回くらい、お手本を見せます (そろそろ `.tex` を見せるのは止めようかな、と思っ

⁷<http://www.math.meiji.ac.jp/~mk/syori2/jouhousyori2-2012-02/node34.html>

kadai5b.tex

```
\documentclass[12pt]{jarticle}
\usepackage[a4paper]{geometry}% 好みの問題
\usepackage{amsmath,amssymb}% 今回は不要かも
\usepackage{verbatimfiles}% 今回これが必要
\usepackage[dvipsout]{graphicx}% 今回これが必要
\begin{document}
\title{情報処理2 課題5Bレポート}
\author{2年16組99番 数学 学}
\date{2012年5月16日}
\maketitle

\section{プログラム}

\verbatimfile{kadai5b.BAS}

\section{プログラムの説明}

(どのように考えたか、説明する。例えば頂点の座標はこんな式で表せるとか。)

\section{プログラムの実行結果}

kadai5b.BAS の実行結果は次のようになる。

\begin{center}
\includegraphics[width=8cm]{kadai5b.eps}
\end{center}

\end{document}
```

A 補足: 色を塗る命令 PAINT, MAT PLOT AREA

A.1 PAINT

Windows 版の十進 BASIC には、PAINT という命令があります。PAINT x, y とすると、点 (x, y) を含んだ「囲まれている領域」を塗り潰します。塗る色は `set area color` で指定できます。

次のプログラムでは、最初に三角形の周を描いてから、内部の点 (ここでは $(1, 1)$) を指定して、塗り潰しています。

三角形を描いて内部を赤で塗りつぶす

```
REM testpaint.bas --- paint の例 (Windows 版でのみ利用可能)
SET WINDOW -1,3,-1,3
DRAW grid
PLOT LINES: 0,0;2,1;1,2;0,0
SET AREA COLOR "red"
paint 1,1
END
```

最後の `PAINT 1,1` の代わりに `PAINT 0,1` とすると、三角形の外部を塗り潰します。

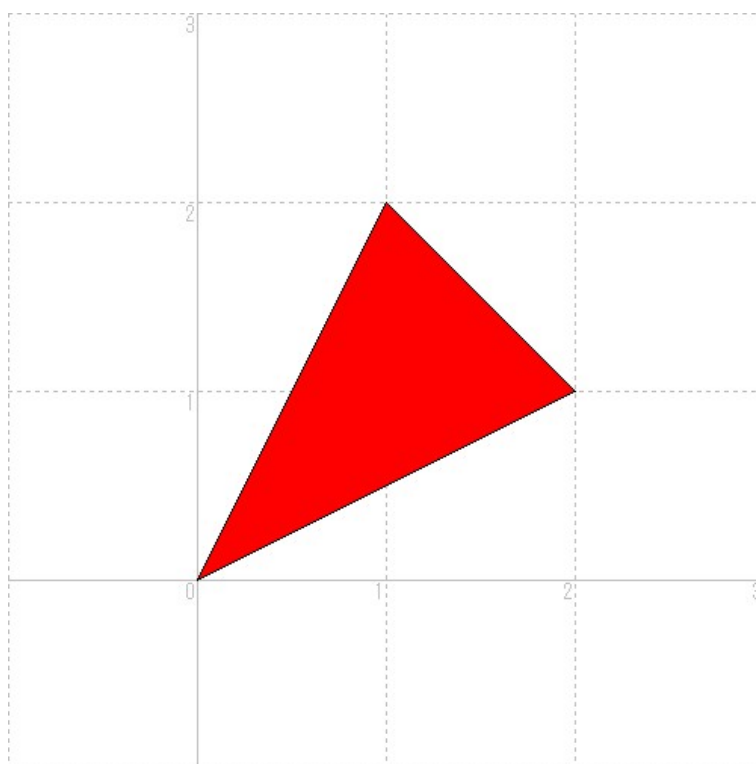


図 1: testpaint.BAS — PAINT で塗る

残念ながら、この PAINT は JIS 規格外の命令であり、いわゆる方言です (例えば Linux 版十進 BASIC にはこの命令はありません)。昔のパソコン BASIC では PAINT に類する命令があるのが普通でしたが、むしろグラフィックス・ライブラリでこの種の命令を持っている方が珍しいと思います⁸。

A.2 MAT PLOT AREA

多角形の頂点の座標を配列変数に記憶させておけば、命令1つで多角形内部を塗り潰すことができます。詳しくはオンライン・ヘルプを参照して下さい。

⁸コンピュータの画面表示の仕掛けには、VRAM (video RAM) が使われていることが多く、VRAM は自由にアクセス (書くだけでなく読むこともできる) できるので、比較的簡単に PAINT の機能が実現できます。そのため Windows にその命令が用意されているけれど、汎用性はなくなる、ということです。

赤い三角形を描く

```
REM testmatplotarea.bas --- mat plot area あ
DIM x(3),y(3)
LET x(1)=0
LET y(1)=0
LET x(2)=2
LET y(2)=1
LET x(3)=1
LET y(3)=2
SET WINDOW -1,3,-1,3
DRAW grid
SET AREA COLOR "red"
MAT PLOT AREA: x,y
END
```

MAT PLOT AREA: x,y で

PLOT AREA: x(1),y(1);x(2),y(2);x(3),y(3)

と同じ意味になります (というわけで、PLOT AREA のオンライン・ヘルプも適宜参照して下さい)。— 三角形ならば PLOT AREA でも良いけれど、星形でやるのは (点が多いので) 面倒ですね。

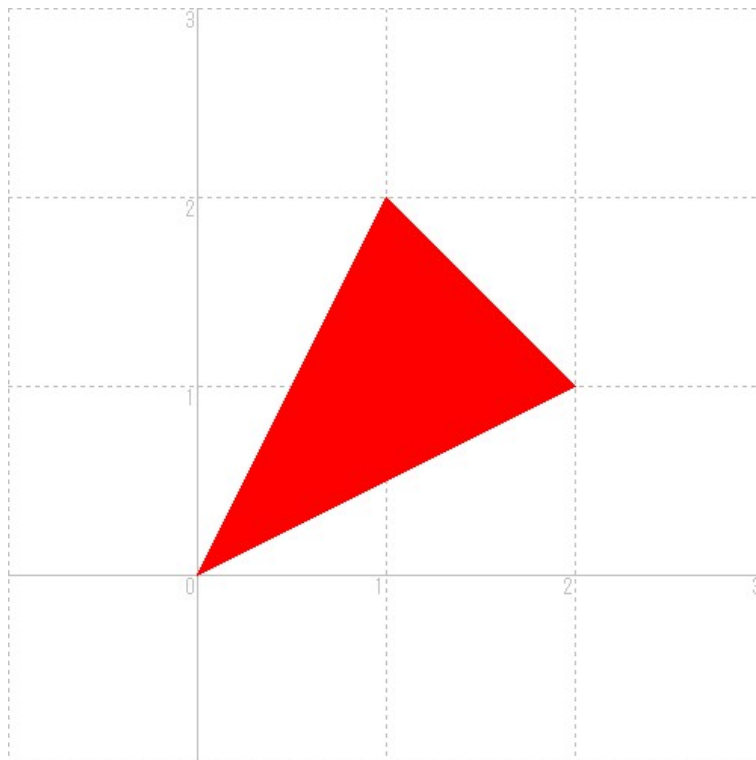


図 2: testmatplotarea.BAS — MAT PLOT AREA で塗る

B 課題5Bへの一つのヒント：タートルグラフィックス

一時期 (今となっては古い話です)、プログラミングの入門用の言語として、LOGO⁹ というものが注目されたことがあります。その特徴である**タートル・グラフィックス**を利用すれば、まだ座標系の概念を学んでいないような小学生にも簡単に図形を描くプログラムが作れるということで「ちょっと面白い」話でした。

タートル・グラフィックスとは、キャンバスの中を亀 (タートル) が歩く軌跡として図形を描画するというものです。亀に伝えられる指令として、「進む (指定された歩数だけ)」、「戻る (指定された歩数だけ)」、「右に曲がる (指定された角度だけ)」、「左に曲がる (指定された角度だけ)」、「ペンを下ろす」、「ペンをあげる」などがあります。

例えば、「50 歩進み、90° 右に曲がる」を4回繰り返すことで1辺の長さが50歩の正方形を描くことができます。

⁹<http://ja.wikipedia.org/wiki/LOGO>

```

REM TURTLE.BAS --- 亀よ正方形と正三角形を描いておくれ
OPTION ANGLE DEGREES
REM right(),left(),walk(),jump() の定義
SUB right(t)
    LET direction=direction-t
END SUB
SUB left(t)
    LET direction=direction+t
END SUB
SUB walk(s)
    PLOT LINES: xp,yp;
    LET xp=xp+s*COS(direction)
    LET yp=yp+s*SIN(direction)
    PLOT LINES: xp,yp
END SUB
SUB jump(s)
    LET xp=xp+s*COS(direction)
    LET yp=yp+s*SIN(direction)
    PLOT LINES: xp,yp
END SUB
REM ----- start -----
LET L=100
SET WINDOW -1.1*L,1.1*L,-1.1*L,1.1*L
REM 亀の現在位置 (xp,yp), 方向 direction (x軸となす角)
LET direction=0
LET xp=0
LET yp=0
FOR i=1 TO 4
    CALL walk(L)
    CALL right(90)
NEXT i
CALL right(10)
SET LINE COLOR "red"
FOR i=1 TO 3
    CALL walk(L/2)
    CALL right(120)
NEXT i
END

```

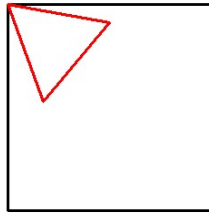


図 3: タートルグラフィックスで正方形と正三角形を描く

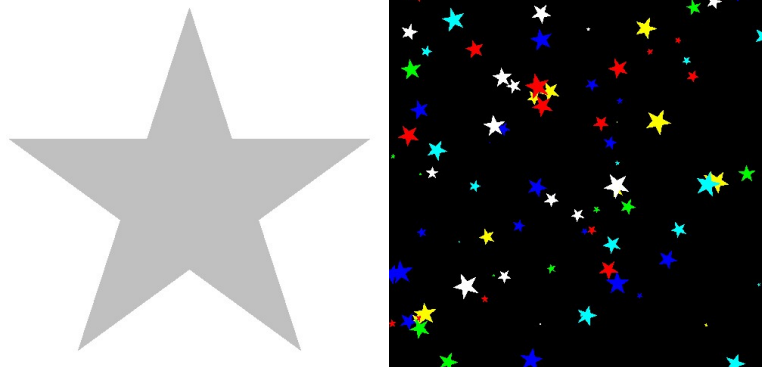


図 4: 星 (銀色に見えると良いけれど灰色ですね&ランダムに散らす)