

Geometric Constructor マニュアル vol. 2

(作図の手引き)

0 . はじめに

Geometric Constructor は作図のための道具である。手で図を描く (drawing) よりも、もっと精確 (全く正確というわけではない) で、そしてその図を動かすことができる道具である。しかし、その反面、手で描くより、少し考えながら使わなければならない。

手で描くよりも少し正確に作図したいときは、定規やコンパスを使うのが普通である。定規やコンパスを使うには、いくつかの基本的な使い方を学ばなければ、使いこなせるようにはならない (例えば、平行線の引き方)。Geometric Constructor を使いこなすには、同じようなことが必要なのである。

定規やコンパスによる作図と比べると、Geometric Constructor による作図は、便利なところもある。定規やコンパスで、例えば垂直二等分線を引くときには、一見そのことには関係のない線も引きながらでないと、目的の直線を引くことはできない。しかし、Geometric Constructor の場合、もう少し直接的に、「垂直二等分線を引く」という思考に対応した形で作図をすることができる。

初めてGeometric Constructor に接する方は、まずいろいろな例を読み込み、変形などを使ってみた後で、まず1章の補助線の追加をしてみてください。Geometric Constructor での作図とはどういうものが大体理解できるはずである。作図を始めから行うには、最初に図を分析しておく必要がある。とは言っても、補助線が簡単に追加できるようになれば、最初からの作図であっても、それほど難しくなくできるはずである。そういう心構えで2章の要点を読んで頂ければ十分である。

静的な図を作るという意味では、2章までで十分だが、図形を「動かして調べる」ということになると、それを出発点として、様々なことが可能である。そしてまた、それを前提とすると、様々なノウハウが副次的に生まれてくる。3章以降では、そのようなことについて、いくつかの観点からまとめてみた。必要に応じて読んでいただければ、十分である。

目 次

- 0 . はじめに
- 1 . 補助線の追加
 - 1.1 基本的な考え方と手順
 - 1.2 練習問題(1) 三角形に対する追加
 - 1.3 練習問題(2) 四角形に対する追加
- 2 . 標準的な作図
 - 2.1 白紙の状態から作図を始めるときの手順
 - 2.2 分析について (標準メニューの中の「分析」)
 - 2.3 練習問題
- 3 . 点の動きの制限
 - 3.1 「点の動きの制限」とは
 - 3.2 基本的な考え方と作図方法 - 事前に準備 -
 - 3.3 「後から」修正するための方法と原則
 - 3.4 練習問題
- 4 . 基本的な編集機能
 - 4.1 色の設定
 - (1) 目立たせたい
 - (2) 隠したい
 - (3) 基本的な手続き
 - 4.2 名前の設定
 - 4.3 点の属性の設定
 - 4.4 直線系の属性の設定
- 5 . 軌跡による図形の描画
 - 5.1 数学での「軌跡」とGeometric Constructor での軌跡
 - 5.2 最も簡単な使い方
 - 5.3 設定の方法
 - 5.4 練習問題
 - 5.5 直線等の掃過領域がなす図形
- 6 . 変数・数式の利用
 - 6.1 円の半径を変える
 - 6.2 一緒に変化させる

- 6.3 数式を変化させる
 - 6.4 練習問題

 - 7 . 条件を満たす軌跡による図形の描画
 - 7.1 例
 - 7.2 OHPシート作戦
 - 7.3 「一人」でするにはたまらない
 - 7.4 アイデアとしての「領域の境界」
 - 7.5 Geometric Constructor における「条件を満たす点の集合」の機能
 - 7.6 ケーススタディ - 円周角の場合 -
 - 7.7 練習問題

 - 8 . 軌跡によるグラフの描画
 - 8.1 関数を調べる手段としてのGeometric Constructor
 - 8.2 大きさを視覚化する
 - 8.3 グラフとして表現する
 - 8.4 練習問題

 - 9 . 変換を調べるための作図
 - 9.1 変換を調べる手段としてのGeometric Constructor
 - 9.2 別の例
 - 9.3 「点と点の対応」はすべて変換と言える
 - 9.4 練習問題

 - 10 . 複素数平面上的関数としての作図
 - 10.1 平面は複素数
 - 10.2 少し「ズーム」して「保持」しておくと便利
 - 10.3 「和」を調べる
 - 10.4 「積」を調べる
 - 10.5 「逆数 $1/Z$ 」を調べる
 - 10.6 複素数の関数（多項式）の性質を調べる
 - 10.7 練習問題

 - 11. 様々なソフトでの作図との比較
 - 11.1 Logo との比較
-

1 . 補助線の追加

1.1 基本的な考え方と手順

最も基本的な作図は、「補助線の追加」である。Geometric Constructor では、次のような考え方を前提としている。

(1) どういう種類の幾何的対象を作図したいのか。

(点, 直線, 線分, 半直線, 円のどれか)

(2) それはどういう手続きによって作図できるのか。

(例, 線分の2等分 点, 垂直2等分線 直線 など)

(3) その手続きは、すでにできている他の対象のどれとどれを元にしたらよいか。

このステップを追ってメニューを選択していけばいい。例として、円周角の定理の図に補助線を入れる場合を考えてみよう。

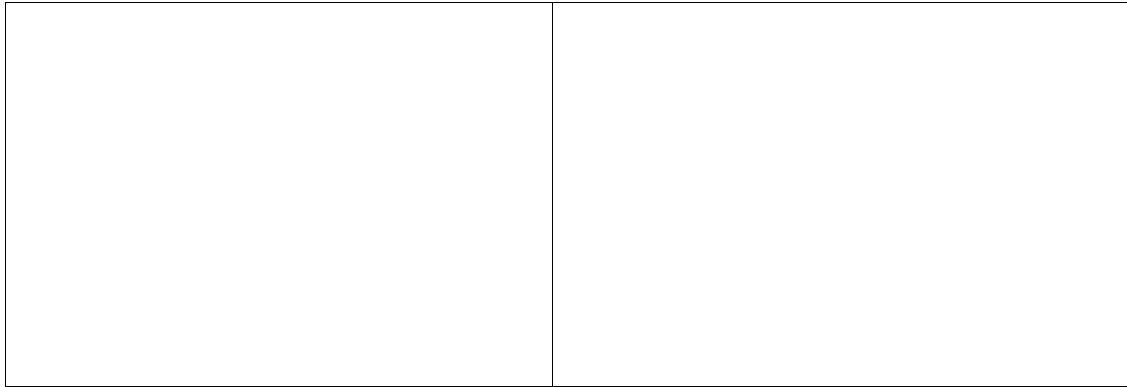


図 - 1.1 元の図

図 - 1.2 完成予定図

元の図を読み込むため、まず、「例」の中の「円周角」を選択する。

そして、補助線を作図するために、「作図の継続」を選択する。

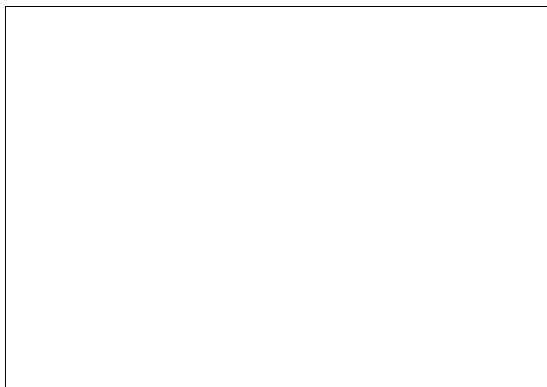


図 - 1.3 作図の継続の選択

(1) 補助線として書き込みたいものは、「半直線」である。「半直線」を選択する。

(2) この半直線は2点を結ぶことによってできるので、「2点」を選択する。

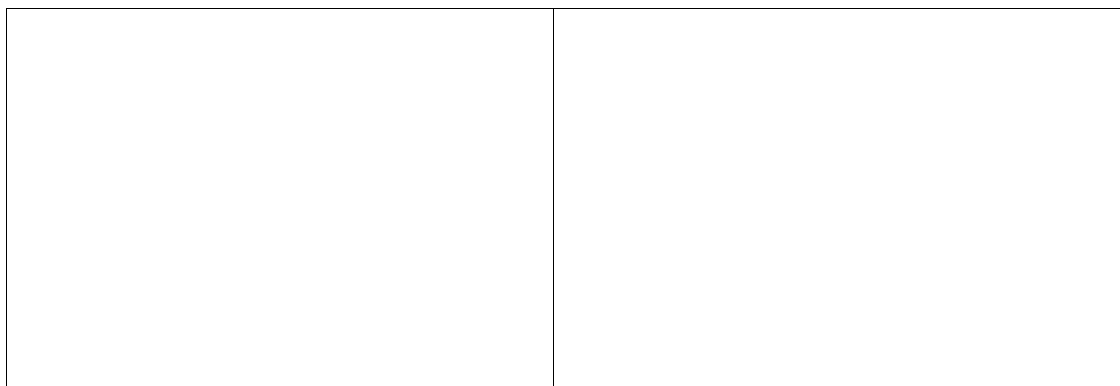


図 - 1.4 半直線の選択

図 - 1.5 「半直線（2点）」の選択

(3) この場合、点Eと点Dとを結べばよいので、 と を使って、候補になる点の中から、点D、Eを選択する。すると、補助線が作図される。

あるいは、マウスによってD、Eを選択してもよい。このとき注意しなければならないのは、マウスの使い方である。キーボードでの「選択候補変更」の意味の「 」としての機能と、「実行」の意味の「リターン」の両方をマウスの左ボタンは兼ねている。そのため、

- ・反転している対象を選択する場合は、「実行」を、
- ・反転していない対象を選択する場合は、「選択候補変更」を意味する。

つまり、後者の場合には、「もう一度クリックする」必要がある。

そういう意味で、Geometric Constructor でのマウスでの選択は、多くの場合、「2回クリックする必要がある」と考えた方がいい。

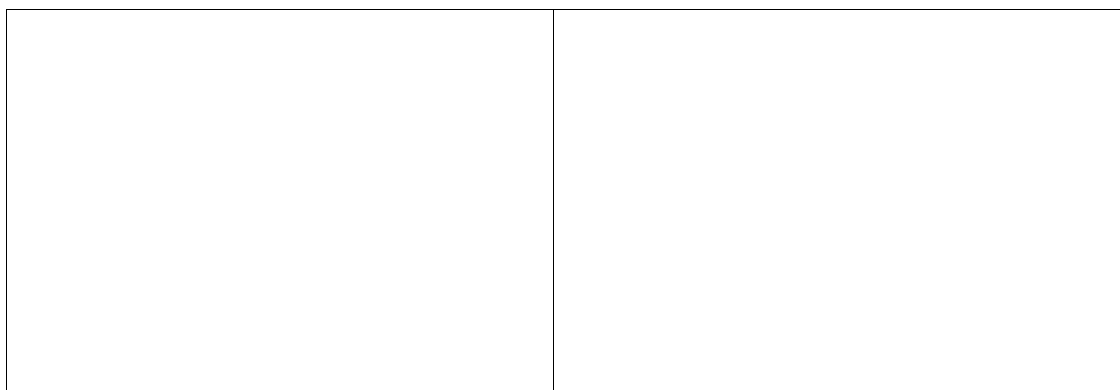


図 - 1.6 点の選択

図 - 1.7 図の完成 (1.2 再掲)

この手順は、Geometric Constructor での作図において、最も基本的である。

1.2 練習問題(1) 三角形に対する追加

上述のような手続きを1回あるいは、何回か使って、左図を右図にせよ。元にするファイルは「三角形」である。これを読み込んで、その図に補助線を書き入れよ。

(1) 外心

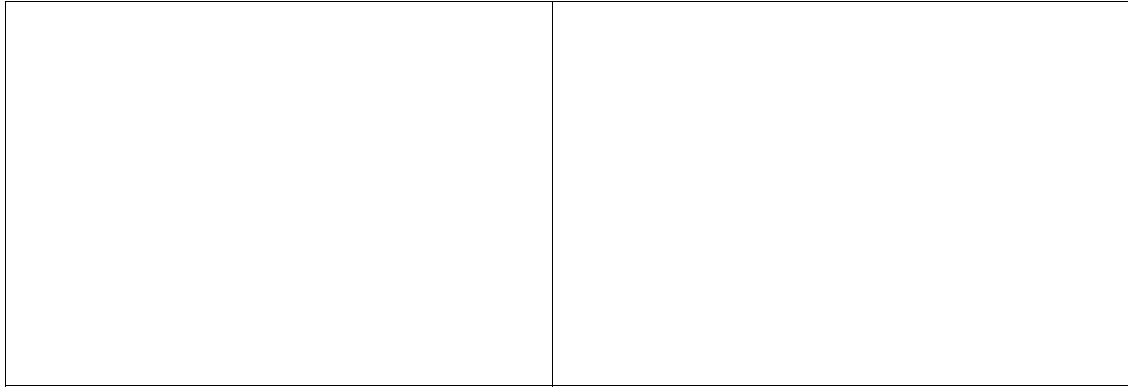


図 - 1.8 三角形の図

図 - 1.9 外心の作図

(2) 内心

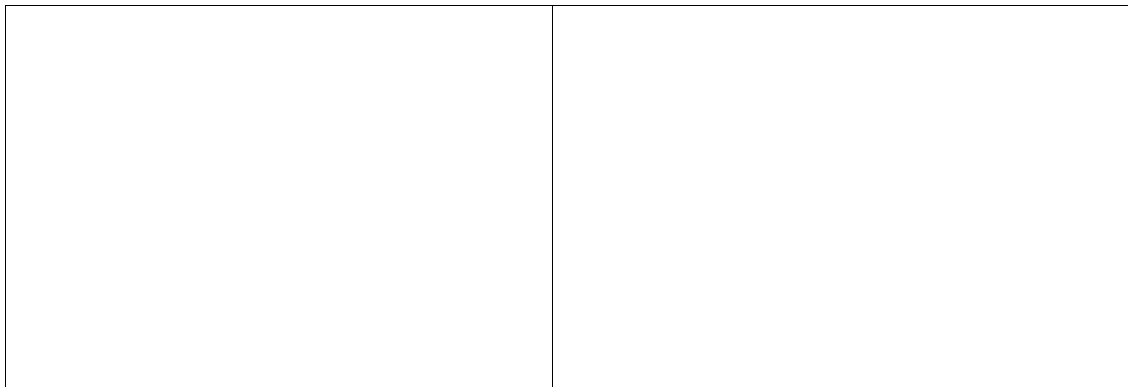


図 - 1.10 三角形の図 (1.8 再掲)

図 - 1.11 内心の作図

(3) 垂心

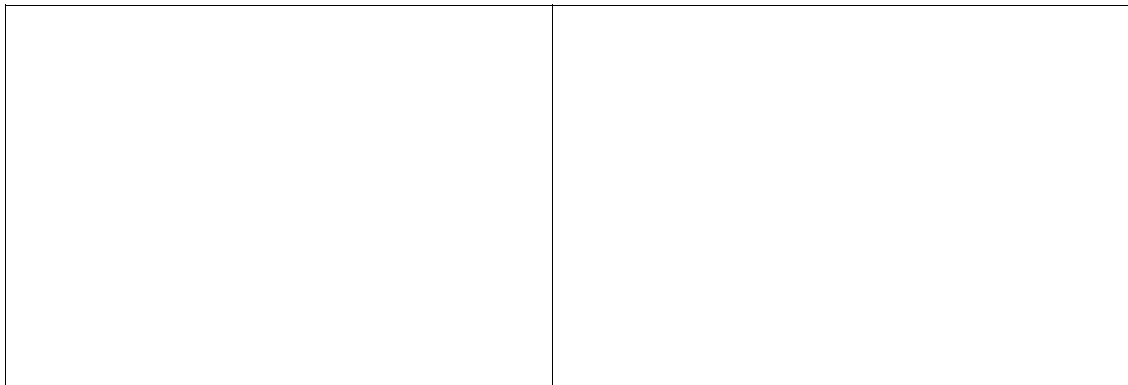


図 - 1.12 三角形の図 (1.8 再掲)

図 - 1.13 垂心の作図

(4) 重心

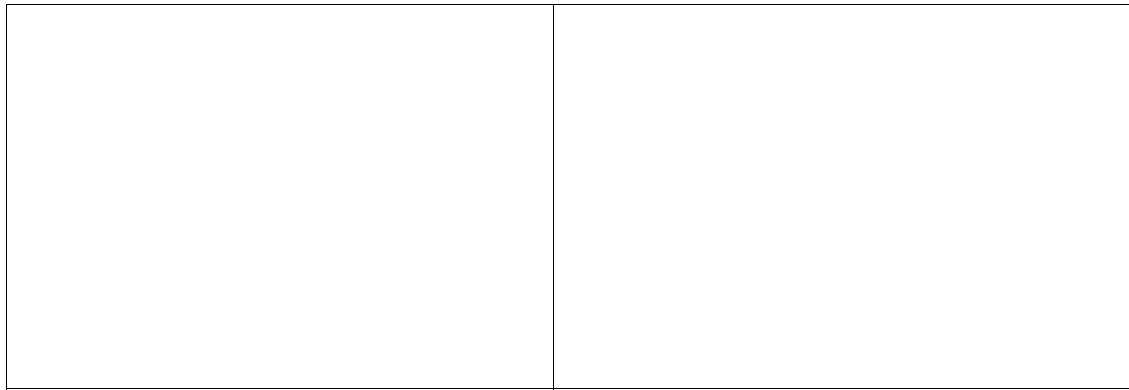


図 - 1.14 三角形の図 (1.8 再掲)

図 - 1.15 重心の作図

(5) 傍心

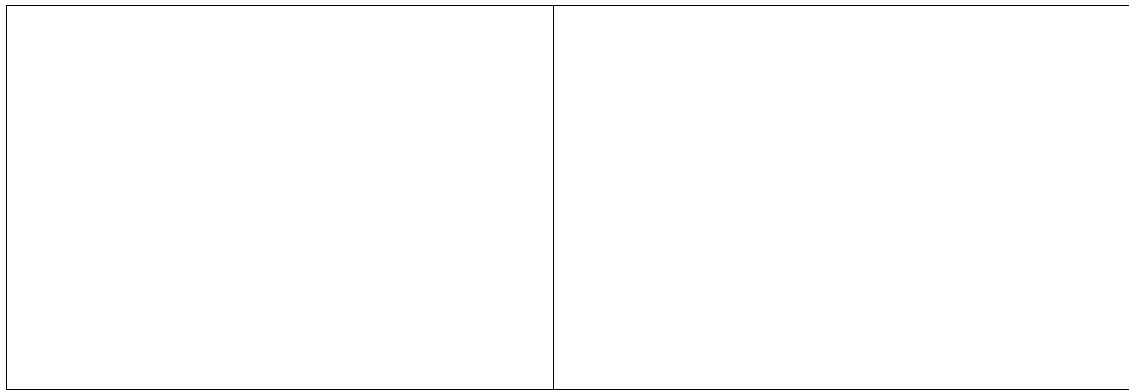


図 - 1.16 三角形の図

図 - 1.17 傍心の作図

1.3 練習問題(2) 四角形に対する追加

上述のような手続きを1回あるいは、何回か使って、左図を右図にせよ。元にするファイルは「四角形」である。これを読み込んで、その図に補助線を書き入れよ。

(1) 4角中点

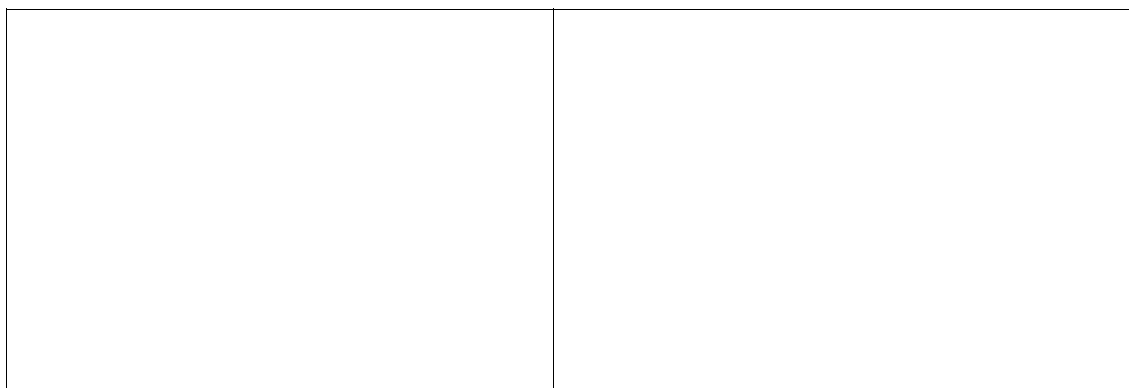


図 - 1.18 四角形の図

図 - 1.19 「4角中点」

(2) 4角角2分

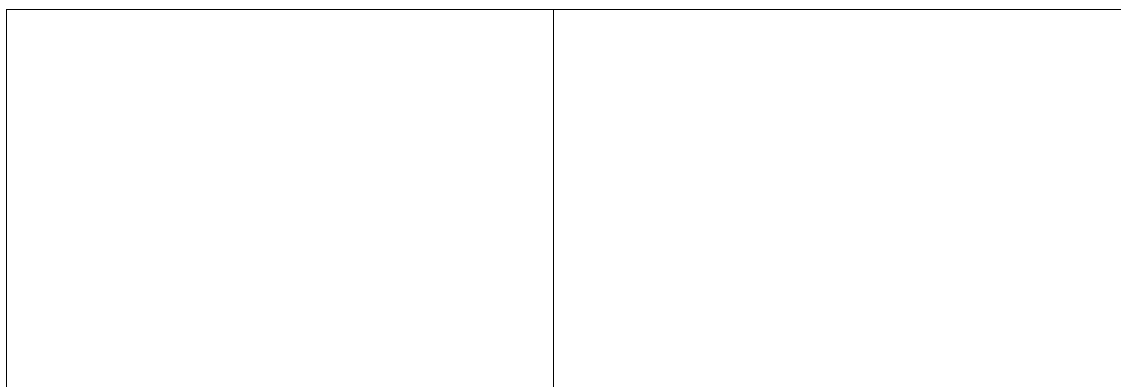


図 - 1.20 四角形の図 (図1.18再掲)

図 - 1.21 「4角角2分」

(3) 4角垂2分

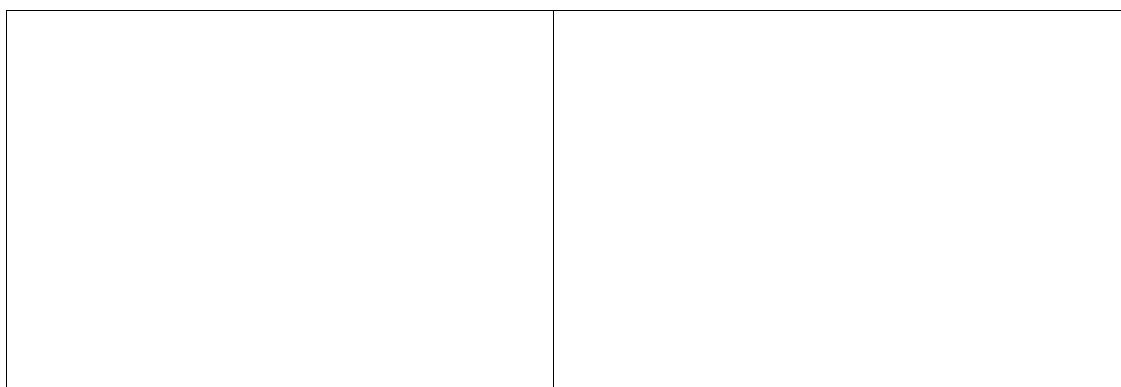


図 - 1.21 四角形の図 (図1.18再掲)

図 - 1.22 「4角垂2分」

2 . 標準的な作図

2.1 白紙の状態から作図を始める時の手順

補助線を追加する場合，元にする図が予めあるが，一般には，白紙の状態から始めなければならない。ここでは，そのように，最初から作図を行う場合について考える。

Geometric Constructor では，あとで点を動かすことを考えているため，他の器具を使った作図の場合とは少し違った図の見方をする。つまり，図形を，

「元にするいくつかの点」と「それを元に構成される他の幾何的対象」に分け，後者のそれぞれについては，「補助線の追加」のときの要領で，構成的に作図をしていく。

例えば，外心の図の場合，

元にする点： 3点 (A , B , C)

と考えるし，

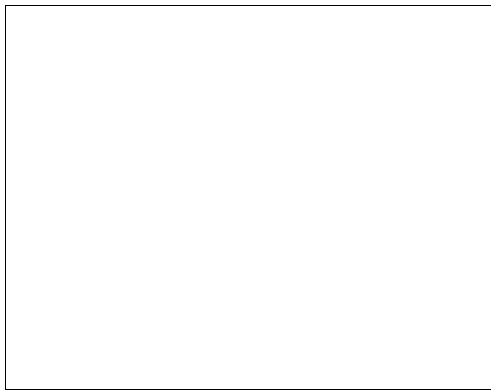


図 - 2.1 元にする3点

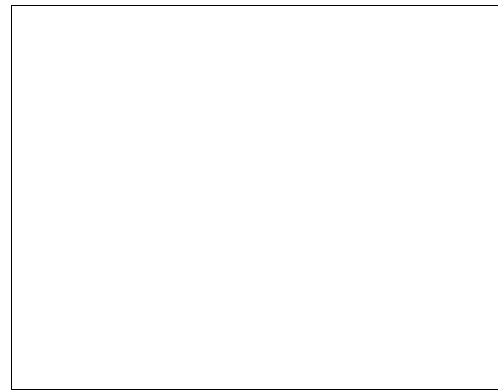


図 - 2.2 外心の図

四角形の各中点を結んで四角形を作る問題の図の場合，

元にする点： 4点 (A , B , C , D)

と考える。

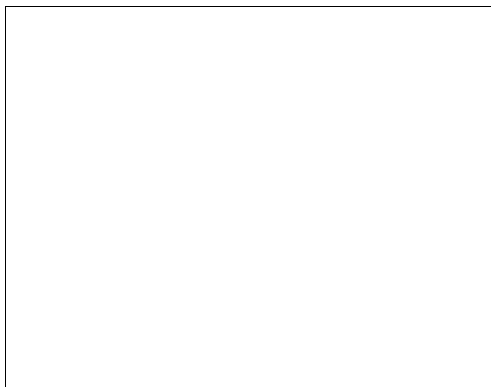


図 - 2.3 元にする4点

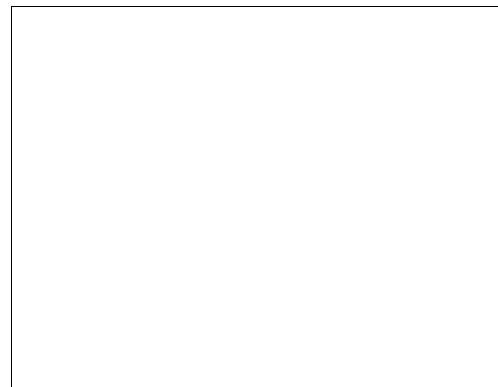


図 - 2.4 4角中点

このように，図形を

「元にするいくつかの点」と「それを元に構成される他の幾何的対象」に分け，まず，元にする点を入力し，あとは順次構成するという手順を経るため，作図に当たって，まず次のことを考える必要がある。

(1) いくつかの点を元にするのか。

(2) その後、どういう手順で作図を進めていけばよいのか。

このことを、外心の図の作図の場合を例にとって、Geometric Constructor での実際の手順を追ってみよう。

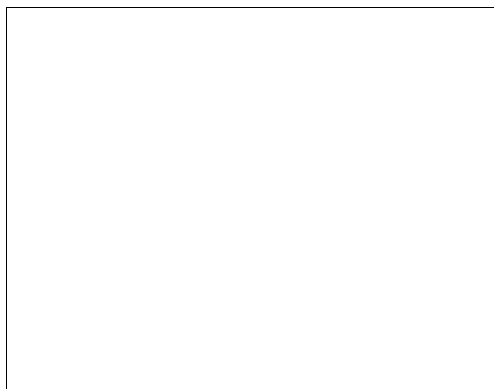


図 - 2.5 完成予定図

まず、元にする点は3つと考え、またその後の手続きを分析しておく。

そして、「新しく作図」を選択し、画面を初期化して、元にする点の数の分だけ、点の位置を入力する。

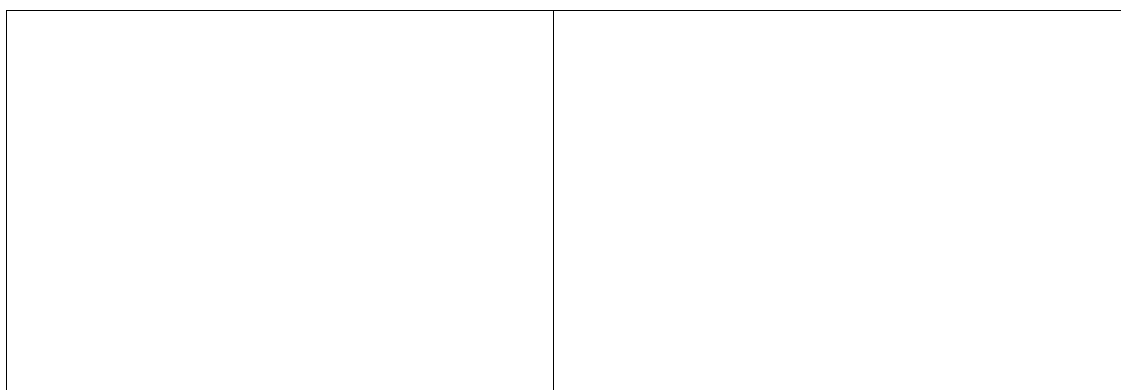


図 - 2.6 「新しく作図」を選択

図 - 2.7 「はい」を選択

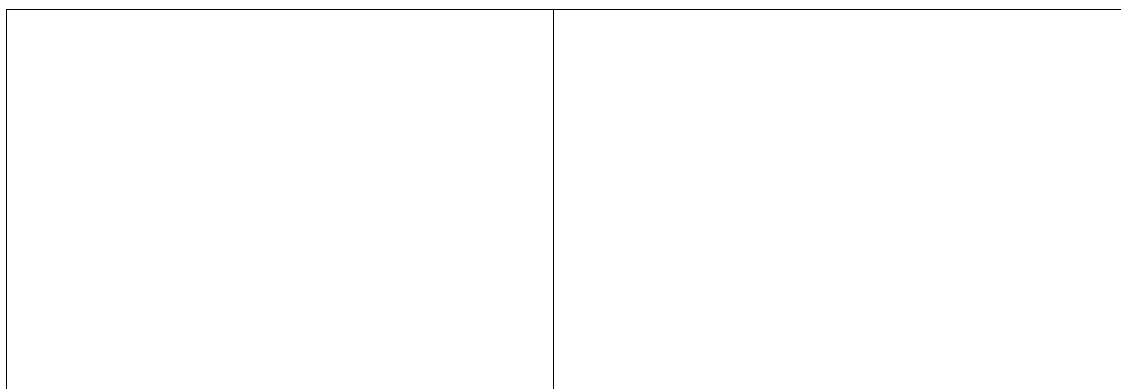


図 - 2.8 元にする点を入力

図 - 2.9 追加する必要がなくなったら Esc

なお、このとき、必要に応じて、次のような機能が使える。

(1) 座標軸の表示 (Aを押せば、座標軸の表示/消去)

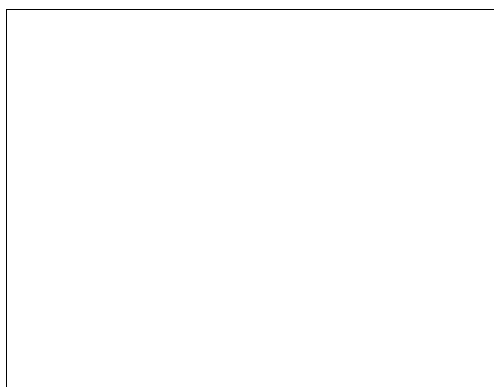


図 - 2.10

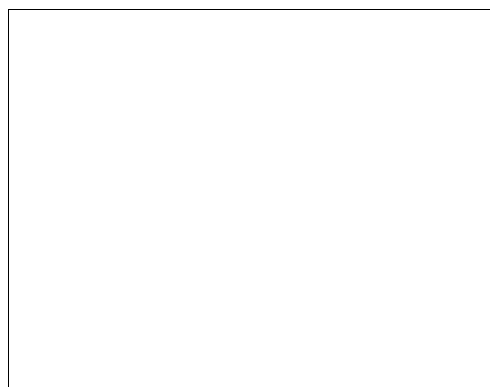


図 - 2.11

(2) 座標の表示 (Cを押せば、座標軸の表示/消去)



図 - 2.12

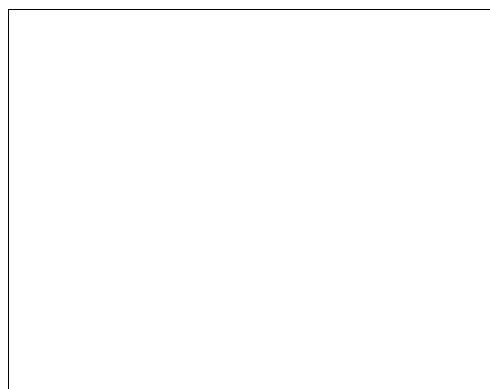


図 - 2.13

(3) Hを押せば、原点に移動

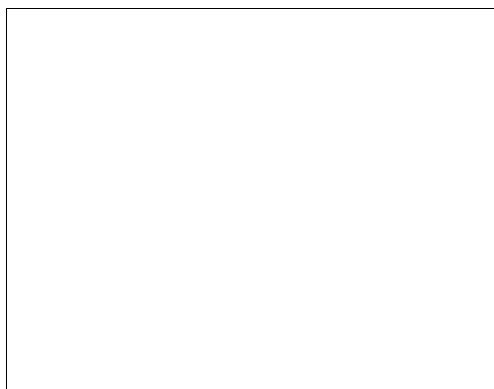


図 - 2.14

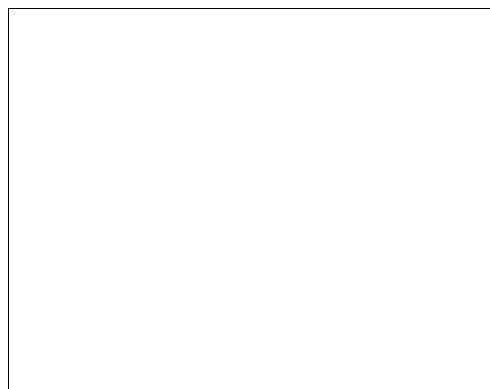


図 - 2.15

(4) 速度の切り換え

- ・ S : 1ドットずつ
- ・ M : 5ドットずつ
- ・ F : 10ドットずつ
- ・ 1 ~ 5 : その長さ分 (ドットではない)

・ 0 : オプション (ドット数で, 自由に設定可能)

次に三角形を作図するため, 「線分, 多角形」を選択し, 「多角形」を選択。そして, 「3」あるいは「0」を入力。 を使って3点を指定すれば, 三角形が作図される。

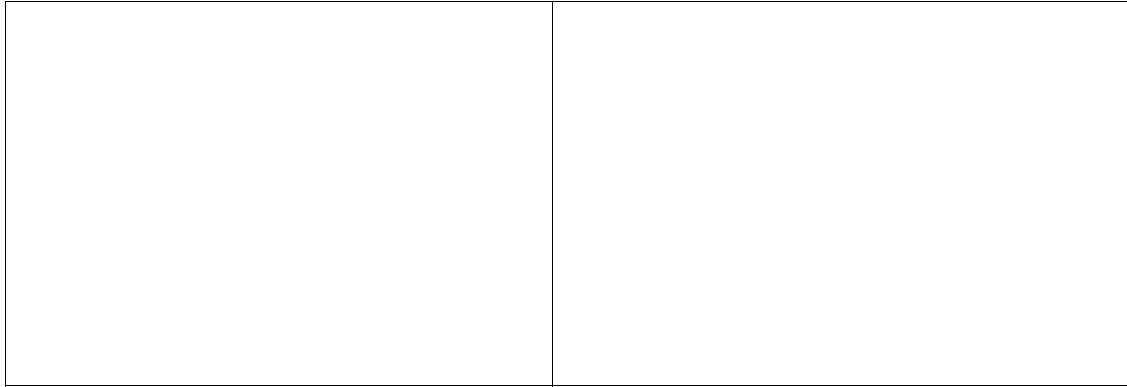


図 - 2.16 「線分, 多角形」を選択

図 - 2.17 「多角形」を選択

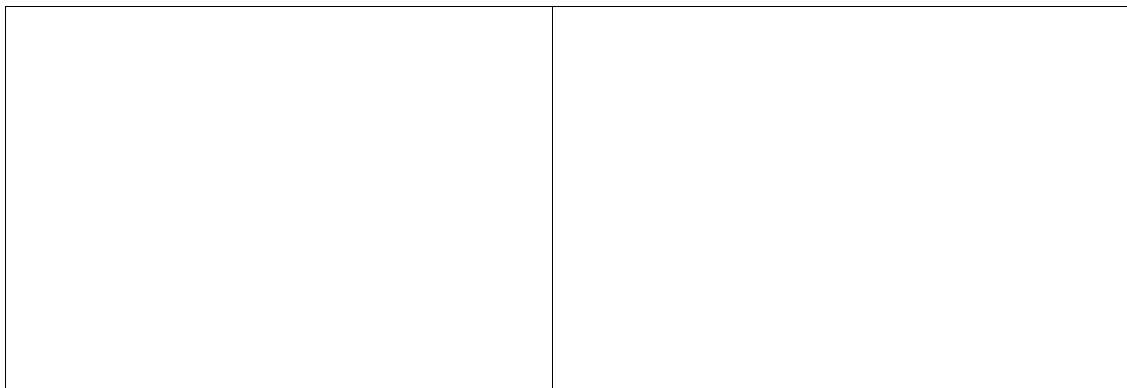


図 - 2.18 「3」角形を入力 (0でも可)

図 - 2.19 を使って3点を指定

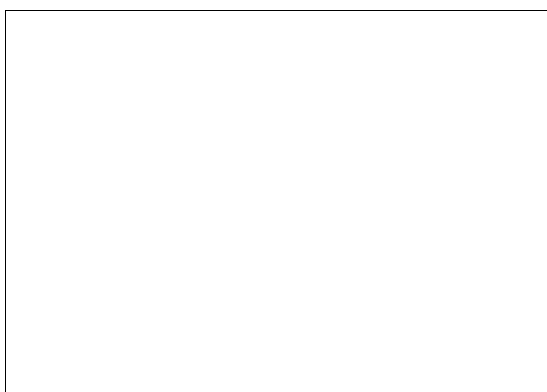


図 - 2.20 三角形の作図終了

次に垂直二等分線を作図したい。これは直線なので、次のような流れの入力になる。

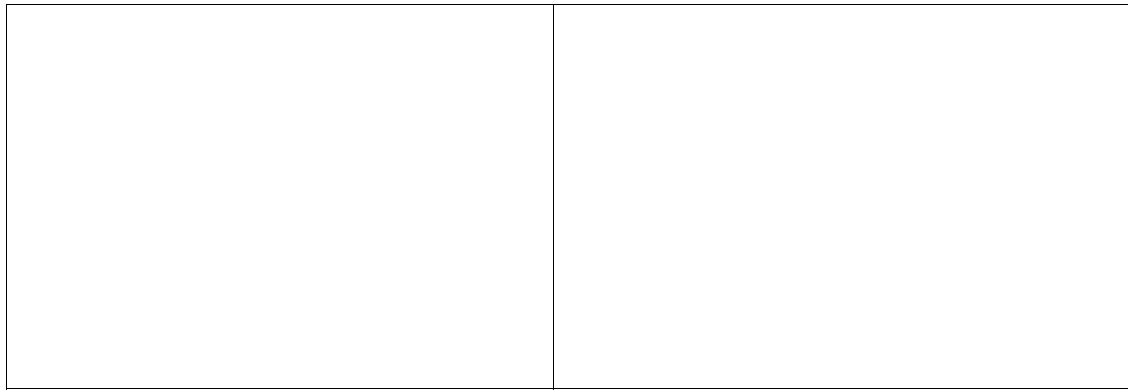


図 - 2.21 「直線」を選択

図 - 2.22 「垂直二等分線」を選択

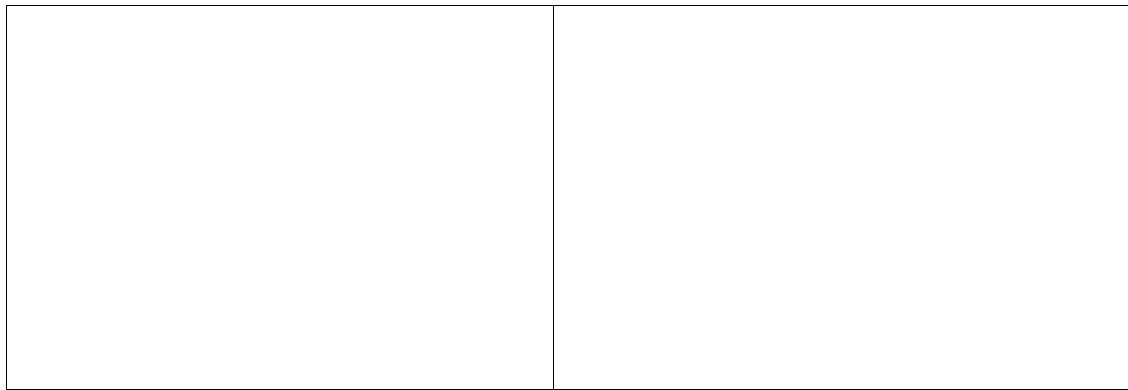


図 - 2.23 を使って線分を指定

図 - 2.24 垂直二等分線の作図終了

これを3回繰り返すと、完成予定図が得られる。なお、外心そのものが必要ならば、更に、交点を「点」として意識化しなければならないので、「点」の中の「様々な交点」の中の「2直線の交点」を指定し、作図する必要がある。



図 - 2.25 「点」を選択

図 - 2.26 「2直線の交点」を選択

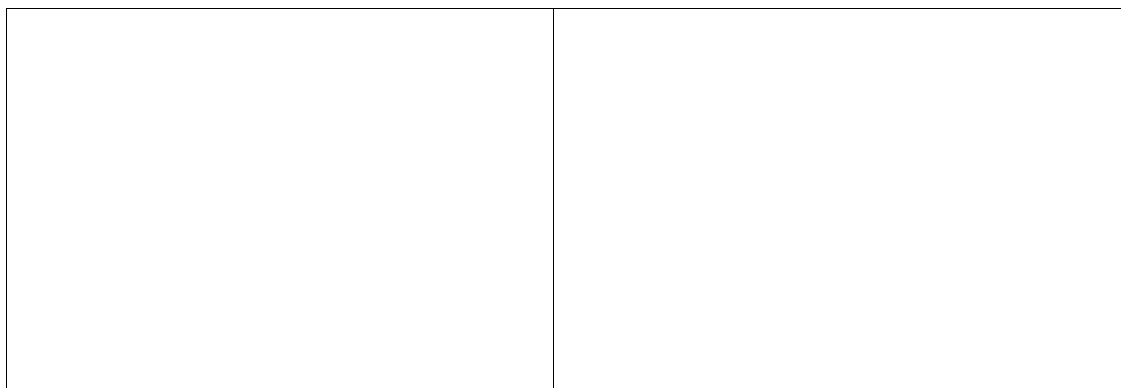


図 - 2.27 2つの直線を選択

図 - 2.28 出来上がり

2.2 分析について

作図の途中で、自分がこれまでにどのような作図をきたのかわからなくなったり、また他の人の作図の仕方について調べたいときには、編集メニューの中の「分析」を使う手がある。「分析」を選択したときに表示されるファイルの中で、「作図データ」と「幾何対象」という二つのファイルが分析結果であるが、普通は、「作図データ」の方を使う。次図のような形で、作図の手順を表示する。

なお、幾何対象の方は、それぞれの変数、点、線、円に関する情報を表示するが、こちらを使うのは、主にソフトのバグ（不具合）発見のためであり、普段は使わない。

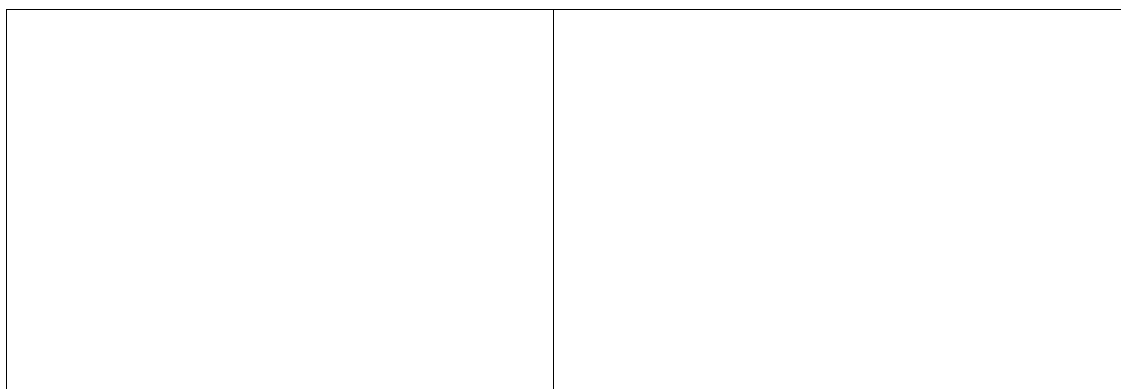


図 - 2.29 「編集」を選択

図 - 2.30 「データ表示」を選択

「データ表示」の結果は、ファイルとして出力したものを、簡易エディタで表示している。印刷したければ、簡易エディタ内からも印刷できるし、DOS 上で、PRINT コマンドを使ってもいい。

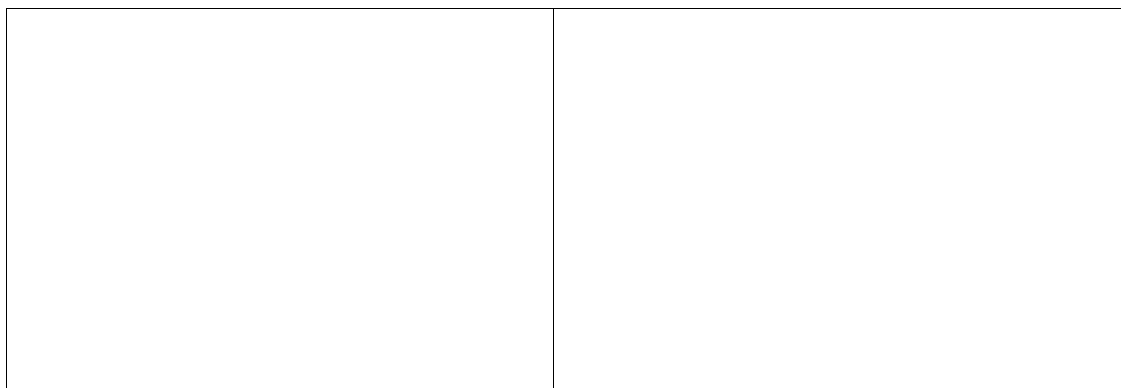


図 - 2.31 「作図データ」の内容

図 - 2.32 「幾何対象」の内容

2.3 練習問題(3) 作図

以下の図と同じような図を作図せよ。なお，どうしていいかわからないときには，「例」の「全ファイル」の中から選択し，分析すること。

(1)

(2)

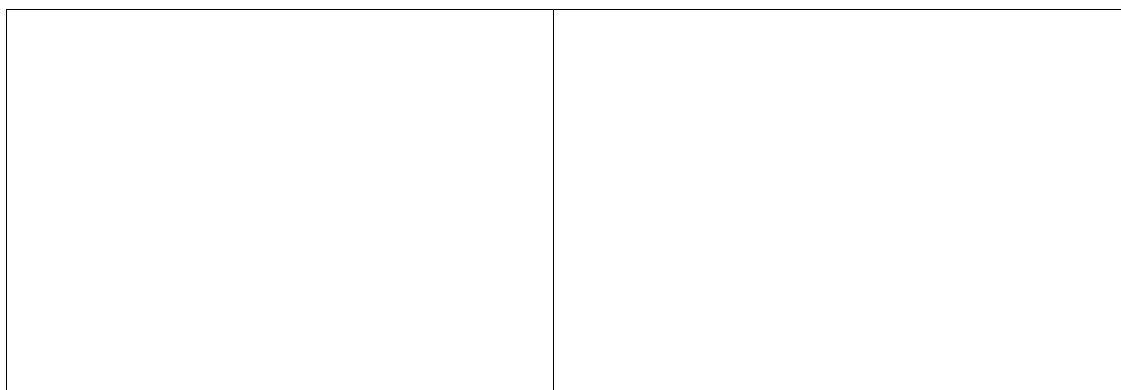


図 - 2.33 平行四辺形

図 - 2.34 垂足三角形

(3)

(4)

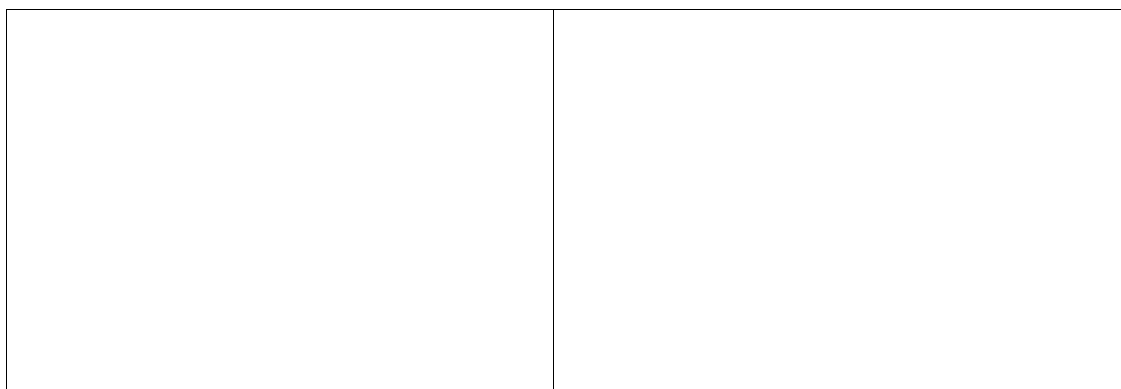


図 - 2.35 辺の中点を結んで三角形

図 - 2.36 辺を2:1 に内分して三角形

(5)

(6)

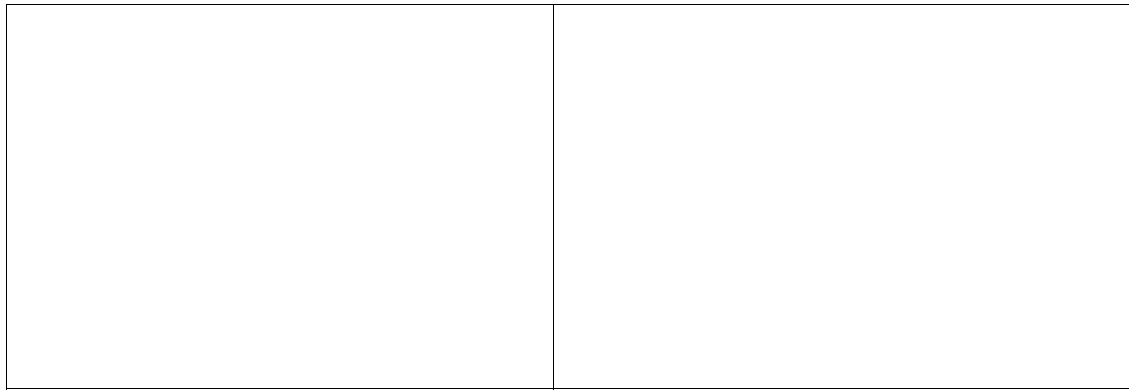


図 - 2.37 (4) の手続きを 2 回繰り返す 図 - 2.38 角の 2 等分から内心

(7)

(8)

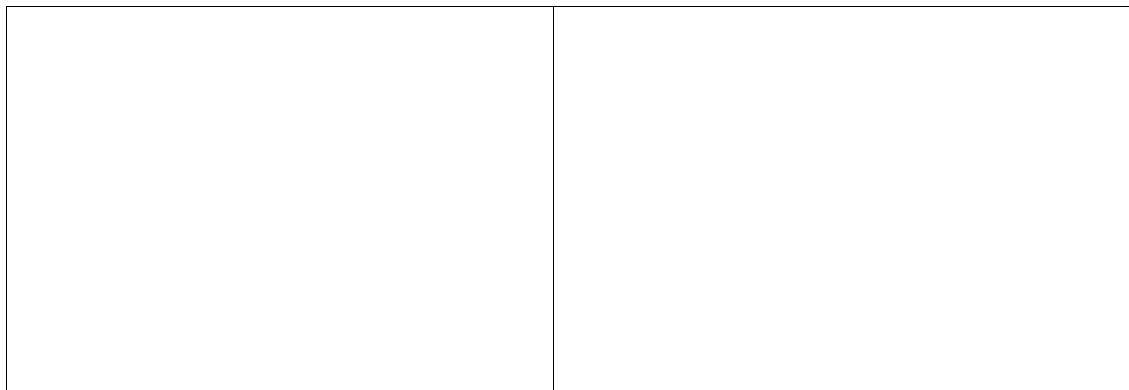


図 - 2.39 角の 3 等分線

図 - 2.40 モーレーの定理

3 . 点の動きの制限

3.1 「点の動きの制限」とは

標準的な作図によって作図した図形は，変形することができる。変形では，キーボードあるいはマウスを使えるが，それぞれ，次のような特徴がある。

キーボード：上下左右の方向に一定は幅ずつ移動する。

上下左右以外の方向に「そのまま」進むことはできない。

コンピュータのCPUの速度がそのまま反映される。

マウス：マウスを動かしたところに移動するので，「大体」思ったところに移動できる。

その移動は「アバウト」であって，精確なものではない。また，目をかなり使う。

コンピュータのCPUが遅い場合には，「たまに描く」ようになるだけでその速度は図形の移動速度には関係ない。

このような動かし方は，いわば「フリー」な動かし方だが，これら不便なときもある。

例えば，ABCの面積が不変であるように変形したい場合，AをBCに平行に動かす必要がある。

(1)BCが水平あるいは鉛直の場合

キーボードを使うと，「上下」あるいは「左右」が目的の動かし方となるので，非常に簡単である。しかし，マウスの場合，ピッタリ上下，左右に動かすことはほとんど不可能に近い。「大体」の様子を知りたいときはこれでいいが，「ぴったり」させたい場合には，工夫がいる。

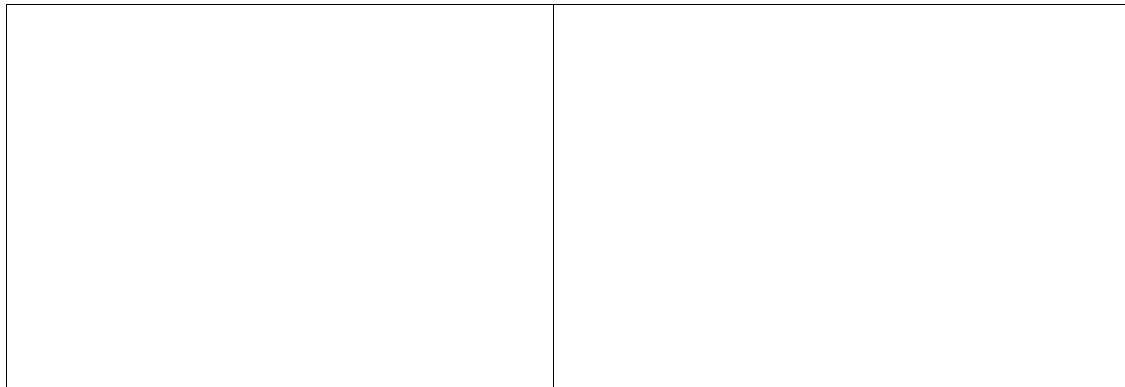


図 - 3.1 キーボードによる移動

図 - 3.2 マウスによる移動

(2)BCが斜めの場合

キーボードを使うと，直接斜めの動きができないので，いくつか右に動いて上に上がるというような不自然な動きしかできない。マウスの場合は上下，左右のときとほぼ同様の操作感である。やはり「大体」の様子を知りたいときはこれでいいが，「ぴったり」させたい場合には，工夫がいる。

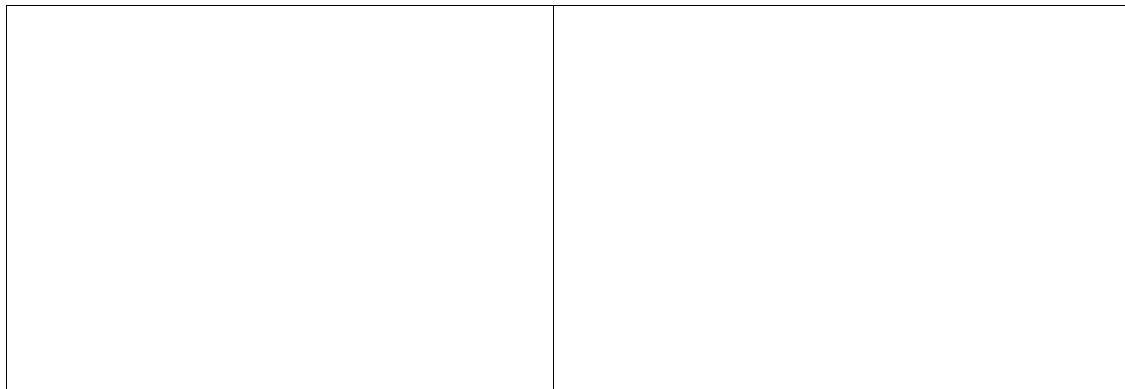


図 - 3.3 キーボードによる移動

図 - 3.4 マウスによる移動

(3) 円上を動かしたい場合

たとえば円周角の定理のように，円上を動かしたい場合は，キーボードでの使いにくさはとても顕著になる。マウスの方がましではあるが，これも角度などを計測しているときには，誤差がとても大きくなってしまう。

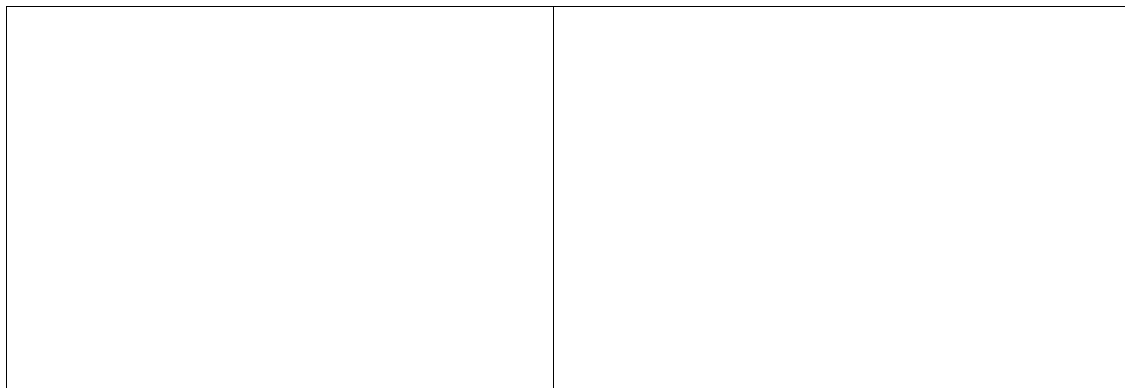


図 - 3.5 キーボードによる移動

図 - 3.6 マウスによる移動

授業での利用を考えた場合，「推測」のために使う場合は，このような使い方ではほぼ妥当であると言えるが，「検証」のために使う場合は，不適切である。

そこで，これらの問題を解消するために，Geometric Constructor では，指定した直線や円などの上のみを動かすように，点の動きを制限することができる。たとえば，上記の3例の場合，次のような動きとなる。

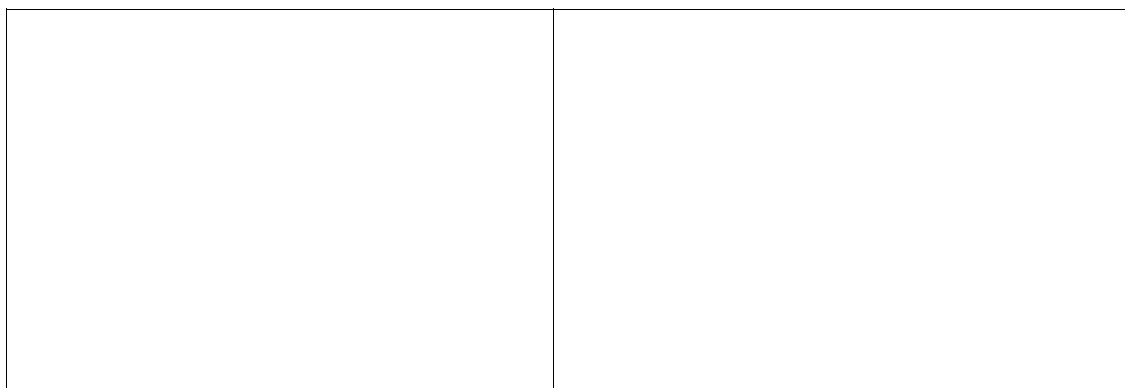


図 - 3.7 水平なBC

図 - 3.8 斜めのBC

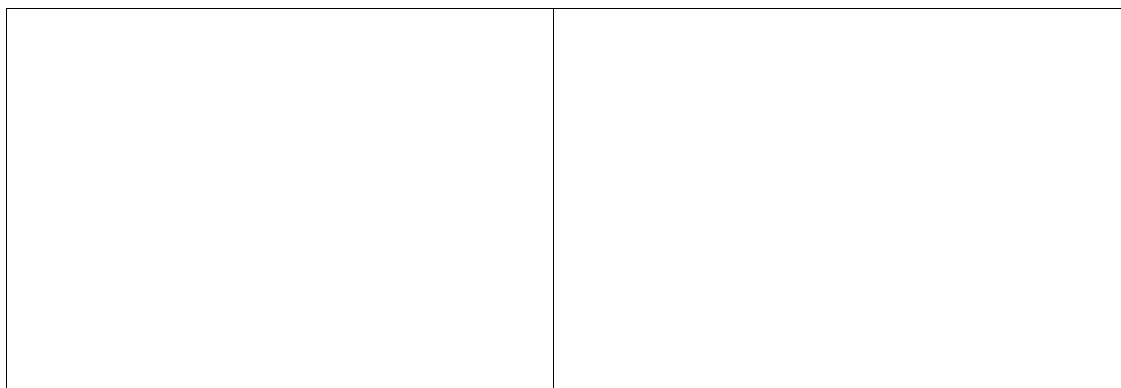


図 - 3.9 円上 (キーボード)

図 - 3.10円上 (マウス)

なお、それぞれの線・円上に動きを制限した場合は、

- ・キーボードの場合は、その上の等速度運動
- ・マウスの場合は、クリックした位置に一番近い場所への移動

というような動きをする。(上図参照)

3.2 基本的な考え方と作図方法 - 事前に準備 -

このような作図をするためには、基本的には、点Aを決める前に、それがのる直線あるいは円を作図し、その上の点として、点Aを指定しなければならない。

そのための基本的な作図の流れは、次のようなものになる。

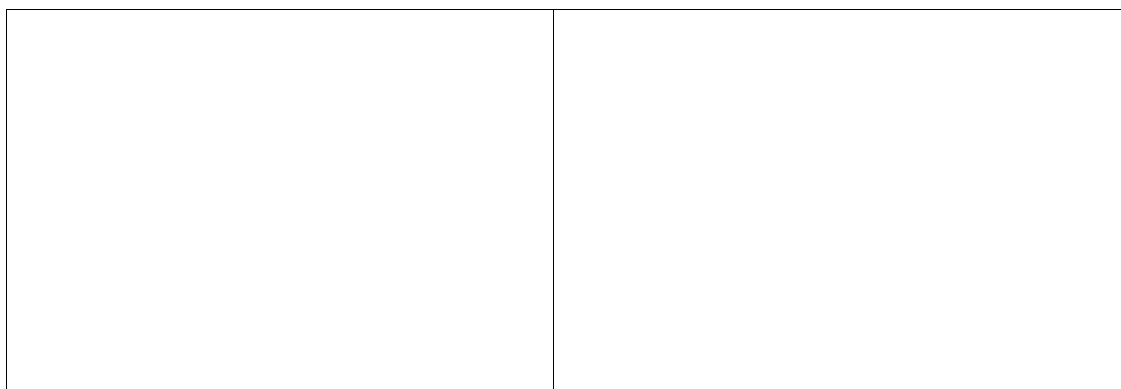


図 - 3.11完成予定図

図 - 3.12点Aの前に、それがのる直線が必要

ここで、直線はBCに平行に引きたいが、BCのみでは決まらない。

何らかの点Xを予め決め、Xを通してBCに平行に直線を決める必要がある。

そのため、次のような手順となる。

- ・ B , C を取る。
 - ・ B C を結ぶ。
 - ・ B C の外に点 X を取る。
 - ・ X を通り B C に平行な直線を引く。
 - ・ 「点」 「新しい点 (図形上)」 「直線」 . . .
- による A の決定
- ・ A B , A C を結ぶ。

順序はどちらが先でもいい。

(ここでの点の名前は、完成予定図での名前等を元にしてている。作図を最初に行うときに

は、自動的に A , B , C が付けられるので、上記の順番にはならない。これらの不具合は、必要があれば、後で編集機能で修正すればいい。)

これらの流れを追ってみると、次のようになる。

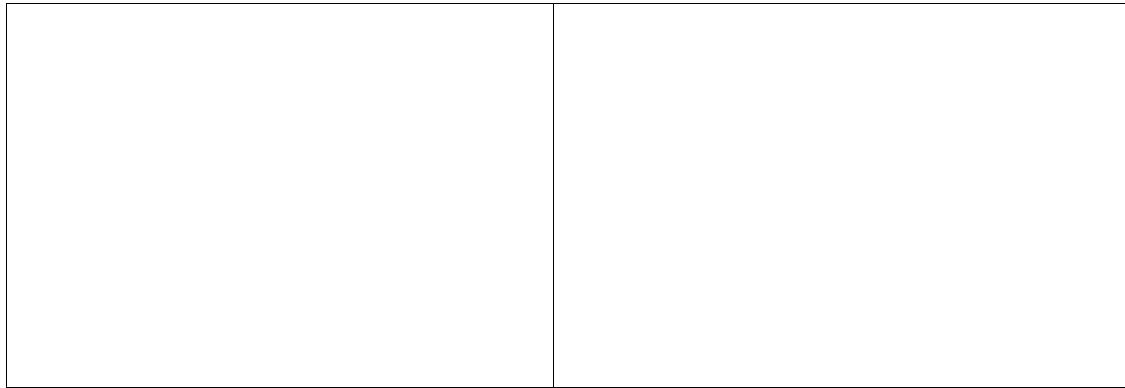


図 - 3.13 2点を取る。

図 - 3.14 2点を結ぶ

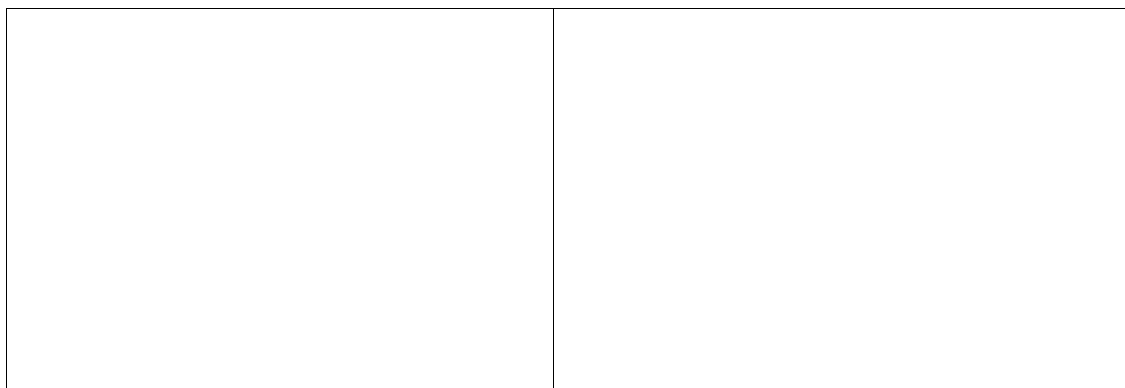


図 - 3.15 外に点を取る。

図 - 3.16 平行線を引く。

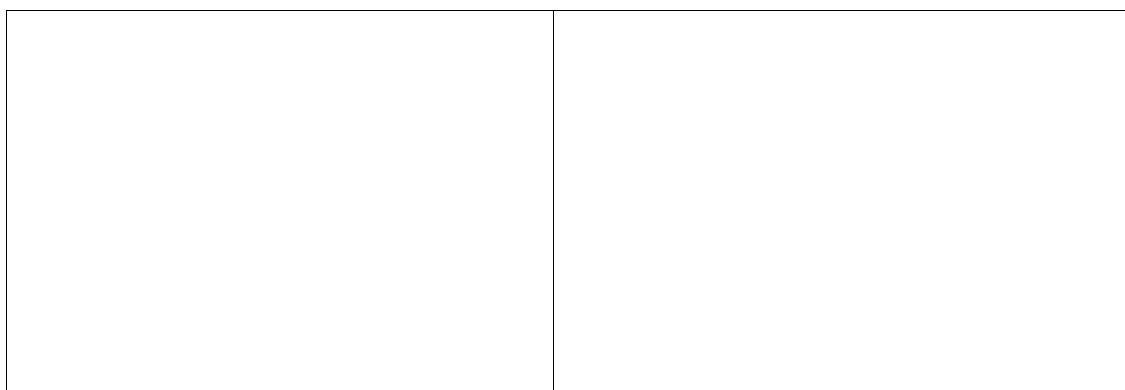


図 - 3.17 平行線上に点を取る。

図 - 3.18 三角形を作る。

3.3 「後から」修正するための方法と原則

しかし、作図をするときは、多くの場合、何らかの問題について取り組んでいるときであり、初めからそのような動きを予定しているときばかりとは限らない。その度に、最初から作図をしなおすのでは、面倒である。

多くの場合、「後から」修正することも可能である。例えば、上記の例では、次のよう

な手順を取ることができる。

- ・ A B C を作る。
- ・ 適当に点 D を取る。
- ・ D を通り B C に平行な直線を作る。
- ・ 「編集」の「点の束縛」で点 A を選択し、「直線」を選び、上記の直線を選択する。
その上で位置を決定する。

最後のメニュー操作の部分の動きを追うと、次のようになる。

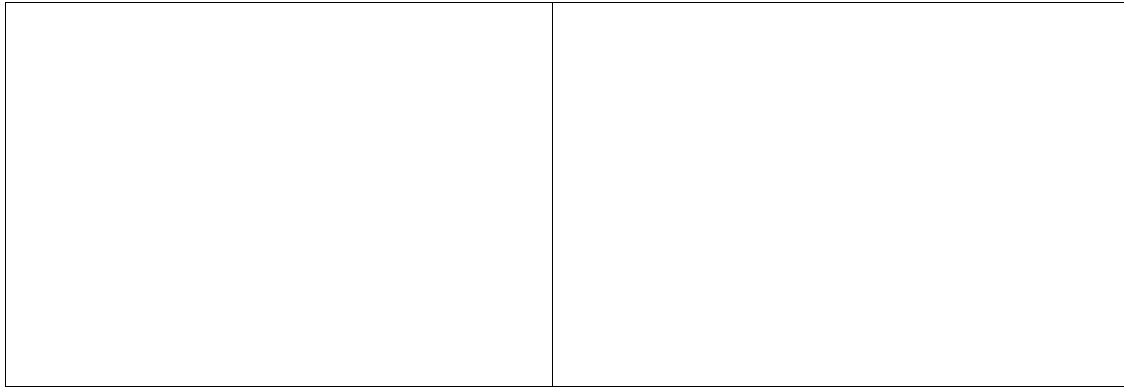


図 - 3.19 編集メニュー選択

図 - 3.20 「点の束縛条件」

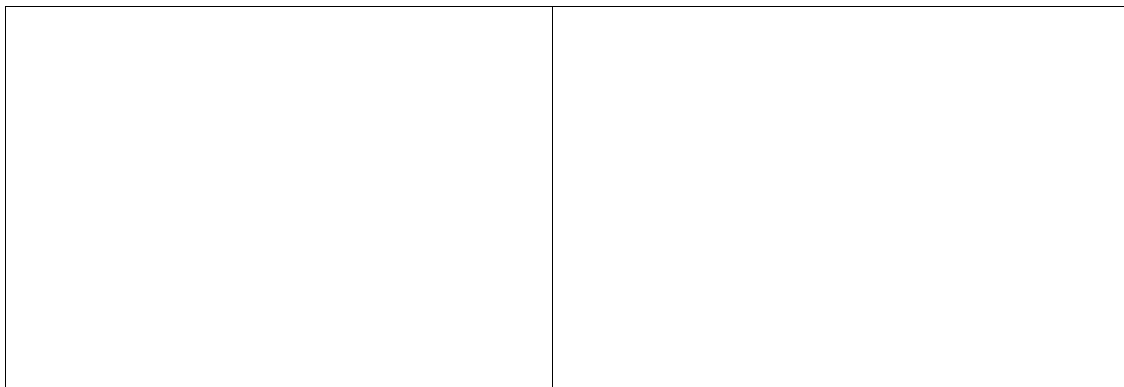


図 - 3.21 点を指定

図 - 3.22 束縛条件を選択

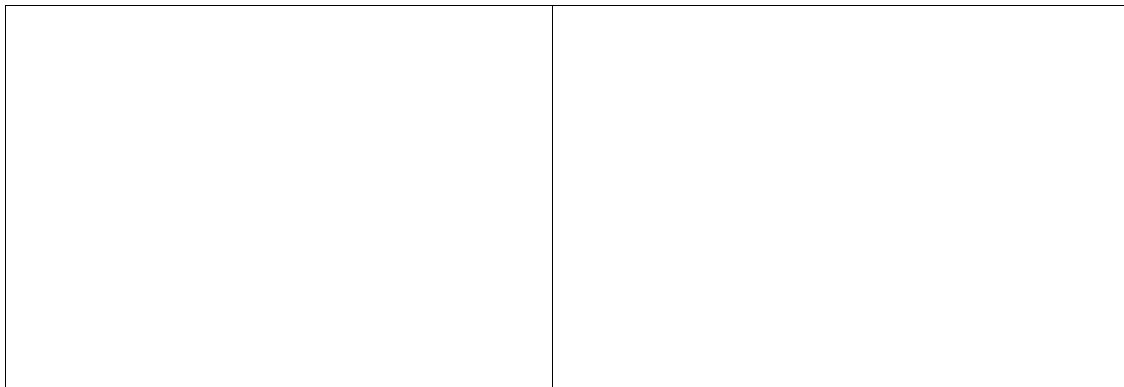


図 - 3.23 線 (円) などを選択

図 - 3.24 次へ (不要ならばEsc)

ところで、この操作をするときに、「どの対象を選択してもいいわけではない」。

たとえば、A B C の点 A を「A B」の上に制限してみたとして。

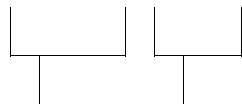
元の A の位置 A B が決まる その上の A の位置 A B が変更

というような、堂々巡りとなってしまふ。基本的に、このような選択をすると、図が異常になってしまふ。(事前にそれを感知して、選ばないようにできるといいのだが、このソフトではそこまで面倒を見ていない。)

どのようにして、それを事前に判断したらよいだらうか。基本的な原則は次のように考えたい。

まず、それぞれの点・直線・円・変数の間の「依存関係」を考えてみる。たとえば、三角形ABCの場合には、

点 A B C (それぞれが独立変数)



A B B C などとなっている。(CAは省略)

たとえば、外心を作図する場合は、さらにABを元にして、垂直二等分線ができ、それらから、外心ができる。たとえば、外心はA, B, Cの関数である。

逆に、たとえば、線分BCは、Aとは無関係である。このように、従属関係において、無関係な対象の上に制限することは問題なく可能である。

そうでない場合、つまり自分自身によって影響を受けるような対象の上に自分を置くことは、ほとんどの場合、トラブルが生じると思っている。

3.4 練習問題

次のような目的の図を作れ。点をどのように束縛するといいか。

(1) 等積変形

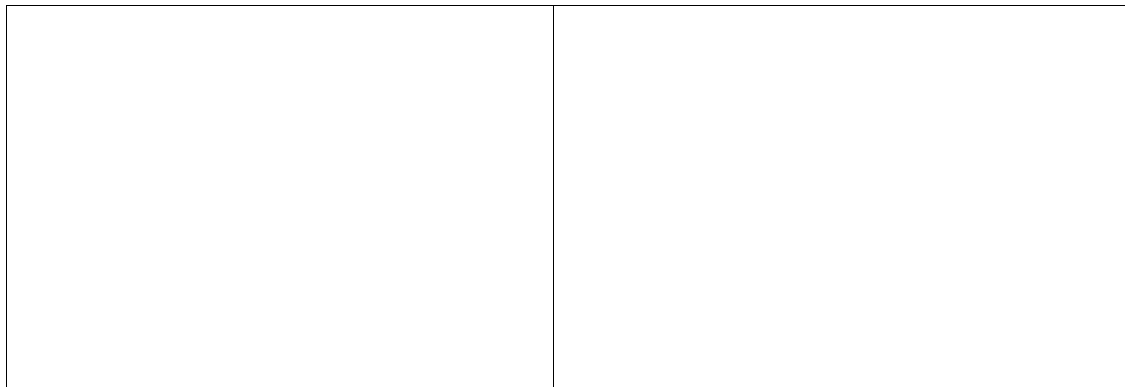


図 - 3.25A

図 - 3.25B

(2) 円周角

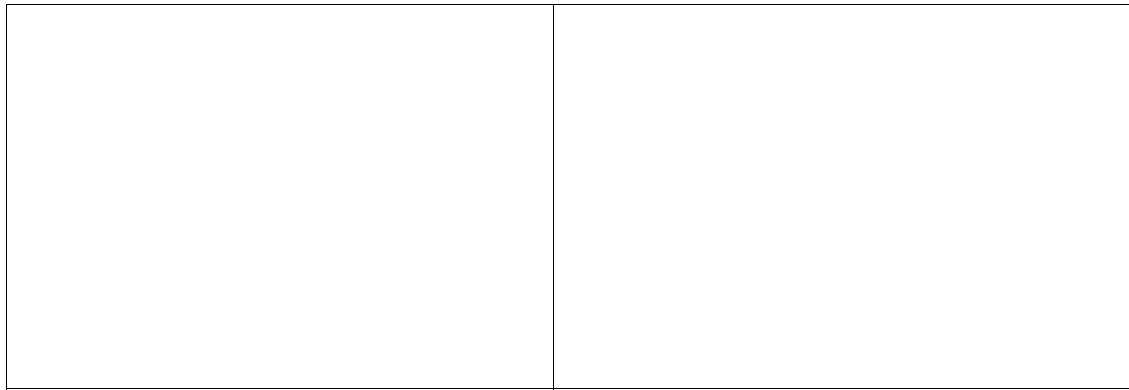


図 - 3.26A

図 - 3.26B

(3) 線分AB上に二つの正方形

(4) 三平方の定理

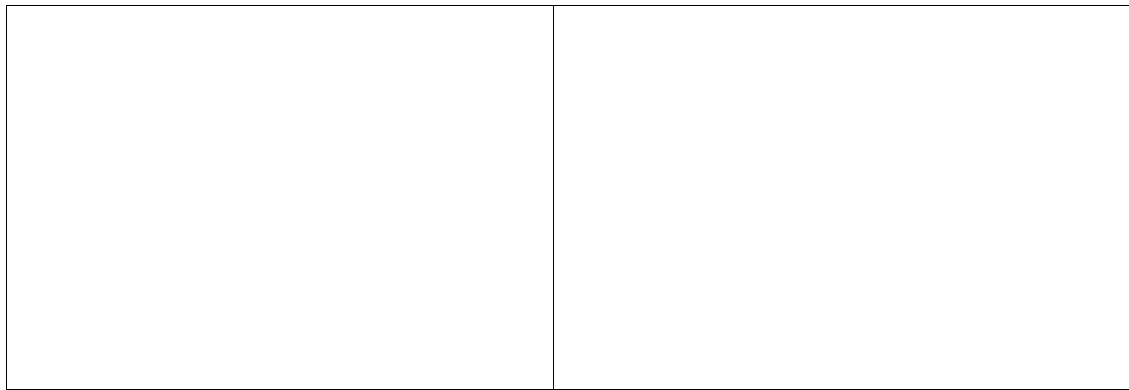


図 - 3.27

図 - 3.28

(5) 円上の点と円外の点との中点の軌跡

(6) 2円の上の点の中点の軌跡

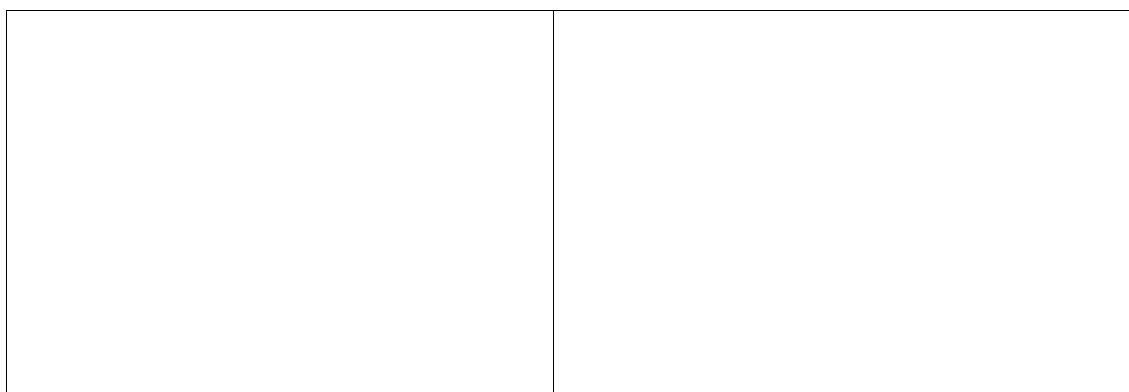


図 - 3.29

図 - 3.30

(7) 三角形の2辺に接する円

(8) 2点を通る円

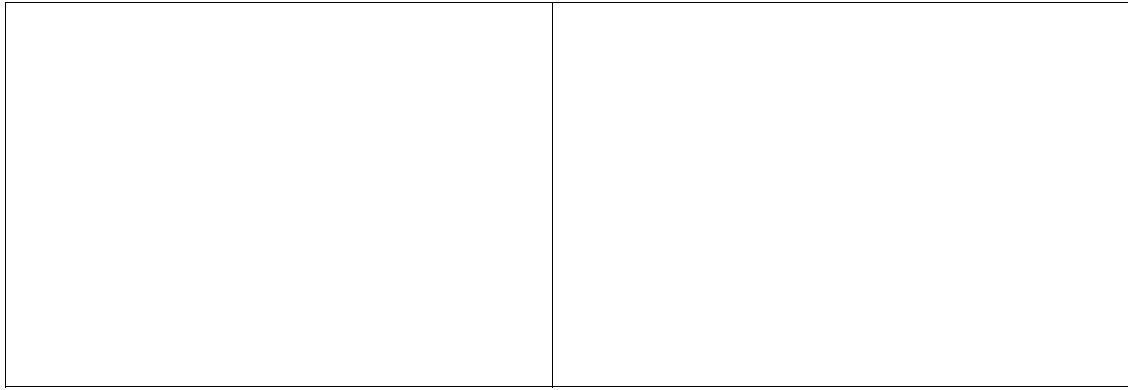


図 - 3.31

図 - 3.32

4 . 基本的な編集機能

4.1 色の設定

色の設定を行う動機は二つある。一つは「目立つ」ようにしたいという場合であり、一つは、「隠したい」という場合である。

(1) 目立たせたい

例えば、次図を考えてみよう。「四角形ABCD」と「四角形EFGH」というように、言葉で言っても分かりにくい場合が少なくない。実際、画面を観察し、「文字を読め」というのは、酷なことも多いからである。

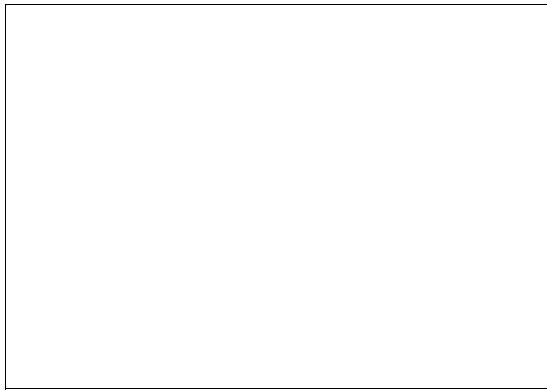


図 - 4.1

そういうときには、「外の四角形」と「中の四角形」という言葉で言うこともできるが、もう一つの手としては、「白い四角形と赤い四角形」という表現であろう。

このように、注目したいものを観察しやすくするためには、色を変えるというのは、有力な方法である。

(2) 隠したい

特に、点の動きを制限するために、何らかの直線や円を作図してあった場合を考えてみよう。このときの直線や円は、いわば「舞台裏」である。あるいは、「足場」である。これがあると目障りだ。しかし、これなしには、図そのものができない。何とか、「隠すことはできないだろうか」と思うことが少なくない。そういうときに、「描かない」という選択肢を選択しよう。

(3) 基本的な手続き

手続きは難しくない。

- ・編集メニューを選択
- ・「色」を選択
- ・「種類」を選択
- ・「対象を選択」
- ・「色」を選択
- ・次へ (あるいは、終了ならば、Esc)

という流れを踏襲すればいい。(図は省略しよう。)

4.2 名前の設定

点の動きを制限した場合などにやはり気になるのが「名前」である。たとえば、「X」などという名前を付けたい場合がある。これも、上記とほぼ同様にできる。つまり、

- ・編集メニューを選択
- ・「名前」を選択
- ・「種類」を選択
- ・「対象を選択」
- ・名前を修正
- ・次へ（あるいは、終了ならば、Esc)

4.3 点の属性の設定

やはり同様に、目立たせたい／区別したいというようなときに、点に関しては、その表示の仕方を変えることができる。

編集メニューの「点の表示」を選択し、実際に点を選択すると、何種類かの表示方法（属性）がメニューで表示される。

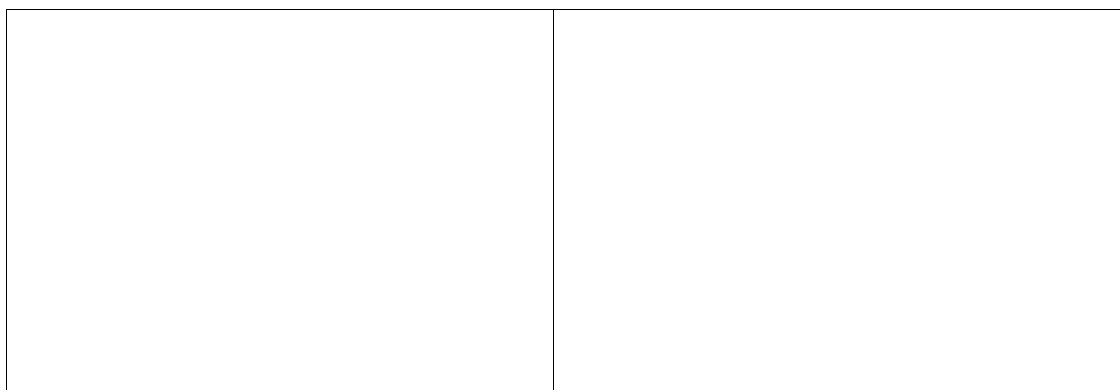


図 - 4.2 「点の表示」

図 - 4.3 「属性」に関するメニュー

は「変形時に動かしている点」を表示するためのものなので、あまり使わない方がいいが、その他に、いくつかのものがあるので、実際に見てほしい。

なお、その中に、「塗る」という機能がある。これは、その点が入っている領域を PAINT 命令で文字通り「塗る」という機能である。

例えば、三角形ABC の内部を塗りたいというような場合には、

- ・ ABC の重心（必ず内部にある点の候補として）を取る
- ・重心の属性を「塗る」に変える

という手続きを取ると、次図のようになる。

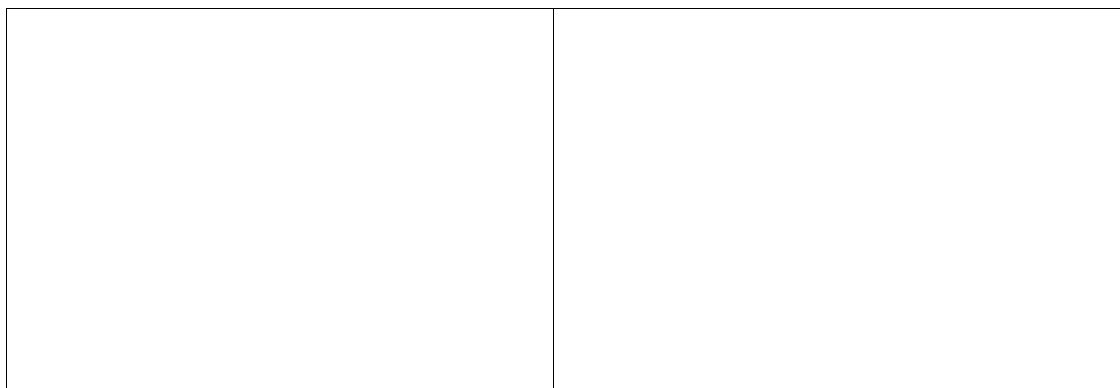


図 - 4.4

図 - 4.5

しかし、この機能には、それほど期待してはいけない。というのも、線分で囲まれている領域を塗るのであって、三角形というものを認識しているわけではないからだ。たとえば、次図のような場合、左図のような配置ならば、目的通りになるが、右図のようになると、目的とは違う塗りかたとなってしまう。

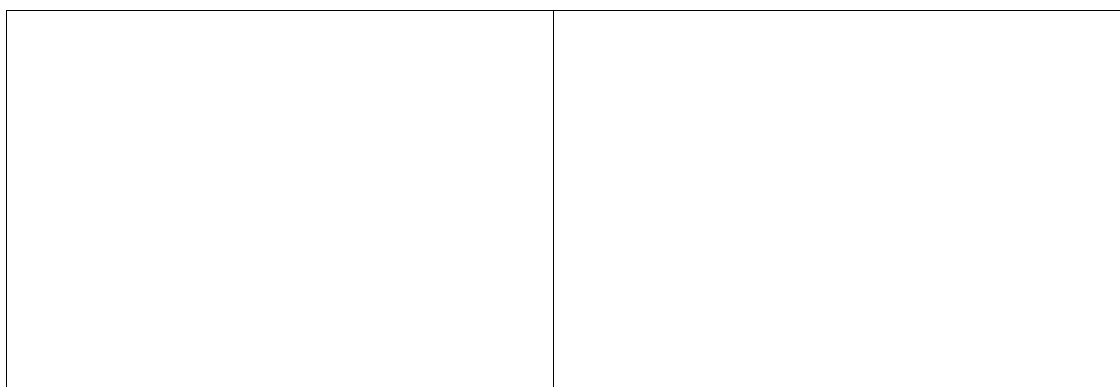


図 - 4.6

図 - 4.7

4.4 直線系の属性の設定

Geometric Constructor 内部では、線分・直線・反直線というのは同一のデータ構造を持っていて、その属性のみを変化させている。そのため、作図した後で変更することが可能である。

たとえば、「4角角2分」の図を変形させ、右図のようになった場合、Geometric Constructor での角の2等分線は劣角の方の2等分線を引くので、引きたい方でない側に反転してしまう。このとき、改めて「直線」を追加するなどの手を講じるのは面倒だったりする。こういうときは、直線の属性を「反対側の半直線」に変えたり、あるいは「直線」に変えたりする方がやりやすいことがある。そういうときのために使うと便利な機能である。

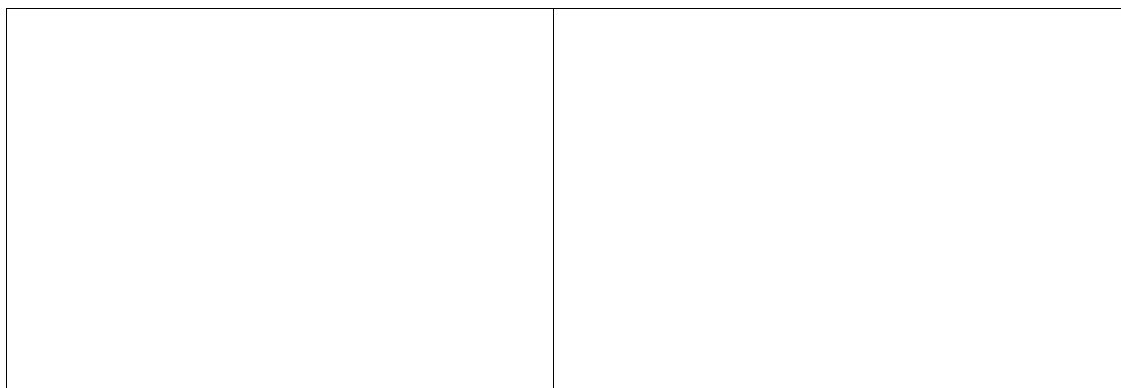


図 - 4.8 元の図

図 - 4.9 変形した様子

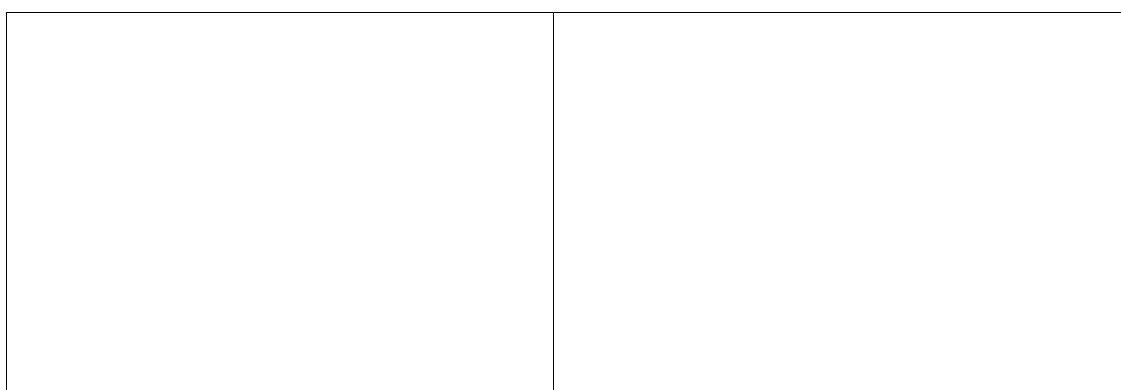


図 - 4.10 「反対側の半直線」へ

図 - 4.11 「直線」へ

5 . 軌跡による図形の描画

5.1 数学での「軌跡」とGeometric Constructor での軌跡

もともと、数学的には、「軌跡」というのは、次の二つの側面を持っている。

- (1) 点の動きの跡
- (2) 条件を満たす点の集合

そして、多くの場合、これらが直線や曲線、いわば1次元多様体をなす場合「のみ」を軌跡と表現するのが伝統的である。

しかし、たとえばGeometric Constructor で扱う場合には、「結果」によって、それを「軌跡」として見なすかどうかを判定することはできない。そのため、機能としては、上記の二つに対応して、

- (GC1) 幾何的対象の動きの跡としての「軌跡」
- (GC2) 条件を満たす点の集合としての「軌跡」

という二つを持っている。

この中の前者について、この章で扱う。後者については 章で扱う。

5.2 最も簡単な使い方

Geometric Constructor における軌跡の利用において、まず基本的なのは、

- ・ 軌跡の設定を行う
- ・ F 9 キーによる ON / OFF

の二つの仕事が必要だということである。前者の軌跡の設定は、ファイルの中に保存されるため、一番やさしい軌跡の使い方は、

- ・ ファイルを読み込む（たとえば、「外心」を読み込む）
（これをそのまま変形しても、軌跡は残らない。）
- ・ F 9 キーを押すことによって、左下に「軌」のマークを表示する。
- ・ そして変形する。

である。

なお、軌跡はメモリを消費する。特に、98などで、日本語を使用しているような場合は顕著だ。適当なところで、軌跡を消去しよう。

シフト + F 9 で軌跡の消去

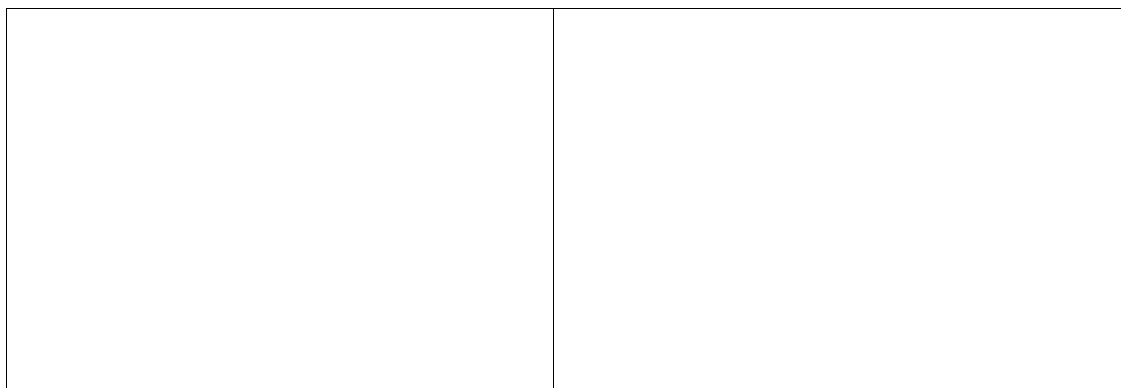


図 - 5.1 外心の軌跡

図 - 5.1 シフト + F 9 で消去

5.3 設定の方法

前述の図形は、事前に設定してあったから、変形するだけで軌跡が残った。しかし、一般には、自分で軌跡を設定する必要がある。重心の軌跡を設定してみよう。

- (1) 「重心.000」のファイルを読み込む。
- (2) 「軌跡の設定」を選ぶ。
- (3) 「点」を選ぶ。
- (4) 「点D」を選ぶ。
- (5) 色として「赤」を指定する。
- (6) Esc を何回か押し、標準メニューに戻る。
- (7) 「軌」のマークを確認し、必要があれば、F 9 キーを押す。
- (8) 変形してみる。

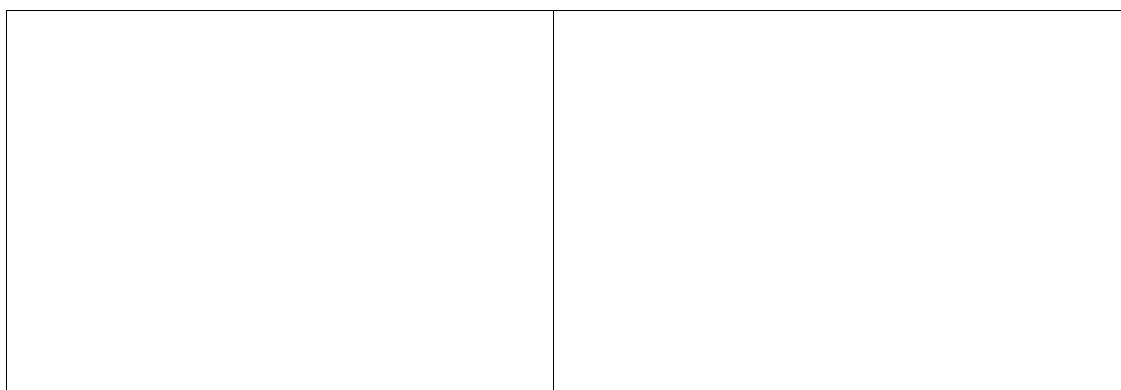


図 - 5.3 軌跡の設定

図 - 5.4 「点」を選択

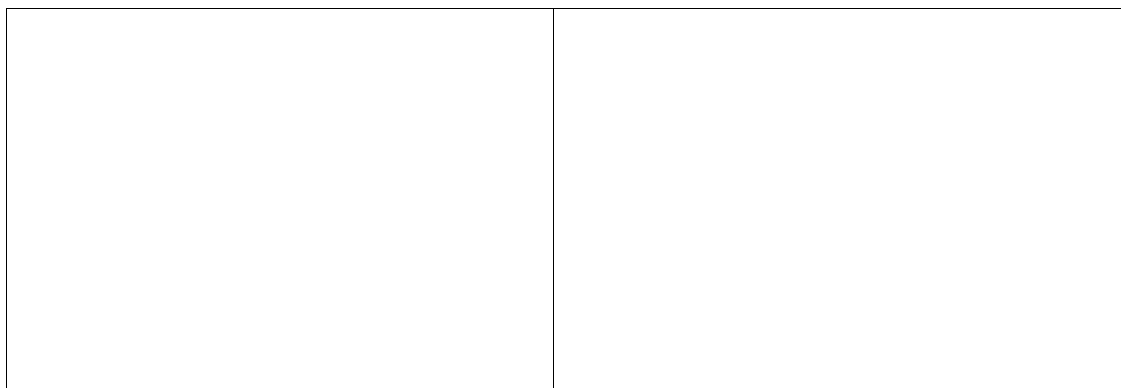


図 - 5.5 点の選択

図 - 5.6 色を選択

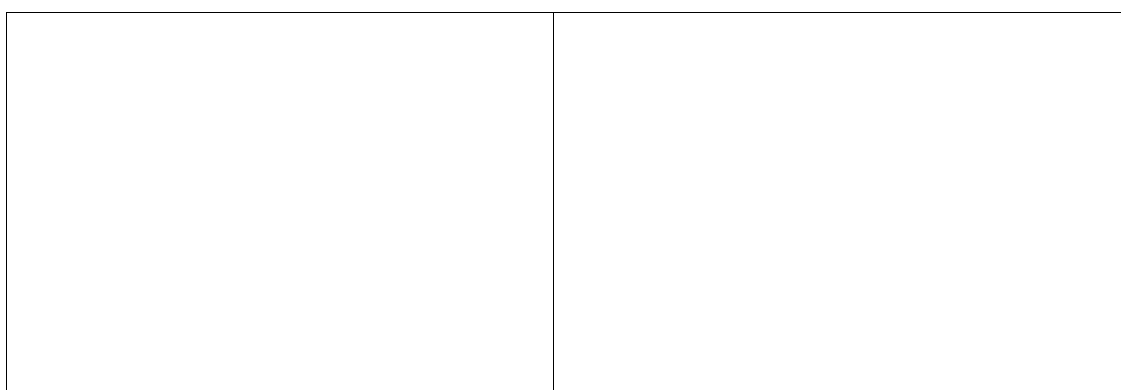


図 - 5.7 次の点の設定 (終了ならEsc)

図 - 5.8 軌跡の様子

5.4 練習問題

軌跡は意外性がある面白いものがいろいろありうる。

解説するよりも、自分で取り組んでみた方が面白いはずなので、練習問題として、いくつかをここで挙げておくことにする。なお、軌跡全般となるとあまりに多くなってしまうので、ここでは、「5心」の軌跡に関する話題をいくつかまとめておくこととする。右側は「スケッチ」用のスペースとして、空白をそのまま置いておこう。

(1) 点Aを自由に動かしてみよう。

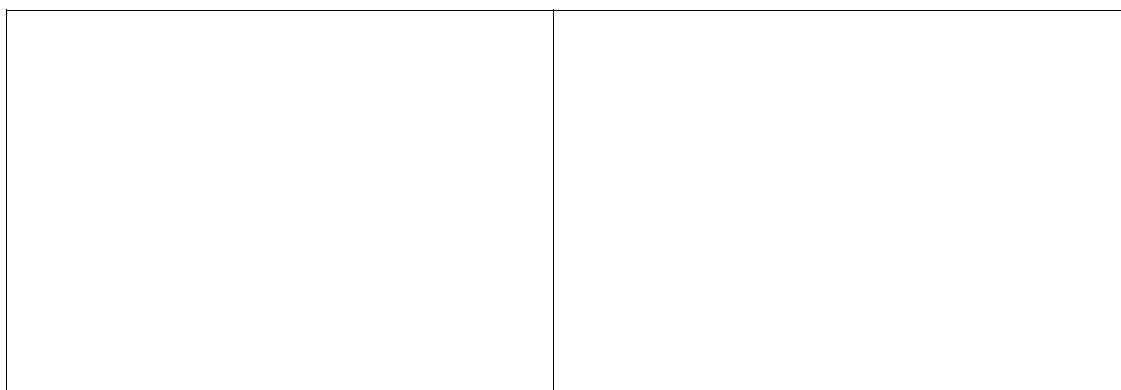


図 - 5.9 外心

スケッチ

(2) 点Aを水平に動かしてみよう。

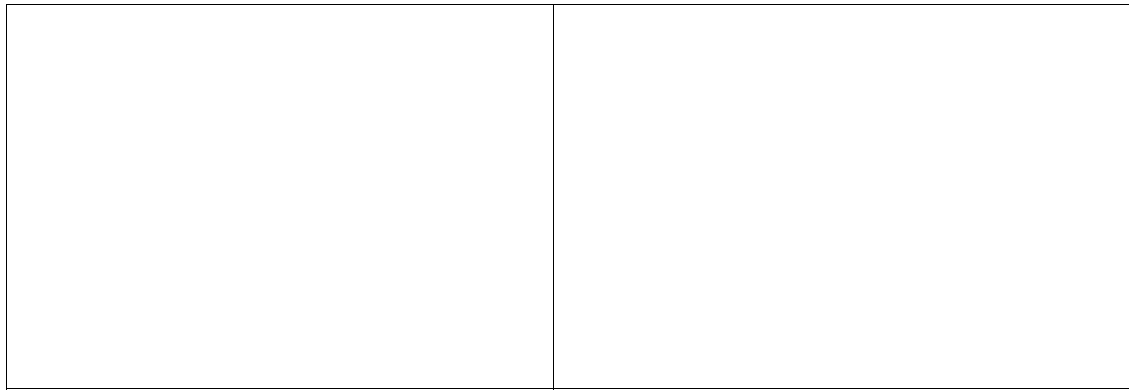


図 - 5.10垂心

スケッチ

(3) 点Aを水平に動かしてみよう。(楕円ではない)

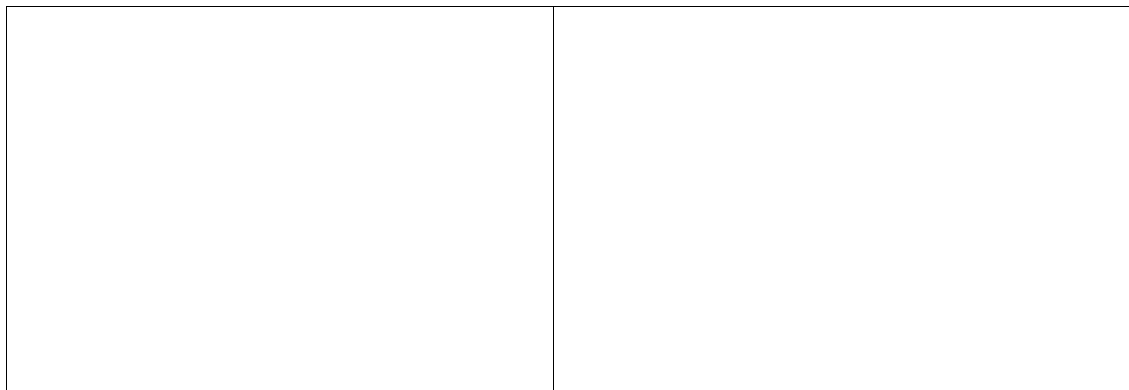


図 - 5.10内心

スケッチ

(4) 点Aを円上に動かしてみよう。

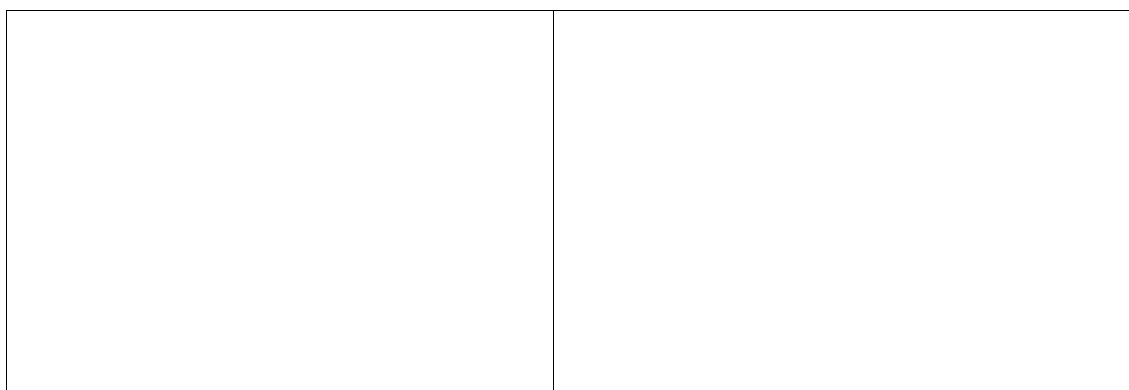


図 - 5.11内心

スケッチ

(5) 点Aを円上に動かしてみよう。

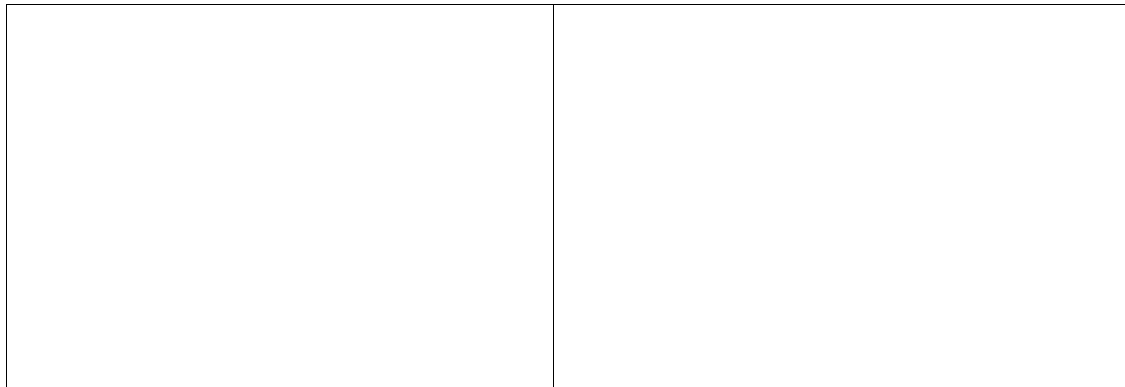


図 - 5.11内心傍心

スケッチ

(6) 点Aを円上に動かしてみよう。

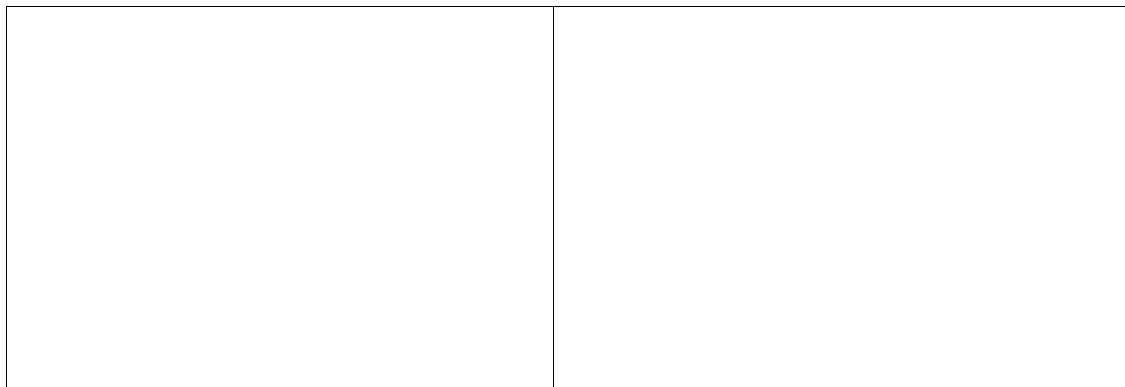


図 - 5.11内心傍心

スケッチ

5.5 直線等の掃過領域がなす図形

上記では、基本的に「点」の軌跡を考えましたが、線や円の「軌跡」が一つの図形を形成することもよくある。ここでは、それを「掃過領域」という言葉で表現する。

手続きは、「点」の場合の軌跡と同じである。例によって、考えてみよう。

(1) 垂直二等分線の軌跡(1)

たとえば、点A Bの垂直二等分線があるとき、点Aを直線上を動かすことを考えてみよう。すると、次のようになる。

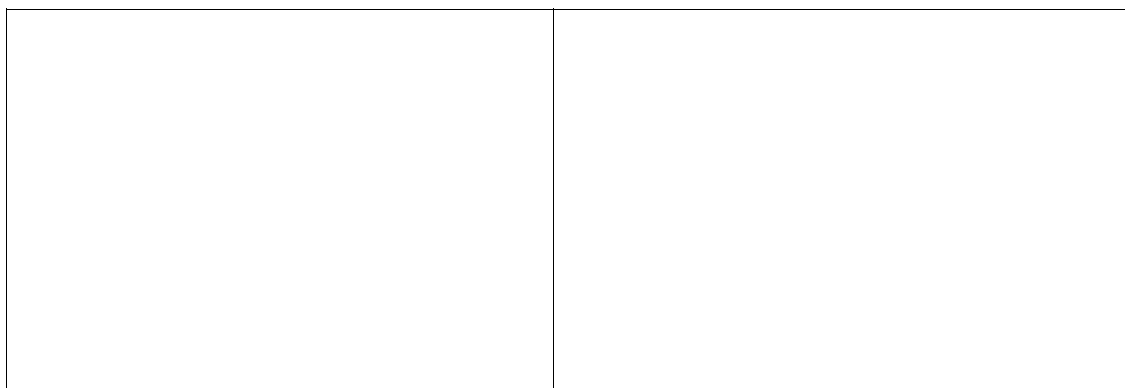


図 - 5.12垂直二等分線

図 - 5.13 軌跡

(2) 垂直 2 等分線の軌跡(2)

直線上でなく，円上を動くように変えてみよう。すると，次のようになる。

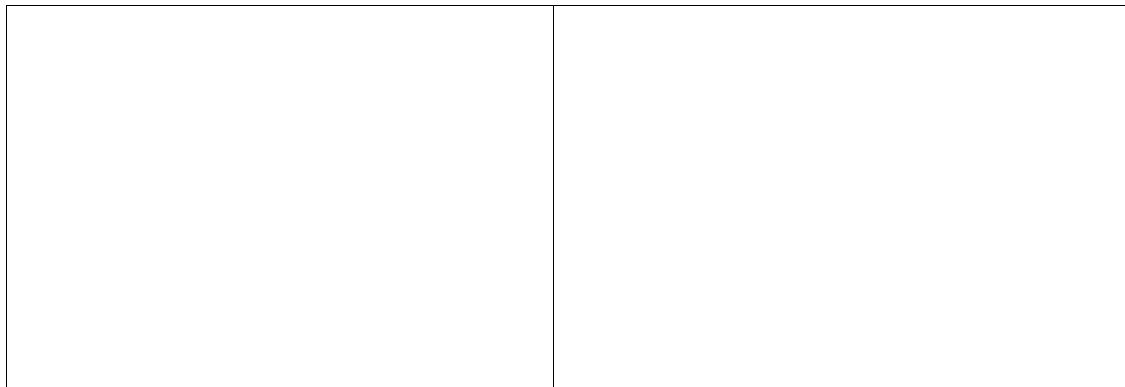


図 - 5.14 垂直二等分線

図 - 5.15 軌跡

(3) 2 点を通る円の軌跡

これはむしろ，中学校の教科書にある図の説明ということになるかもしれないが，典型的な例と言えらると思うので，挙げておく。

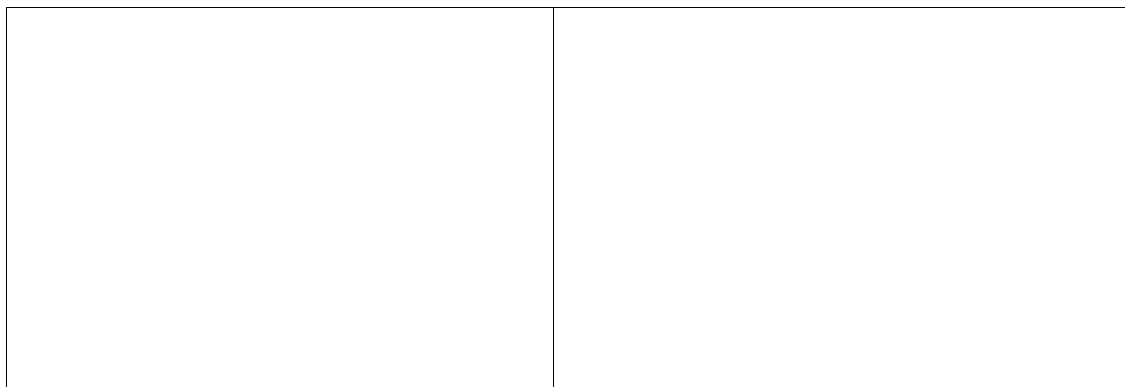


図 - 5.16 2 点を通る 2 円

図 - 5.17 軌跡

(4) 2 辺に接する円の軌跡

これも同様の図である。上記が外心の説明であるのに対して，こちらは内心の説明の図である。

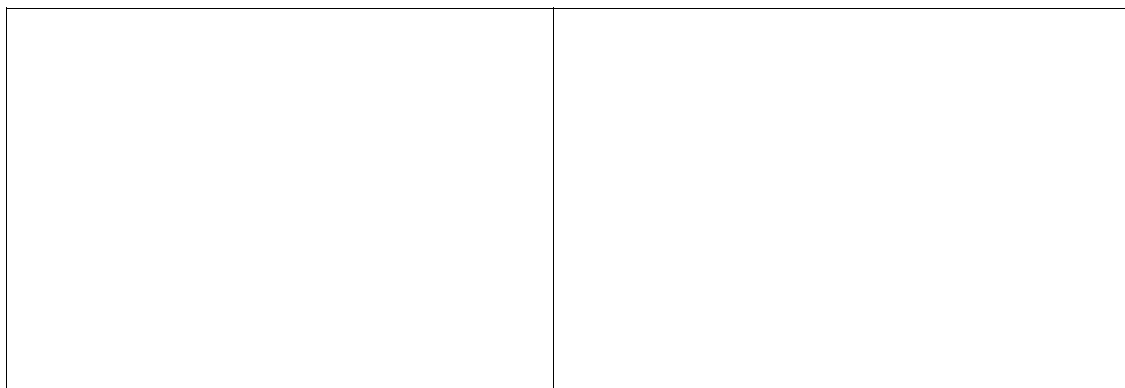


図 - 5.17 2 辺に接する円

図 - 5.18 軌跡

他にも，様々な例を作ることが可能だが，ここでは，この程度としておく。

6 . 変数 ・ 数式の利用

6.1 円の半径を変える

まず，半径が変わる円を作図してみることにしよう。いくつかの手がある。一つは，「中心と1点」によって，「2点」を元に作図し，その1点を移動することによって変形する方法だ。

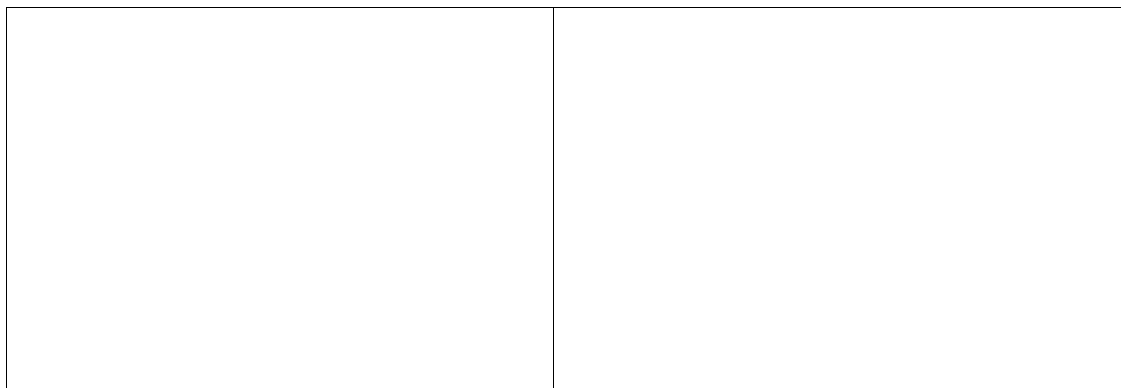


図 - 6.1

図 - 6.2

もう一つの考えは，「Aを中心にして，半径xの円を描く」という考え方を使う方法である。この場合，次のような手順となる。

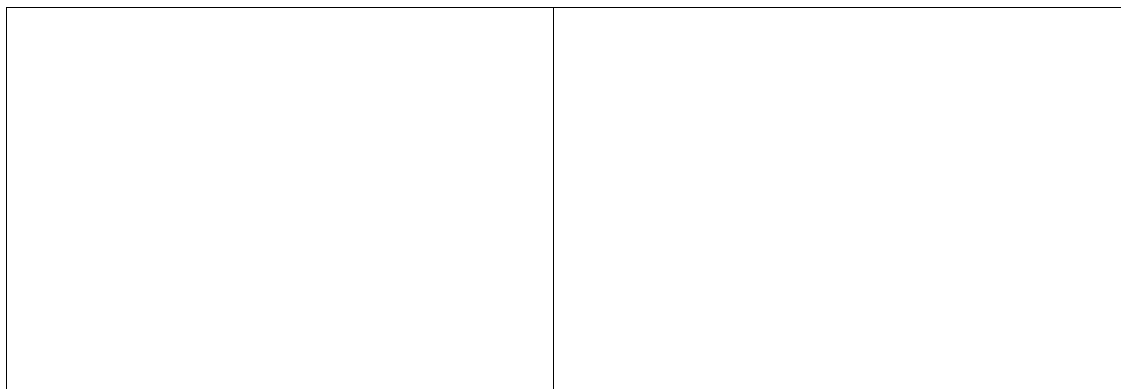


図 - 6.3 「円」 「中心と半径」

図 - 6.2 中心を決める

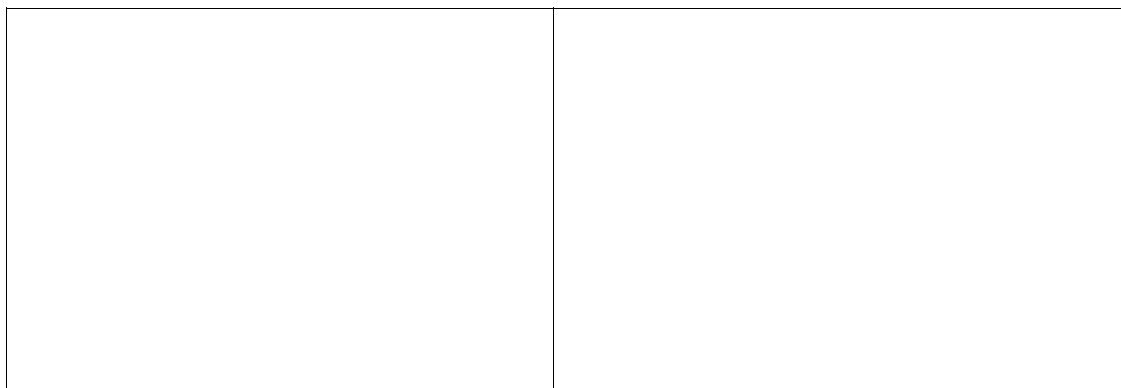


図 - 6.4 「新しい変数」を選択

図 - 6.5 数値の入力

すると，「数値の変化」を使えば，半径が変わる円を作ることが可能になる。

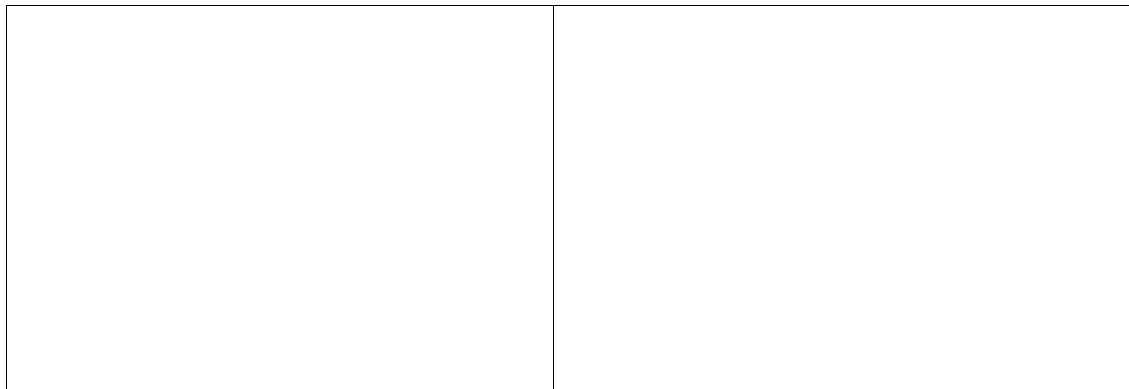


図 - 6.6 「数値の変化」を選択

図 - 6.6 変化の過程

6.2 一緒に変化させる

次に，2点から等距離にある点の集合を求めてみよう。6.1 の場合には，二つの方法のどちらを使ってもあまり差はないのだが，今回の場合は，

AとBを中心とする，それぞれ半径 x の円を作図し，その交点を作るという考え方が，一番自然である。

「両方とも，半径 x 」という考え方を，ここでは，「同じ数値を使って」と考え直すことによって，次のように作図を続けることができる。

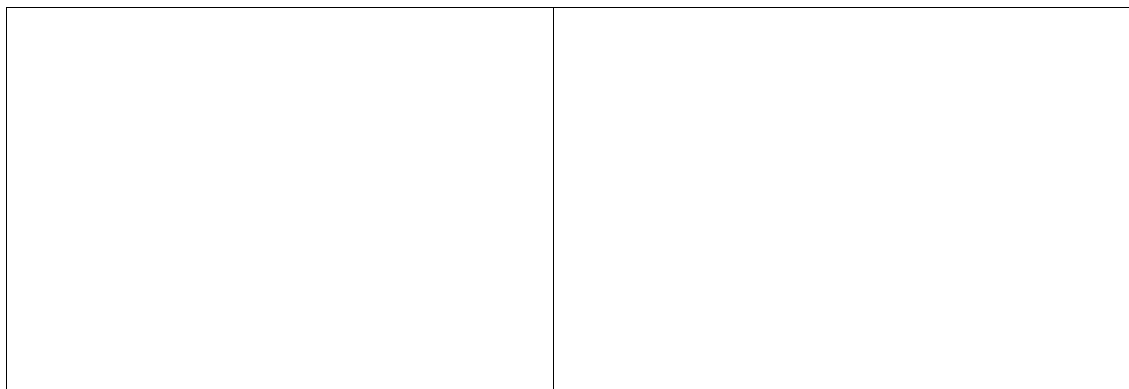


図 - 6.7 「円」 「中心と半径」

図 - 6.8 中心の選択

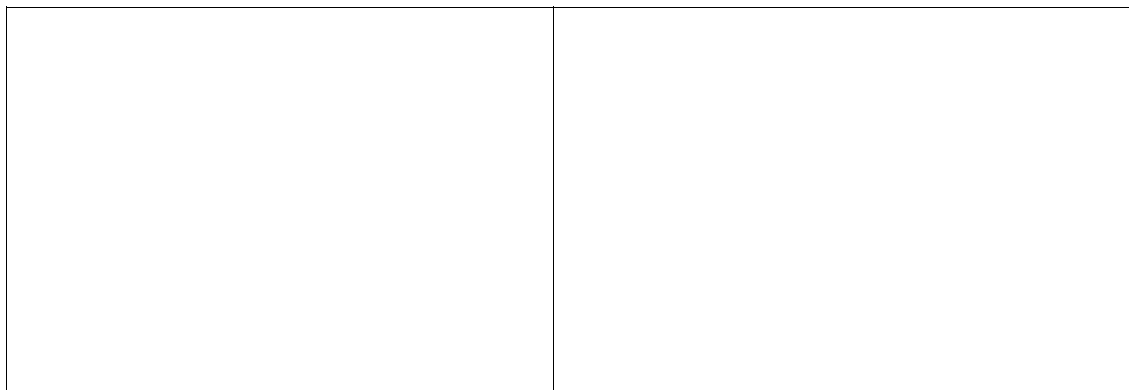


図 - 6.9 「今までの数値」の選択

図 - 6.10数値の選択

すると、同じ数値を使っているので、次のような変化となる。

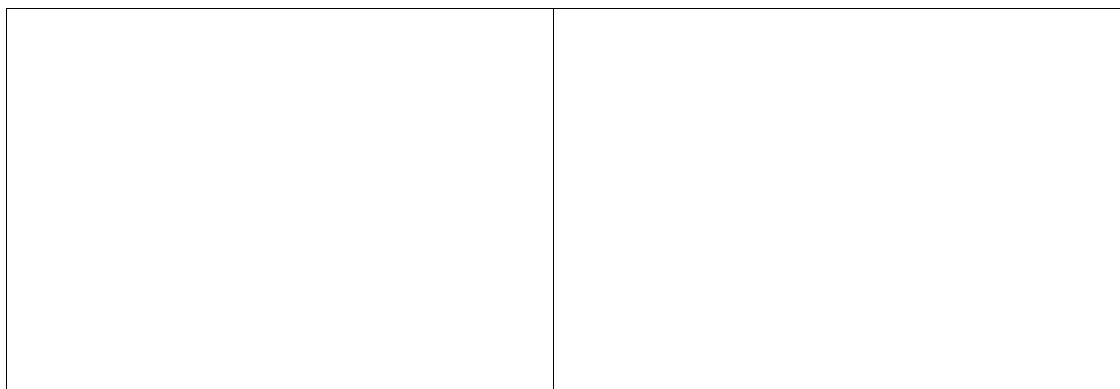


図 - 6.11 「数値の変化」を選択 図 - 6.12 変化の過程

さらに、交点の軌跡を求めると、次のようになる。

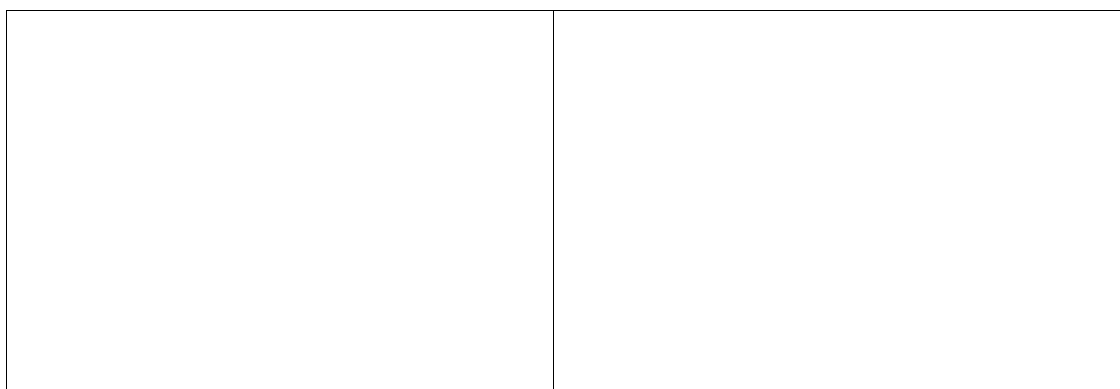


図 - 6.13 「点」 「2円の交点」 図 - 6.14 「軌跡の設定」 「点」 ..

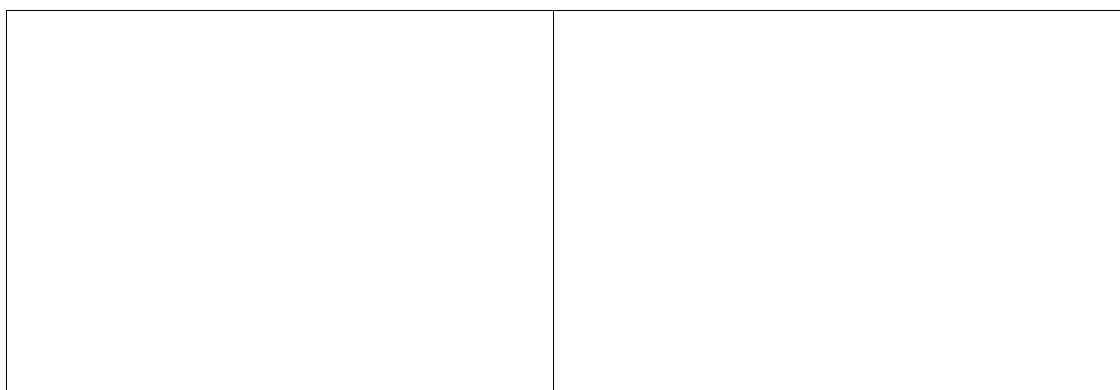


図 - 6.15 「数値の変化」を選択 図 - 6.16 軌跡

6.3 数式を変化させる

次に、楕円の作図を考えてみよう。楕円というのは、

$$PA + PB = \text{一定}$$

となる点の集合である。これを条件を満たす点の集合として作図する方法もあるが、

$$PB = \text{一定} - PA$$

と読み変えれば、たとえば $\text{一定} = 10$ の場合、

PAの条件：Aを中心として、半径 x の円を作る

P Bの条件：Bを中心として，半径 $10 - x$ の円を作る。
として，この2円の交点の軌跡として作図することが可能になる。

そのためには，「 $10 - x$ 」という変数を作らなければならない。これは，次のような手順で可能になる。

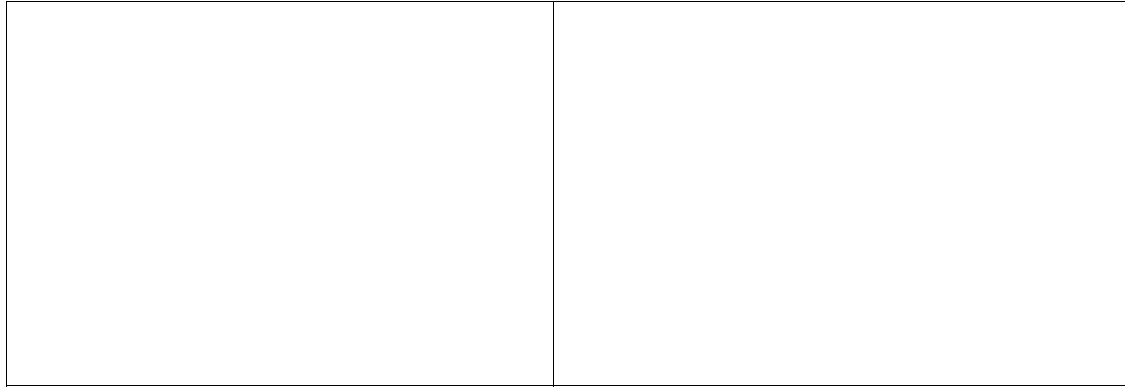


図 - 6.15 「数値の変化」を選択

図 - 6.16 軌跡

6.4 練習問題

次の条件を満たす図形を作ってみよう。

- (1) 2点A，Bからの距離の差が一定 (例： $PA - PB = 10$)
- (2) 2点A，Bからの距離の比が一定 (例： $PA = 2PB$)

7 . 条件を満たす軌跡による図形の描画

7.1 例

「条件を満たす集合」というのは、たとえば次のようなものを指す。

- | | |
|-------------------------------------|--------------------|
| (1) 二点 A , B から等距離にある点の集合 | A B の垂直二等分線 |
| (2) 一点 A から等距離にある点の集合 | A を中心とする円 |
| (3) $APB = \text{一定}$ となる P の集合 | A B を弦とする円 |
| (4) $AP + BP = \text{一定}$ となる P の集合 | A , B を焦点とする楕円 |
| (5) $AP - BP = \text{一定}$ となる P の集合 | A , B を焦点とする双曲線の一葉 |

7.2 OHPシート作戦

これらについて、Geometric Constructor などで調べる場合、特に変わった機能を使わなくても調べることはできる。一つの発想としては、

すべてをソフトに頼るな

ということが挙げられる。たとえば、測定機能はソフトの内部にあります、

ディスプレイ上の図を定規で測って何が悪い

とも言える。もちろん、ディスプレイ上の図は多少歪んでいたりする可能性はあるわけで、ある意味ではナンセンスですが、私達の目的が「CADを使った製図」のようなものではなく、「図形の問題について考えるという数学的探究」であることを考えると、

簡単な方法でとりあえずアバウトな結果を得て、概要を知る

というのは、目的に合わせて考えてみると、妥当である。それは一般に、ツール型のソフトの利用全般についても一般化可能なことであり、つまり、

機能に関する学習は最低限にしたい

考える場面や作業内容として生徒にあったものを確保したい

ことが、授業で使うときには一つの原則となる。

話を元に戻すと、条件を満たす点の集合を調べるためには、

条件を満たす点の位置を画面上に「プロット」してしまえ

というのが、一つの手となる。しかし、ディスプレイにマジックや鉛筆で書かれたらたまらない。だから、

ディスプレイにOHPシートを張りつけ、そこに書き込め

というのが、一つの手となる。この手は、教育的には別の効果がある。つまり、

重ねてOHPで提示すると、何人分もの結果を「集約」することができる

ということだ。

一人で調べるのは大変だけど、みんなの結果を集めたら、ここまで分かる

という「演出」ができるという意味でも、OHPシート「作戦」と呼ぶに値する。（この方法は、川崎市内で開発・洗練された。）

7.3 「一人」でするにはたまらない

しかし、この手は「授業向き」であっても、自分で調べるための方法としては、かなり無理がある。特に教材研究等をする場合には、とりあえず、いろいろな可能性について調べ、その中から選択するわけで、一つの問題に対する概略的な結果は、

あまり考えなくても、作業が進められる

のでないと、疲れてしまう。それを実現するための機能として、Geometric Constructorでの「条件を満たす点の集合」がある。

7.4 アイデアとしての「領域の境界」

この機能を理解する上で、補足になると思うので、追加するが、OHP作戦の場合でも、たとえば、

「 $APB = 60^\circ$ となるときに点を取れ」

と言うと、ほとんど点を取ることができない。特に、測定の桁を小数点以下2位くらいまで表示していたら、生徒はそれにこだわって、「 60.01° でもだめじゃないか」と思う。

川崎で議論していたときに、こういうアイデアが出た。

「 60° 以上のときに青い。以下の時に赤い を書かせよう」

（授業のときに実際に扱ったのは、別の事例）

これは二つの意味で面白いアイデアでした。

- (1) 60° の場合が、「青と赤の境界」として現れる
- (2) 調べた結果、 60° より大きいところ、小さいところの結果も、書き込まれる
（そうでないと、「調べていない」のと区別がつかない）
（また、生徒が書き込める回数も増える）

7.5 Geometric Constructor における「条件を満たす点の集合」の機能

Geometric Constructor での機能は、上記の作業を部分的に自動化したものである。基本的には、

ある変数の符号の変化に応じて、動かしている点の軌跡を変える
というものである。

そのため、 $APB = 60$ となる場合を調べるには、数式機能を使って、

「 $APB - 60$ 」

について調べることになる。

調べる手順は、次のようになる。

- (1) 作図する。

(2) 測定する。必要があれば、数式を作り、その符号で判定可能にしておく。

(3) 「軌跡の設定」で「変数」を選択し、(2) の変数を選択する。

(4) F 9 のスイッチを確認。

(5) 変形してみる。

このとき、キーボードを使った場合には、等間隔に記号が並ぶが、マウスの場合には、もっと細かく調べたり、概略を調べられる反面、あまり綺麗にはならない。

また、描画の最中に、画面が汚くなるが、Esc キーを押すと、描き直すので、あまり気にしないで、まずは変形をしよう。

この機能を使う場合も、メモリに対する配慮は必要なので、「画面全体を調べよう」というようなことはやめ、大体調べたいところが分かったら、「シフト + F 9」でクリアしよう。

また、9 8, F M の場合には、F 7 キーで画面をフロッピー等に保管し、あとでまとめてハードコピーを取ることなどができる。

7.6 ケーススタディ - 円周角の場合 -

(1) $APB = 60^\circ$ となる場所を探そう

まずは、「 60° 」の場合を、マウスを動かしながら、探してみよう。

さて、どの程度大変なことだろうか。あるいはどの程度簡単なことだろうか。

そして、たとえば10分という作業時間の中で、どの程度のことができるだろう。

(そういう「体験」をまずしておかないと、授業化は難しい)

(2) 桁数を変えてみる

あまり本質的なことではないが、必要に応じて、桁数を変えることもできる。

「オプション」 「桁数の設定」

で、たとえば、「0」を設定してみよう。あるいは、「20」にしてみよう。

(3) 「条件を満たす点の集合」を調べてみる

60° にしてもいいのだが、「この角と等しい」という対象が見える方がいいので、

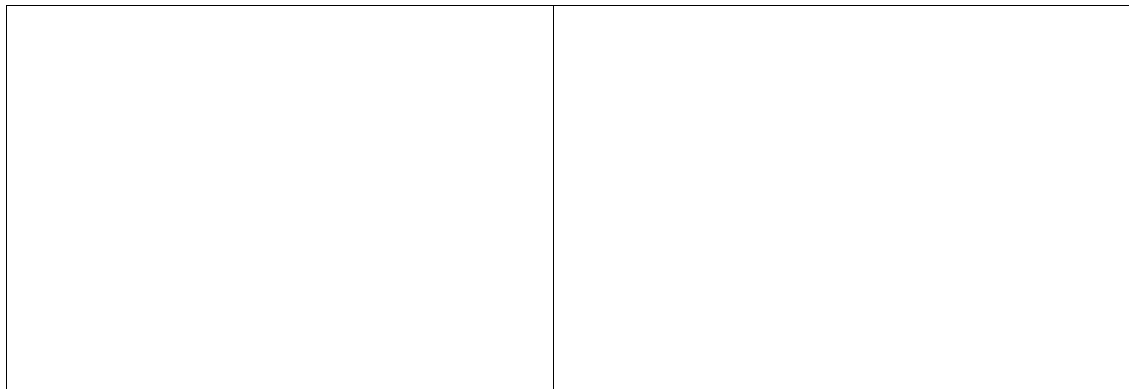
$$\angle APB = \angle AQB$$

つまり、

$$\angle APB - \angle AQB = 0$$

となる場合を調べてみよう。

ここでは、キーボードを使った方がいい。



7.7 練習問題

次の条件を満たす点の集合を，求めてみよう。

- (1) 二点 A , B から等距離にある点の集合
- (2) 一点 A から等距離にある点の集合
- (3) $AP + BP = \text{一定}$ となる P の集合
- (4) $AP - BP = \text{一定}$ となる P の集合

8 . 軌跡によるグラフの描画

8.1 関数を調べる手段としてのGeometric Constructor

作図ツールは「図形」だけとお感じの方もいらっしゃるかもしれない。しかし、測定機能等を使うと、関数を調べるのに適した場面も少なくない。

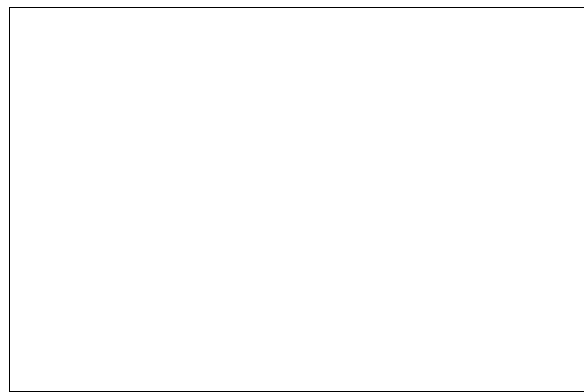
そのような使い方を考える場合には、

「完全」でない方が教育的に適している場合が多い

ことに注意してみよう。

たとえば、次の問題を考えてみよう。

問題 右図のような四角形 $ABCD$ がありその辺上を動点 P が毎秒 1 cm の速さで、 $B \rightarrow C \rightarrow D \rightarrow A$ の順に動く。このとき、 P が AB の面積を S とすると、 S の変化の様子をグラフに表せ。



この問題に対して、「ボタン一つを押せば、グラフを描画してくれるソフト」というのは、どの程度効果的だろうか。「正解を確かめる」程度の価値しかないと思う。では、作図ツールをつかうとどうなるだろう。まずは、上図のような図を作り、点 P を作ることになる。

折れ線上に制限することはできないではないか

と問題点を指摘される方もいらっしゃるかもしれない。確かにそうだ。そして、場合によっては、このことは致命傷にもなりえます。しかし、 BC 、 CD 、 DA を縦横にしておけば、キーボードの矢印キーを押すことで、線分上の点を取ることができる。そして、「グラフ用紙に記録せよ」と指示を出せば、その様子は、関数として記録することができる。こういうときには、

コンピュータの結果ですべてを知ろうというのではなく、概略を知りたい

という目的で行う場合がほとんどなので、「概略」を知るのに、最も適切な方法が必要なのである。そのためには、何も「きっちり線分上」である必要はなく、「動きを制限」することの方がむしろ不適切であり、適当に矢印キーを押して、それぞれの場合を「プロット」し、関数関係を予想する方が意味があると言える。

8.2 大きさを視覚化する

また、「数値で観察」することが、面倒臭いように感じる場合もある。そういう場合には、変化を具体的に視覚化してとらえることによって、かなり改善する余地が生まれる。たとえば、「中心と半径」によって円を扱う場合のことを考えてみよう。

円以外にも、棒グラフのようなものなど、いろいろ作れますが、円が一番簡単だ。

--	--

--	--

8.3 グラフとして表現する

たとえば，次の問題を考えてみよう。

問題

右の図で， $PA + PB$ が最小になる位置を求めなさい。

これを解くには，普通は，よくある例の方法が考えるわけだが，この方法では，最小値の位置しか分からない。それを，「関数のグラフ」の形で知るには，たとえば，以下のような方法がある。

--

--	--

数値のみ

円で表示

--	--

グラフ化

グラフ化（軌跡として）

8.4 練習問題

(1) 8.3 の問題で，2乗の和が最小になる場所を探してみよう。

（満足のいく「答」がある。）

(2) 8.3 の問題で，「直線」を「円」に変えてみよう。

（大体の場所はコンピュータで分かるが，数学的にきちんと処理する方法を私は知らない。逆に言えば，そのような，「どうなるか分からない」場合について，実験的に結果を求め，それを元に推論を進めるようなときにはかなり有力である。）

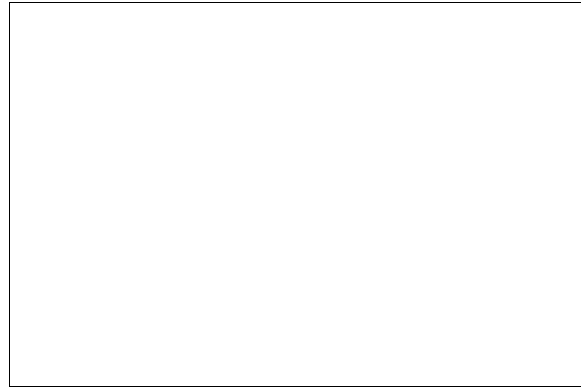
9 . 変換を調べるための作図

9.1 変換を調べる手段としてのGeometric Constructor

(1) 次の問題は本質的には「変換」の問題

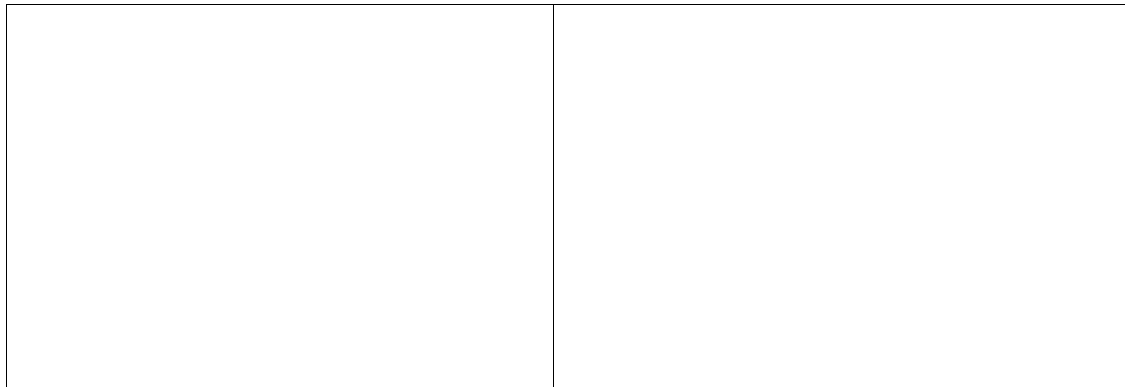
問題

点Pを半直線OXについて線対称移動した点をQ, Qを半直線OYについて線対称移動した点をRとします。XOYが 40° のとき, PORは何度でしょう。



つまり, この問題では, 線対称移動を2回合成すると, 何変換になるでしょうというのが, より本質的な問題である。

ちなみに, Pを動かしてみよう。QやRの軌跡はどうなるだろう。



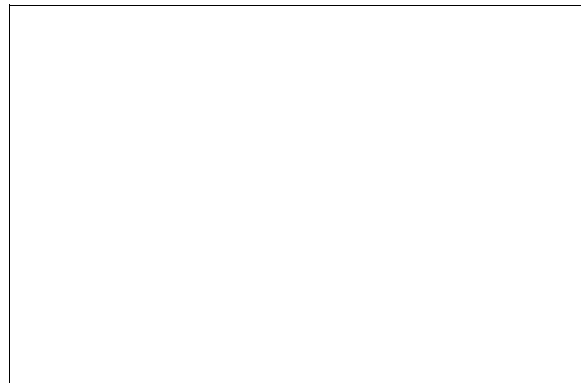
観察結果

このように, 現行の教科書の問題の中にも, 「変換」として意識する方が妥当な問題もないわけではない。

9.2 別の例

問題

定点Aと直線lがあり, l上に点Bがある。ABを一辺とする正三角形ABCを作り, 点Bを直線上を移動するとき, Cの軌跡を求めよ。



これも、「正三角形」ということにとらわれていると予想ができないが、「 60° 回転」ということが分かれば、なんということもない問題となる。

「直線」という条件を変えるとどうなるだろう。また、「正三角形」という条件をどのように変えると、どのような結果が生まれるだろう。



観察結果

9.3 「点と点の対応」はすべて「変換」

その他にも、様々な「点と点の対応」として作図することによって、変換を実現することができる。また逆に、「点と点の対応」をつけるような作図であれば、実質的に、それは何らかの変換となる。また、変換そのものとして意識しなくても、それが関与するような、新しい問題を見つけることも時には可能である。

具体的に記述すると、かなり長くなってしまっているので、ここでは、いくつかの例を次の練習問題として載せるにとどめておこう。

9.4 練習問題

- (1) 平行な2直線がある。線対称移動を2回繰り返すとどうなるか。
- (2) 四角形と1点Pがある。四角形の4点に対して、Pを点対称移動すると、どんな四角形ができるか。
- (3) 四角形と1点Pがある。四角形の4辺に対して、Pを線対称移動すると、どんな四角形ができるか。
- (4) 四角形と1点Pがある。PをAについて点対称移動した点をP1、P1をBについて...としてできたP4はPと一致することがあるか。あるとすれば、元の四角形あるいはPについて、どういう条件が必要か。
- (5) (4) を三角形について考察してみよう。

10 . 複素数平面上的の関数としての作図

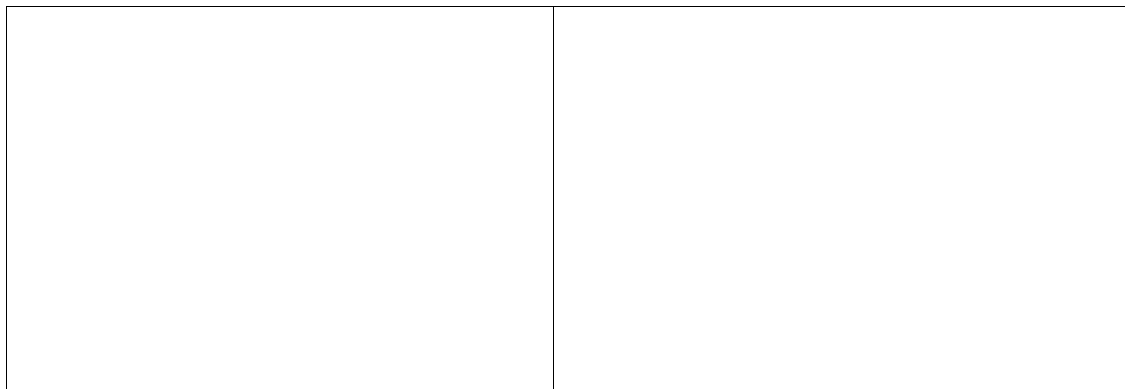
10.1 平面は複素数

これは、前提のようなことだが、特に中学校の先生方にとっては、普段接していない考え方なので、改めてここに書いておく。つまり、 (x,y) という点を、 $x + y i$ という複素数と対応させれば、平面と複素数の世界として見ることができる。

Geometric Constructor は、平面図形を扱っていますが、特にその中の「点」に注目すれば、それは、「複素数」そのものでもあるわけで、必ずしも「平面幾何」あるいは、ユークリッド的な幾何に限定されるわけではないことが分かるだろう。

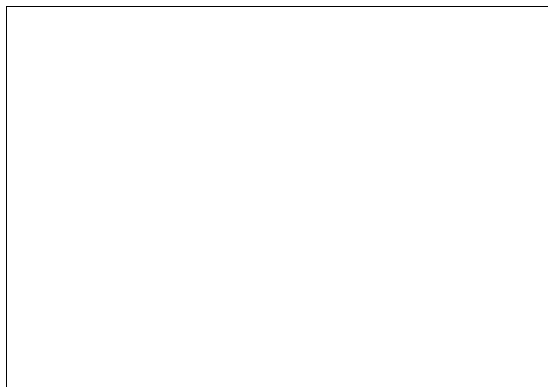
10.2 少し「ズーム」して「保持」しておくと便利

また、複素数を扱う場合には、単位円の回りでいろいろな変化が起こることが多いが、Geometric Constructor のデフォルトでは、少し細かすぎる。複素数を扱う場合には、「ズーム」ボタンを2回程度押し、「保持」のために「/」キーを押すと便利である。



「ズーム」

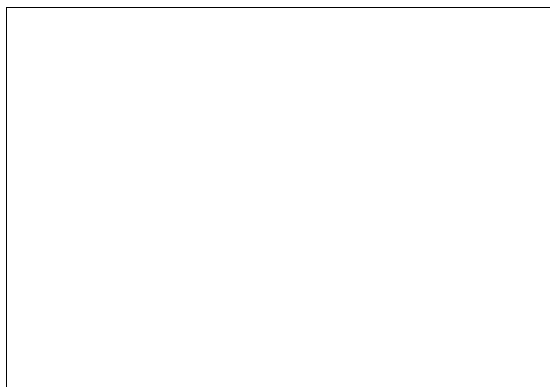
「/」で保持



「標準」に戻る

10.3 「和」を調べる

「和」はベクトルとしての和を調べるだけのことだから、あまり面白くはないかもしれない。

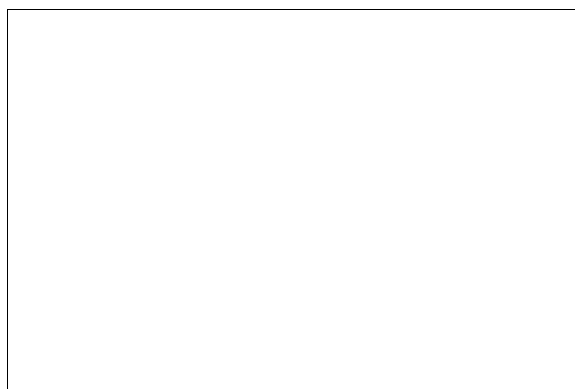


10.4 「積」を調べる

次の図では、 A, B に対して、 $A*B$ が作図されているわけだが、たとえば、

- (1) A を実軸上を動かすと、 $A*B$ はどういう動きをするか。
- (2) A を虚軸上を動かすと、 $A*B$ はどういう動きをするか。
- (3) A を適当な直線上を動かすと...
- (4) A を円上を動かすと...

というような問いを考えることができる。

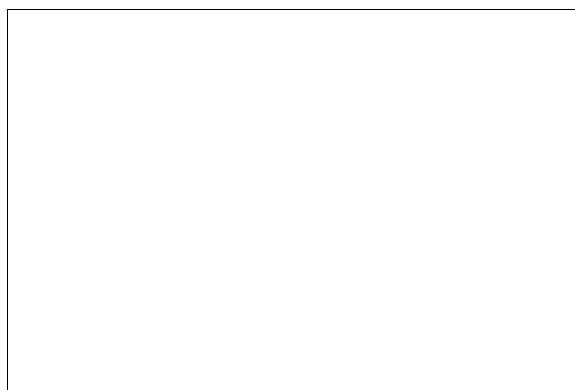


10.5 「逆数 $1/z$ 」を調べる

次の図では、 z に対して、 $1/z$ を作図している。さて、

- (1) z を直線上を動かしたとき、 $1/z$ の軌跡はどうなるだろう。
- (2) z を円上を動かしたとき、 $1/z$ の軌跡はどうなるだろう。

というような問いが作れる。



10.6 複素数の関数（多項式）の性質を調べる

- (1) 素朴な疑問

たとえば、2次方程式

$$x^2 + x + 1 = 0$$

を考えてみよう。

この「解はあるだろうか」。

実数の範囲を考えれば、「ない」。実際、グラフを描いてみると、

$$y = (x + 1/2)^2 + 3/4$$

だから，大体，次のようなグラフになる。x 軸との交点がないのだから，解がないというのは，とても妥当なように思える。

さて，解の公式を使って，

$$x = \frac{-1 \pm \sqrt{1 - 4}}{2}$$

を考えると，ルートの中で負だったら，そこを i を使えばいいじゃないという雰囲気もするし，実際，高校では，それが妥当なものとして扱われるわけだが，さて，

「その解はどこにあるの」

という素朴な疑問が湧かないだろうか。

あるいは，

任意の多項式について，その解を実験的に求める方法はないのだろうかと思わないだろうか。

実際，関数のグラフが納得いく人にとっては，形式的に定義することや，そう定義すると解になることは納得できても，何か合点がいかないなあと，不思議に思えるはずだと思うのだが。

(2) 多項式を「平面から平面への対応」として「作図」する

さて，

$$x^2 + x + 1 = 0$$

について考えるために，

$$f(z) = z^2 + z + 1$$

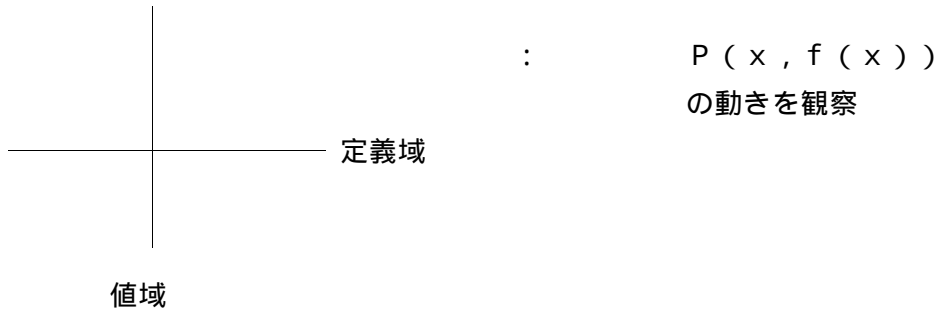
を考えることにする。Geometric Constructor を使うには，

- z : 独立な点
- z^2 : 「巾」によって作れる点
- $z^2 + z$: z^2 と z の和
- 1 : 定数 (点)
- $z^2 + z + 1$: ($z^2 + z$) と 1 の和

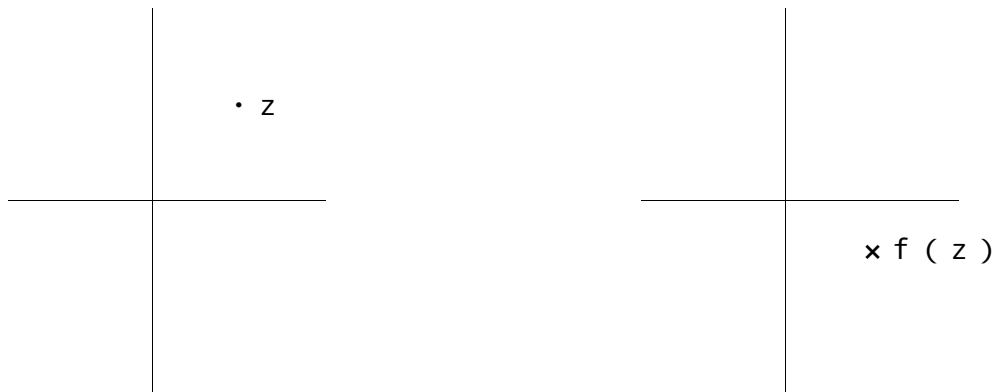
と考えれば，作図をすることができる。対応が明確に分かるようにするためには，他の点は「消して」しまう方がいいと思う。

(3) z を適当に動かしてみる。

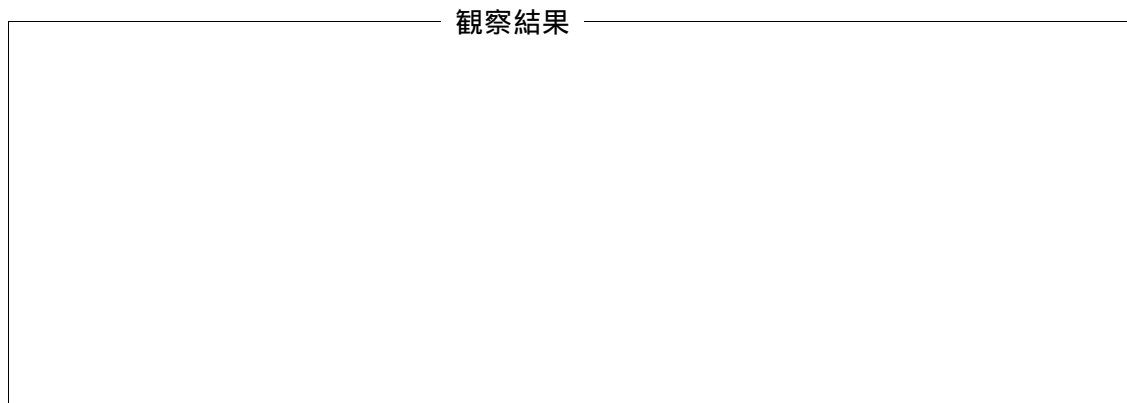
まずは、 z を適当に動かしてみよう。動かす点と動く点と二つある。一方は定義域にあり、一方は値域にあると思おう。実数値関数の場合には、



ということになるが、複素数でこれをするには4次元が必要になるため、



という二つを並列したものを、更に重ねていると思おう。



(4) z を実軸上を動かしてみよう

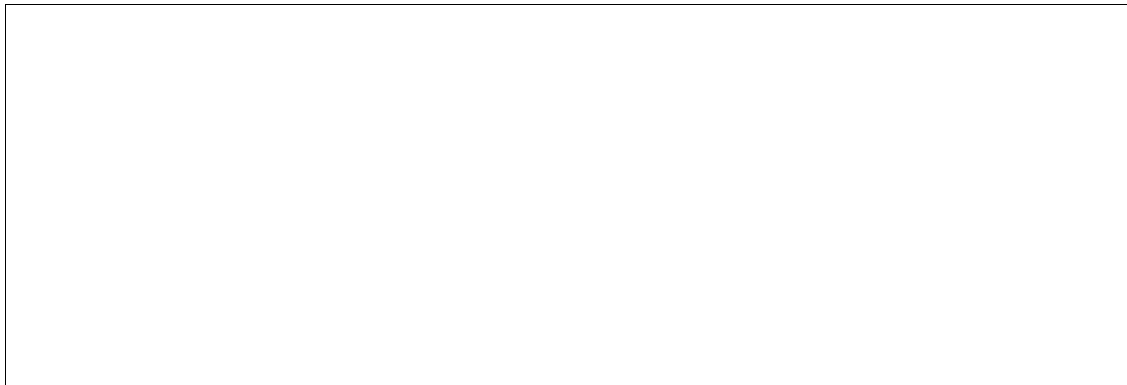
z を実数軸の上を動かしてみると、普通の実数値関数をこのような表しかたで表すとどうなるかを観察することができる。 $f(z)$ は右の方から原点に近づくけれども、また戻ってしまうことが観察できると思う。



観察結果

(5) z を虚数軸上を動かしてみると

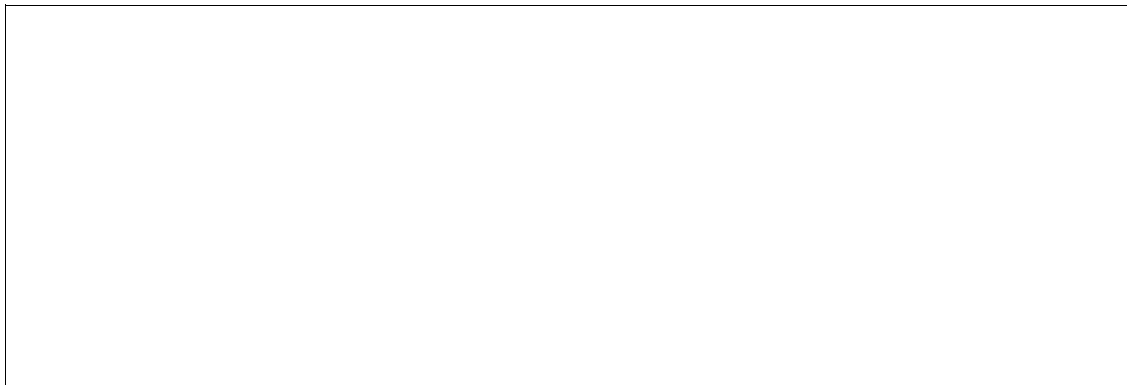
次に、虚数軸を上を動かしてみよう。また、実数軸に平行に動かしたり、虚数軸に平行に動かしてみたりして、その軌跡を観察してみよう。



観察結果

(6) 単位円上を動かしてみると

さらに興味深いのは、単位円の上を動かした場合である。



観察結果

(7) 他の多項式についても、同様にして、実験的にいろいろと調べられる。

こんな具合に、いろいろな多項式に関して、その値を実験的に調べることが可能である。

10.7 練習問題

(1) $f(z) = z^2 + z + 1$ のように、係数が実数の他の多項式について、
x 軸，y 軸，単位円の像を調べてみよう。

また、係数を純虚数にしてみると、どうなるだろう。

また、両方の成分を持つ複素数にしてみるとどうなるだろう。

ということは、係数が、実数の場合は、どういう特徴があるのだろうか。

(2) $f(z) = z^2 + z + 1$ を原点を中心とする単位円以外の円上で動かし、その軌跡を調べてみよう。

a. 解の中に含んでいるときは、どういう像となるだろう。

b. 解が円上にあるときは、どういう像になるだろう。

c. 解が円内にはないときは、どういう像になるだろう。

(3) 3 次式などについて、解と像との関係を調べてみると、どうなるか。

11. 様々なソフトでの作図との比較

11.1 Logo との比較

作図に関する重要な言語にLogoがあるが、LogoとGeometric Constructor の相違点を述べておく。

(1)汎用言語ではなく、より限定されたアプリケーションである。

Logoはそれ自体が一つの言語である。その中で、再帰的なプログラミングが可能であり、タートルを使いながら命令を再帰的に構成できる点に特徴があるが、Geometric Constructor では、そのような、汎用のプログラミングを目指してはいない。

別の観点から述べるなら、Logoにはあまりに自由性があるため、多少慣れないと、それなりのプログラミングをすることが難しい。特に初等幾何的な図形を作図したり、それを動かすためにはある程度の時間を要する。それに対してGeometric Constructor では、主として初等幾何的な図形に限定して、より容易に、適切な作図あるいは、その動的な扱いを支援することを目的としている。

(2)動的な図形の捉え方が異なる。

Logoが表現している幾何学はタートル幾何学であり、長さと角度の幾何学である。あるいは、点のベクトル的な移動の幾何学と言うこともできる。換言すれば、Logoで考えられている動的な図形とは、ちょうど、微分方程式の解曲線として曲線を捉えることができることを元としている。

それに対して、Geometric Constructor においては、主として初等幾何における作図が構成的であること、そしてその依存関係を明示すると、ある定数を変数化したときに、全体の図形を動かすことができるという点を基本としている。そのような意味において、Geometric Constructor は物理的な教具を一般化して実現したものに他ならない。

(3) (タートル + 向き + 長さ) vs. (幾何的对象 + 構成)

したがって、利用者が注目すべきものの自体が異なっている。Logoの場合、重要な役割を果たすのはタートルであり、それを自分の動きと同化して、その向きや移動の長さを決めていくという手続きである。それに対してGeometric Constructor においては、幾何的对象、つまり図形を構成している要素を、点、直線、線分、半直線、円に分解して、どれがどれによって構成されているのかを明確化することを目的としている。

以上の3つが基本的相違点である。実際の学習においては、小学校段階でLogoの方が使われやすく、中学校の方でGeometric Constructor の方が使われやすいと想像されるが、必ずしも発達段階に依拠しているわけではない。むしろ、両者が表現しようとしている幾何学的なアプローチが異なる点が重要である。そして、両者それぞれにおいて扱いやすい内容と扱いにくい内容がある。

実際、多角形の外角の和の性質などはLogoの方が扱いやすいし、また動きの単位をより

微小化することによって、点の軌跡としての曲線の性質、そしてそれを規定するものとしての微分方程式や差分方程式を扱いやすいのである。そして、後者に注目させるためには、幼少の時期よりも、むしろ高校生や大学生になってからのほうが妥当かもしれない。また、Geometric Constructor の場合、初等幾何的なものを扱うことは容易であると考えられるが、Logoが表現しようとしている点の軌跡としての曲線の性質などを表現することは難しい。さらに、両者のみで幾何学における図形に対するアプローチが十分かと言えば、もちろんそんなことはない。簡単な例を上げれば、位相的な見方はどちらを具現化していないし、カバリエリの原理のようなものでさえ、表現できない。すなわち、重要なのは、幾何に対するアプローチが多様であるのに応じて、様々な図形処理の仕方が実現可能であるということ、そして様々なアプローチをさらに明確にして、それぞれのアプローチは、教育の中でどのような活動を支援することができるかを明確にしていくことなのである。