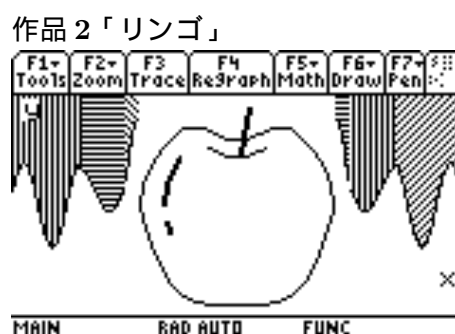
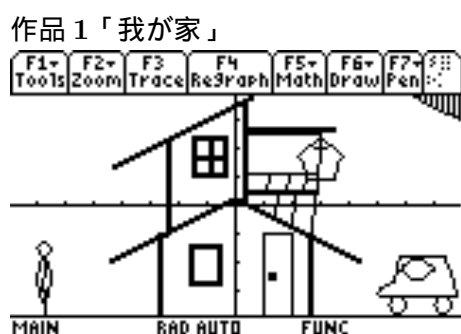


関数グラフアート — 新たなる出発

石川高専 阿蘇和寿

石川高専は、全国関数グラフアートにほとんど毎年出品し、優秀な成績を収めてきたといってもよいと思う。最近の2年間では、環境都市工学科の萩澤さくらさんが特別賞、最優秀賞と連続して賞をいただくことができた。それらはともに媒介変数を用いたものであったので、今年は、1年生を対象に、 $y = f(x)$ 型のグラフだけによるグラフアートを課した。その中で、次の2作品を取り上げてみたい。(作品名はともに筆者による)



1 グラフアートを課した状況

関数グラフアートの課題を課した状況は次の通りである。

- 時期：第1学年7月
- 対象クラス：電気工学科41名、電子情報工学科41名
- 提出期間：1週間

この課題に取り組むために配布したプリントについては§2に示すことにし、§1では、まず、この作品に使われた関数を示しながら学生たちの活動を探っていきたい。

1.1 作品1「我が家」

(§2で示すように) この関数グラフアートでは、8個の関数を指定し、それらを拡大縮小、平行移動、対称移動することによって描画する、ということになっているが、この作品はその意図に沿ったものではなく、技術的にも高い水準とはいえない。

しかしながら、この作品は残る91個の関数をすべて使ってあり、非常に丁寧に粘り強く描いていることは驚くばかりである。

(1) 1階の屋根を描くための直線 (style: thick)

$$y = \frac{1}{2}x + 0.25 \mid -4.4 \leq x \text{ and } x \leq 0, \quad y = -\frac{1}{2}x + 0.25 \mid 0 \leq x \text{ and } x < 4.5$$

- (2) 1階の壁を描くための垂直線を3次関数で実現

$$y = (-30x + 80)^3 \mid 2.6 < x, \quad y = (30x + 80)^3 \mid x < -2.6$$

- (3) x の範囲指定に工夫して1階のドアの水平線を定数関数で描く。

$$y = \frac{-x}{x} \mid 1.1 < x \text{ and } x < 2$$

- (4) 2階の壁(右側)を描くために、突如として三角関数を使用

$$y = 1.7 \sin(100000x + .5) + 1.8 \mid .2 < x \text{ and } x < .5$$
$$y = 1.7 \sin(-100000x + .5) + 1.8 \mid .2 < x \text{ and } x < .5$$

- (5) 右にある車の窓を2次関数で描画

$$y = (x - 6.4)^2 - 2.6 \mid 5.9 \leq x \text{ and } x \leq 7, \quad y = -(x - 6.4)^2 - 1.9 \mid 5.8 \leq x \text{ and } x \leq 7$$

- (6) 右上部の雲のようなものを分数関数で描く (style:Above)

$$y = 4 \cdot \frac{6}{x} \mid 0 \leq x$$

- (7) 車のボンネットとフロントガラスの描画のために無理関数を使用

$$y = \frac{1}{2} \sqrt{x - 5} - 3.1 \mid x \leq 5.5, \quad y = 2\sqrt{x - 5.6} - 2.7 \mid x \leq 5.8$$

これらの関数を選んだ理由はよく分からない。おそらく、学んだすべての関数を使うことを目指したのであろう。

作者の感想「グラフの範囲指定が大変だった」

1.2 作品2「リンゴ」

見て分かるとおり、絵心がある学生であろう(本人も絵が好きであるといっていたが)、リンゴの曲線が綺麗である。これをどう表現するのだろうか。

これは指定された関数を用いて描かれた。1桁の番号の関数は全員に与えておいたものである。

- (1) リンゴ下部の曲線は次関数で実現 (x 軸方向の拡大を使用)

$$y3 = x^4, \quad y21 = y4 \left(\frac{1}{2.5}x - 0 \right) - 3.5 \mid -3.4 \leq x \text{ and } x \leq 3.4$$

- (2) リンゴの上部の左右の曲線を無理関数で実現

$$y7 = \sqrt{x}, \quad y22 = y7(5x + 17) \mid x \leq -2 \quad y23 = y7(-5x + 17) \mid x \geq 2$$

- (3) リンゴの上部を三角関数でつないで輪郭が完成

$$y8 = \sin x, \quad y24 = \frac{1}{4}y8(2x - 1.6) + 2.5 \mid x \geq -1.9 \text{ and } x \leq 1.9$$

- (4) それを平行移動して枝の付け根のくぼみを表現

$$y25 = \frac{1}{4}y8(2x - 1.6) + 1.9 \mid x \geq -1 \text{ and } x \leq 1$$

(5) 枝自体は3次関数の一部を使用 (style : thick)

$$y2 = x^3, \quad y26 = y2(x - 1.6) + 5.1 \mid x \geq 0.1 \text{ and } x \leq 0.4$$

(6) 光が当たっている部分の上は無理関数 (輪郭部の平行移動と範囲制限), 下は4次関数を使用

$$y3 = x^4, \quad y28 = y3 \left(\frac{1}{2.5}x - .3 \right) - 3.5 \mid -2.5 \leq x \text{ and } x \leq -2.4$$

(7) 葉の部分は三角関数を左右それぞれ3つずつ使用 (style:Above)。 $y8 = \sin x$ から例題1 (§2) で作った関数を使っている。周期が2の方が範囲指定しやすかったのだと思われる。

$$y19 = 3 \sin \pi x, \quad y19 = \frac{1}{2}y19(x - 2) \mid x \leq -5.5$$

以上, 使用した関数の個数は14個。作品1と比べると手際の良さが目立つ。
本人の感想「みんないろいろなアイデアがあって面白いなと思いました」

2 「初めての関数グラフアート」解説プリント

以下に, この課題のために配布したプリントを掲載しておく。

これで分かるように, この課題の目的は次の2点である。

- (1) べき関数, 分数関数, 無理関数, 三角関数 (正弦) の基本的な形を覚える。
- (2) 平行移動, 拡大縮小, 対称移動が自由にできるようになる。

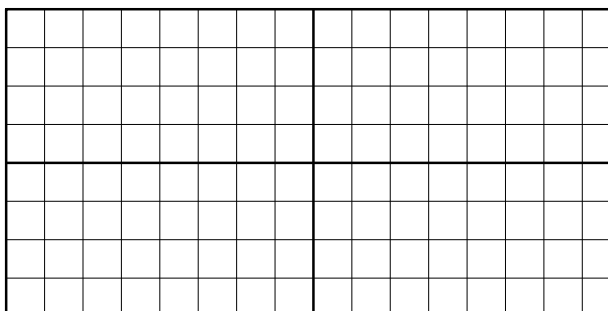
作品2がこちらの意図したとおりの作品となった。授業は1駒 (100分) を用いて解説をした。それ以前には授業の中でグラフ描画の練習をしている。

何となく書いているうちに面白い形になった, ということでは意味が薄いので, グラフアートを行うにあたって §2.8 に示した「設計図」を作らせている。

2.1 WINDOW の設定

GRAPH 画面で f2 Zoom 4:ZoomDec を選ぶと WINDOW の設定は次のようになる。

```
xmin=-7.9
xmax=7.9
xscl=1.
ymin=-3.9
ymin=-3.9
yscl=1.
xres=2.
```



ここでグラフの滑らかさ $xres=1.$ と設定を変えておく。次のことを覚えておくこと。

$xres=1.$: グラフを描くとき $x = 0.1$ ごとに点を打つ

2.2 基本的な関数

ここでは使える関数を 4 種類, 8 個に制限する。[Y=] 画面で次を入力する。

(1) $y1 = x^2$

(2) $y2 = x^3$

(3) $y3 = x^4$

(4) $y4 = x^{99}$

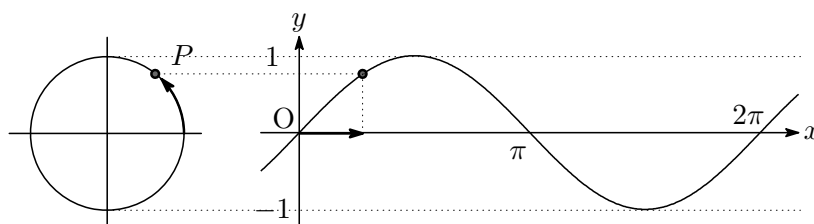
(5) $y5 = x^{100}$

(6) $y6 = \frac{1}{x}$

(7) $y7 = \sqrt{x}$

(8) $y8 = \sin x$

$y8 = \sin x$ はサイン関数といい, 点 P が, 点 (1,0) を出発して, 原点を中心として半径 1 の円周上を, 速さ 1 で反時計回りに運動するとき, 出発してから x 秒後の点 P の y 座標を表す。1 周するのに 2π 秒かかるから, グラフは次のようになる。



2.3 描画する・しない

関数を描画するかしないかは, [Y=] 画面で [f4] (check) によって区別する。

$y1$ から $y8$ まではチェックをはずしておく。

2.4 関数の移動と変形

関数のグラフは, 8 個の基本的な関数の平行移動, 対称移動, 拡大・縮小を行って作る。

2.1 (グラフの平行移動) $y = f(x)$ のグラフを x 軸方向に p , y 軸方向に q だけ平行移動したグラフの方程式は

$$y = f(x - p) + q$$

2.2 (グラフの対称移動) $y = f(x)$ のグラフを対称に移動したグラフの方程式は次の通りである。

- (1) $y = f(-x)$: y 軸について対称
- (2) $y = -f(x)$: x 軸について対称
- (3) $y = -f(-x)$: 原点について対称

2.3 (グラフの対称移動) $y = f(x)$ のグラフを拡大・縮小したグラフの方程式は次の通りである。

- (1) $y = af(x)$: y 軸方向に a 倍
- (2) $y = f(ax)$: x 軸方向に $\frac{1}{a}$ 倍

例題 1 (グラフの移動と変形) 次のグラフを描き, 基本的な関数をどのように移動, 変形したか述べよ。たとえば (1) の場合, $y = y1(x)$ と $y = y9(x)$ とを比較・観察すること。

- | | |
|---------------------------------------|---------------------------------------|
| (1) $y9 = y1(x + 1) - 3$ | (2) $y10 = y2(-x), y11 = -y2(x)$ |
| (3) $y12 = y3(10x), y13 = 10 * y3(x)$ | (4) $y14(x) = y4(x/3), y15 = y4(x)/3$ |
| (5) $y16(x) = y5(2x) - 3$ | (6) $y17(x) = y6(-x) + 1$ |
| (7) $y18(x) = y7(2x - 5)$ | (8) $y19(x) = 3 * y8(\pi * x)$ |

2.5 描画方法など

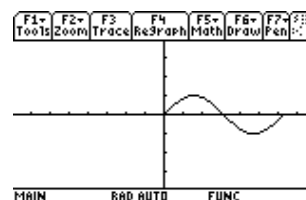
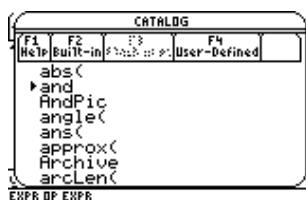
Y= 画面で **F1 Tools** 9:Format で描画方法などが設定できる。

Coodinates	...	RECT
Graph Order	...	SEQ
Grid	...	OFF
Axis	...	ON・OFF
Leading Cursor	...	ON
Label	...	OFF



2.6 描画範囲の制限

描画範囲の制限は不等号を使う。

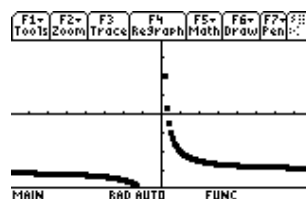


and は2つの条件を同時に満たすことを意味する記号で, **CATALOG** の中に入っている。カタログで探すときは頭文字のアルファベットがあるキーを押す。

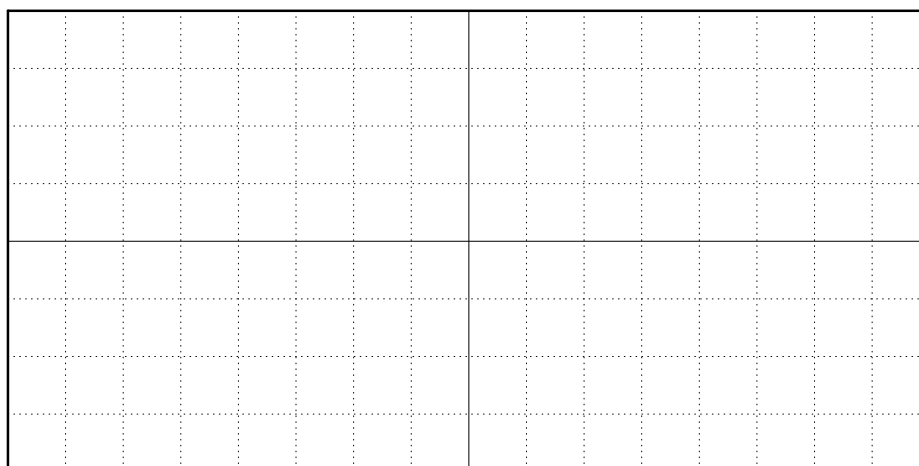
等号付き不等号は, , または

2.7 スタイルの使用

Y= 画面で, スタイルを適用したい関数にカーソルを合わせ, **F6 Style** とすることで, グラフの線種を選ぶことができる。



2.8 設計図を書いてみよう



3 おわりに

この課題の時点ではまだグラフ電卓にも馴染みが薄く、どのようにしたらよい作品ができるかも分からなかったものと思う。授業時間も十分ではなく「来週までに提出」という期限付きであった。

提出後にすべての作品をプリントアウトして全員に配布したので、それぞれ考えるところがあるのではないかと思う。2作品目を夏休みの課題としたので、発表時点では新たな作品も提示する予定である。