目 次

1 Tgif の概要 **1**
1.1 機能の呼び出し - メニュー - . . . 2
1.1.1 モード メニュー . . . 3
1.1.2 線の属性 . . . 4
1.1.3 ペン種と塗りつぶし - Pen & Fill - . . . 4
1.1.4 フォント . . . 5
1.2 保存方法 . . . 5
1.3 日本語の入力 . . . 5
1.4 アレンジメニュー . . . 6
1.4.1 グループ・アングループ . . . 6
1.4.2 フロント・バック . . . 6
1.4.3 整列 . . . 7
1.4.4 グリッド・グリッド スナップ . . . 7
1.4.5 ズーム . . . 8
1.4.6 キー入力コマンド . . . 8

2 テクニック上達 **9**
2.1 基本要素を描く . . . 9
2.2 四角 , 円 . . . 10
2.2.1 単純な組み合わせ . . . 10
2.2.2 Fill–White の応用 . . . 10
2.3 折れ線で描けるもの . . . 12
2.4 ベジエ曲線の応用 . . . 13
2.5 変形 . . . 15
2.6 xbm,xpm,eps 画像の読み込み . . . 16
2.6.1 xpm 形式で画面のダンプを取る方法 – xgrab – . . . 17
2.7 GNUPLOT からの Tgif-obj 形式出力 . . . 18
2.8 写実的に見せる工夫 . . . 19

3 図面集 **20**

1　Tgif の概要

実験装置などの図面も論文には必要です。図面は大きくペイント系 (ビットマップとして一点ごとの色で保存) とドロー系 (オブジェクトとして、形と位置を保存) に分かれます。ペイント系では任意の絵が描けますが、拡大縮小に耐えません。一方、ドロー系では出力先の大きさに合わせて綺麗に拡大縮小が可能ですから、余り複雑でない装置の概念図などはドロー系のツールで書くのが得策でしょう。

X 上のツールとしてここでは tgif の使い方を学びます。全体に操作に不統一な点がありますが、図を綺麗に描くという目標をもって基本的な図の作成方法を学ぶには十分な機能があります。

では、tgif を引数なしで起動しましょう

```
tgif &
```

画面右上にツールバーがあります。図形や機能を選択するのには、ツールバーの上段左端のアイコンにカーソルを合わせて、マウスの左右ボタンどちらかを押します。するとアイコンが変化して、図形や機能が選択されます。図では "T" が現れてますが、これは文字列が選択されていることになります。このメニューは Mode メニューと呼ばれます。

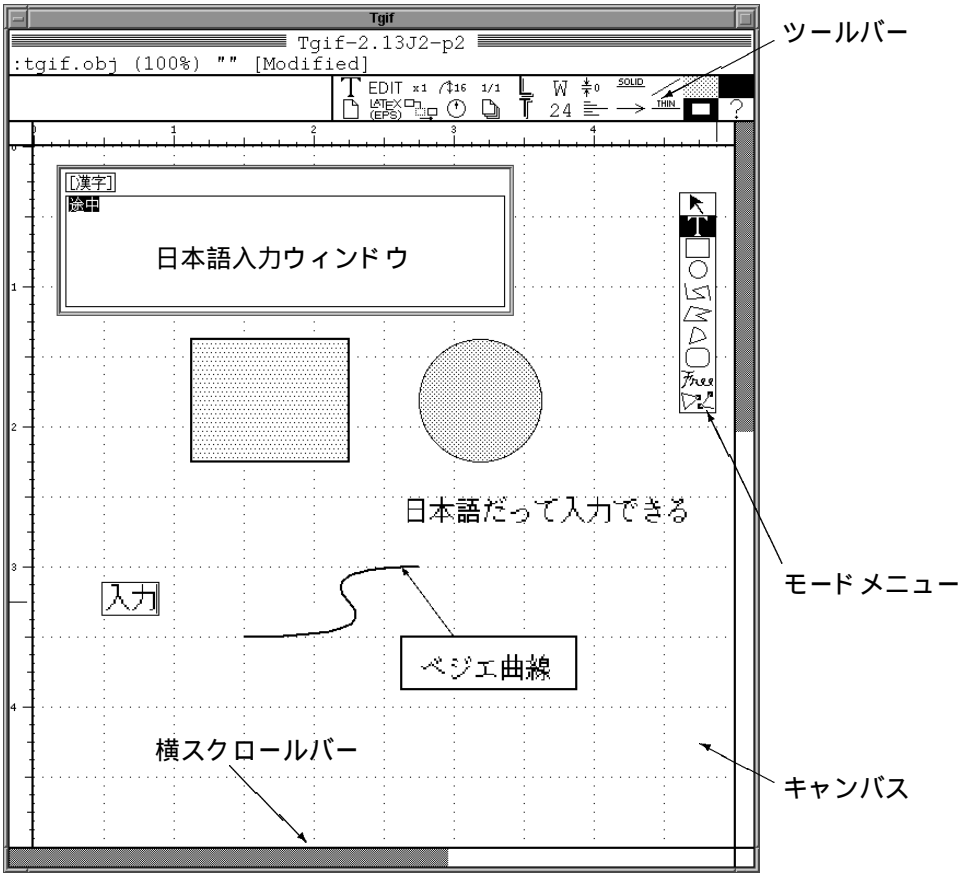


図 1　tgif の起動画面

キャンバス上でマウスの右ボタンをおすと、上記の mode が一覧になったメニューが現れます。図のキャンバス右上にあるのがそれで、マウスドラッグによりその中から選択することができます。

1.1 機能の呼び出し - メニュー -

tgif では、以下に示すように機能の選択を複数の方法で行うことが可能です。

- ツールバーのアイコン (=機能) を変化させる
- ツールバーでメニュー (機能の一覧) を呼び出す
- キャンバスでメニューを呼び出す
- キャンバスでキー入力により機能を直接指定する

これはユーザーの習熟度に合わせて最適な機能選択が行えるように配慮されたものですが、逆に操作が複雑であるかの印象を与えます。ここでは、ツールバーのアイコンをマウスの中ボタンでクリックしてメニューを呼び出す方法を中心に説明します。まず、呼び出されるメニューを図 2 と図 3 に示しておきます。個々のメニューを見ながら、図の描き方について説明していきます。

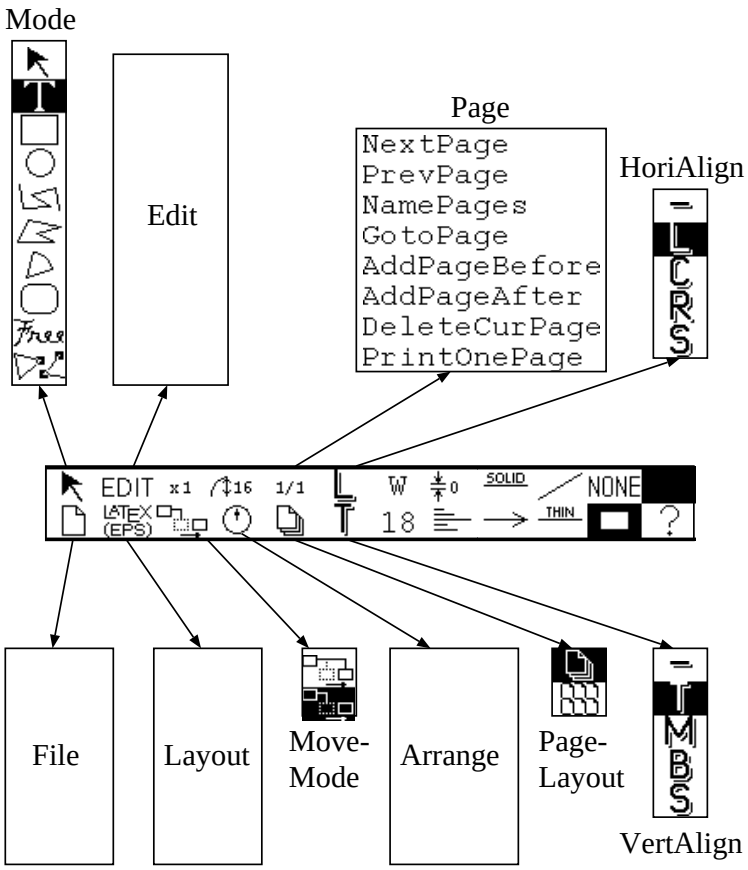


図 2 ツールバーのアイテムを中ボタンでクリックして現れるメニュー (左半分)

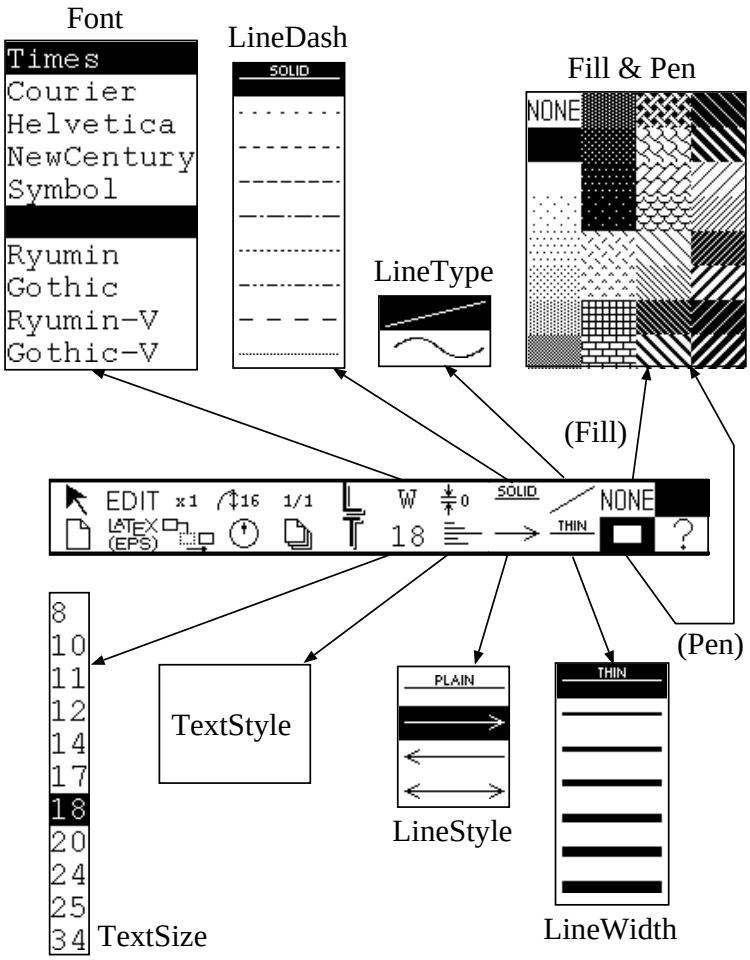


図 3　ツールバーのアイテムを中ボタンでクリックして現れるメニュー (右半分)

1.1.1　モード メニュー

最初は、もっとも頻繁につかう Mode です。アイコンを見ると大体想像できますが、描く図形を選びます。ただ、一番上の ”矢印” は矢印を描くのではなく。図形の選択 : 移動や複写や拡大縮小の対象として選択状態にすることの機能を呼び出すものです。また、一番下は、ちょっとわかりにくいですが、折れ線の折点の削除 : 折れ線を描いた後に、余分な折点を指定して削除するを行いうためのモード です。他は、見たとおり、上から順に文字・四角・楕円・折れ線 (閉じない)・折れ線で閉じられた図形・扇型・丸四角・フリーハンド です。

図の描き方ですが、例えば四角の場合は対角の 2 点 (始点と終点) を指定します。始点位置で左ボタンを押してマウスカーソルをドラッグすると、カーソルが指に変わって、始点位置と現在のカーソル位置を対角とした四角の外形がラバーラインでトレースされますから、自分の望む大きさになったら、左ボタンを離して確定します。簡単ですね。楕円と丸四角とフリーハンド が四角と同じように、左ボタンのみの操作 (押してドラッグ離して確定) で描けます。文字も、描きたい位置 (左端) を指定するだけなので、左ボタンのみです。

線分を描くときには、右ボタンも使います。左ボタンを押して線分の端 (折点) を指定し、右ボタンで終了です。したがって、ドラッグ (ボタンを押し下げたままカーソルを移動させる) は意味を持ちません。閉図形では、始点と最後の点を結んで閉じた図形が完成します。

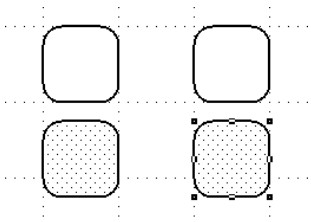


図 4　図形の選択：選択されると右図下の丸四角の様にマークが付けられる。

1.1.2　線の属性

図形を描く線の種類 (実線や破線)・太さ・タイプ (直線と曲線)・両端形状 (矢印) を変えることができます。これらは、ツールバーの右から 3,4 番目にあります。デフォルトは実線 (Solid)、細線 (Thin)、直線、終端矢印となっています。

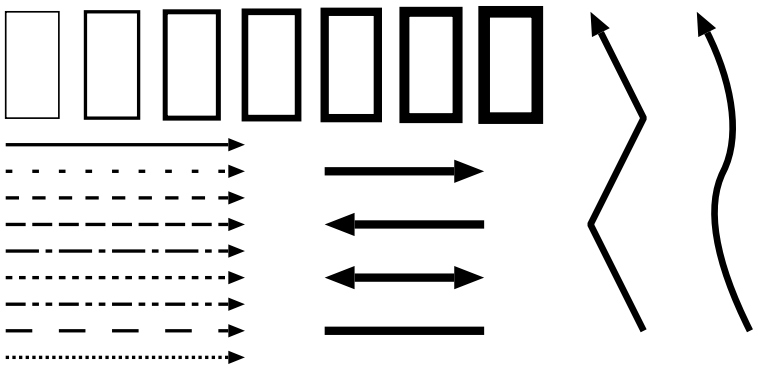


図 5　線のいろいろな属性

1.1.3　ペン種と塗りつぶし - Pen & Fill -

右から 2 番目にあるのが塗りつぶし (上段 : Fill) とペン種 (下段 : Pen) の指定ですが、全く同じメニューが現れます。すなわち、指定できる属性が同じということです。図形の外形を描くのに使う線の属性が Pen で、囲まれた図形内部の塗りつぶし属性が Fill です。デフォルトでは、Pen は黒 (枠) ですから分かるのですが、 Fill は None となっています。None とは何も色を塗らない = 透明ということです。したがって、図形を重ねた場合、下の図形が透けて見えることになります。

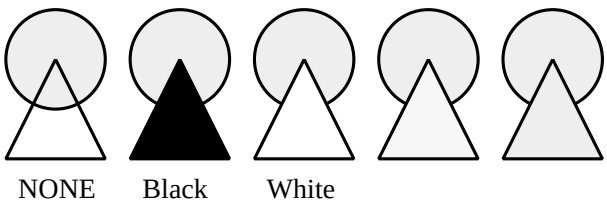


図 6　いろいろな塗りつぶし (ペン種は黒)

1.1.4　フォント

文字列を描く時にはフォントの種類・大きさ・スタイルを選びましょう。難しいことはないのですが、あまりいろいろな種類のフォントを使うと煩わしいので、塩梅を考えて使いましょう。また実際に印刷される文字はプリンターに依存しており X 上の画面出力とは異なっている場合もありますから、プリンター印字出力結果をよく確かめるよう心掛けてください。英数フォントの書体は、**Times, Courier, Helvetica, NewCentury, Symbol** から選べますが、プリンターによっては印字出来ないものもあるようです。

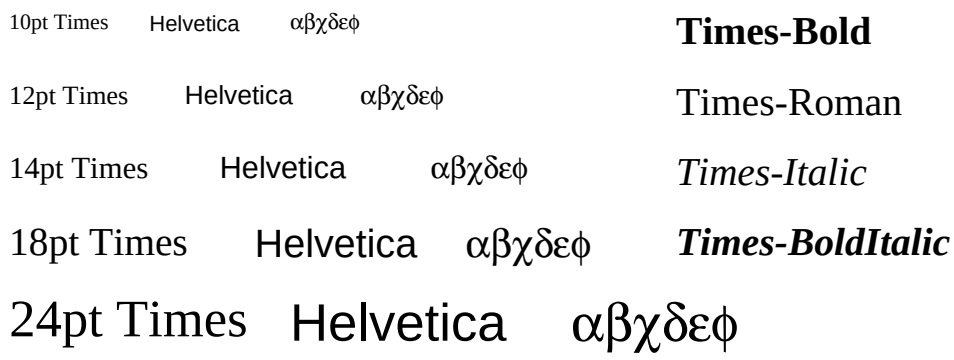


図 7　いろいろな英数フォント

1.2　保存方法

図を作成したら保存しましょう。それにはファイルメニューで "save" を選択すればいいのですが、この時の保存形式は tgif 独自の形式で、 拡張子が **.obj** となります。LaTeX で使うためには EPS 形式のファイルが必要ですから、後でこの obj ファイルを EPS ファイルに変換します。それには、

```
tgif -print -eps filename.obj
```

と tgif をコマンドラインで使うことにより可能で、*filename.eps* が得られます。

もう 1 つ方法があります。それは、File メニューで "**print**" を実行する方法です。ツールバー下段左から 2 番目のアイコンが print 時の出力形式です。デフォルトは "LaTeX(EPS)" となっています。他にも、PS ファイル、XBM ファイル、PS プリンターへの直接印刷が選択可能ですから、EPS ファイルとなっていることを確認してください。もちろん本来の obj 形式での保存を忘れないでください。後で tgif で読み込んで編集可能なファイルは obj 形式だけですから。

> 保存のついでに、読み込みについて話します。tgif はビットマップ画像を読み込むことができます。ファイルメニューの 6 番目に **ImportXBitmap**、 7 番目に **ImportXPixmap** があります。これらを選択するとファイル選択画面が現れますから読み込みたいファイル名前を入力します。そして、読み込んだ画像に手を加えて、EPS で保存すれば LaTeX で利用できることになります。また 8 番目には **ImportEPSFile** があり、確かに直接 EPS 画像が読み込めますが、残念なことに枠のみの表示で内容は表示されません。

1.3　日本語の入力

日本語化されている tgif では当然のことながら日本語が描けます。フォントメニューの下半分にある **Ryumin, Gothic, Ryumin-V, Gothic-V** が日本語のフォントです。これらフォントを選択するだけで

は日本語は入力できません、そう日本語入力システムが必要なのです。変換エンジンは Mule 同様 **canna** ですが、X 上の入力クライアントとして **kinput2** を使います。具体的な手順は、

(i) mode を文字列描画にする。

(ii) 文字を描きたい場所で左ボタンをクリックする。

(iii) Ctrl–SPC により kinput2 を呼び出す。

です。日本語入力ウィンドウが現れれば成功です (図 **1** 参照)。ただし、kinput2 は `.xinitrc` に次の 1 行を追加して起動させておかなければなりません。

```
kinput2 -canna &
```

1.4　アレンジメニュー

図面が正確であることはもちろん、見た感じを美しく仕上げるためには個々の要素を描いた後に全体をアレンジする必要があります。そのためのメニューが用意されています、項目は

> **Front, Back, Group, Ungroup, Lock, Unlock, AlignObjs, AlignToGrid, AlignToPage, DistributeObjs, FlipHorizontal, FlipVertical, RotateClockWize, RotateCounter, AlignObjsTop, AlignObjsMiddle, AlignObjsBottom, AlignObjsLeft, AlignObjsCenter, AlignObjsRight, AbutHorizontal, AbutVertical, CenterAnEndPoint**

となっています。沢山あって全部は説明できませんから、頻繁に使う主なものについて簡単に説明していきます。

1.4.1　グループ・アングループ

個々の要素をまとめて 1 つの図形にする機能がグループです。グループ化すると全体がまとめて移動できますし、拡大・縮小も可能となります。むろん線分の属性変更などはグループ全体に及びます。したがって、グループ内のある要素を変更したい場合はグループを解除してそ要素のみを選択しなければなりません。

グループ化のためには複数の要素を選択状態にできなければなりません。それには、Shift キーを押しながら左ボタンをクリックします。

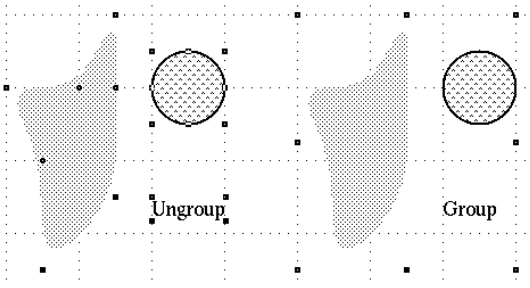


図 8　グループになると個々のマークが全体のマークに変化する

1.4.2　フロント・バック

オブジェクトは時間的に後で描いたものが図面上は前面にきます。図面上でのオブジェクトの前後関係を変化させる場合にこれらの命令を使います。

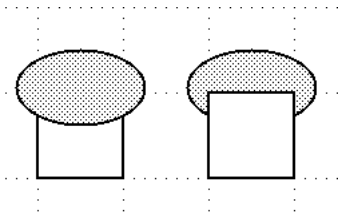


図 9　前後関係を変える

1.4.3　整列

オブジェクトを整列させます。垂直方向の位置関係は上端 (Top) 下端 (Bottom) 中央 (Middle) のいずれかで揃うように指定します。水平方向は左端 (Left) 中央 (Center) 右端 (Right) の中から指定します。単に **AlignObjs** を行うと左上が揃います (Top と Left を行ったのに等しい)。

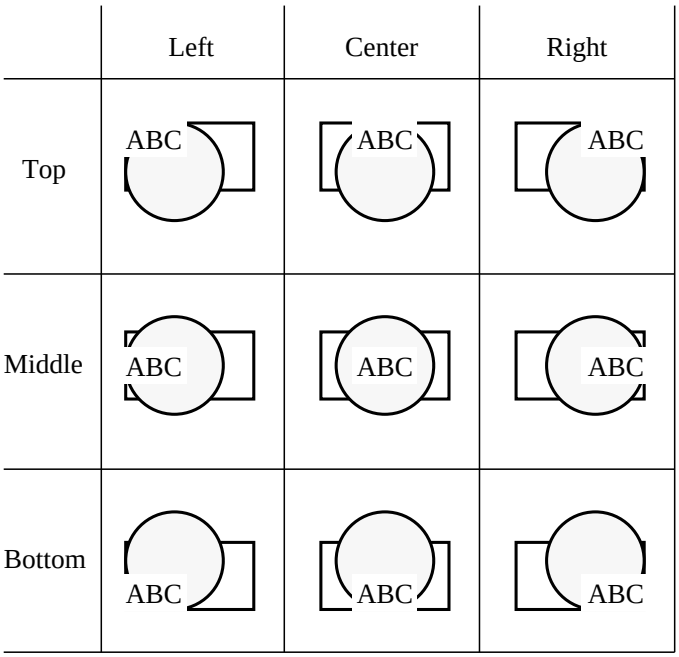


図 10　整列の様子：垂直方向 3 通り水平方向 3 通りの組み合わがある

1.4.4　グリッド・グリッド スナップ

図形を整然と並べるためには、全く任意の位置におけるよりは、ある決まった位置にしか置けないようにするべきです。そのために正方のグリッド が用意されており、グリッド 位置にしか置けないモード ：グリッド スナップ on 時には、図形 (枠の左上隅) はグリッド の位置にのみ置かれます。図ではその粗さが、左から 1/8 (デフォルト), 1/16, 1/32 インチの場合のルーラーの様子を示します。また、逆にグリッド スナップ off 時にも全く任意の位置に置ける訳ではなく、1/128 インチ単位 (最も細かい) でしか位置決めできません。

この項目はレイアウト メニューに属していますから、迷わないようにしてください。

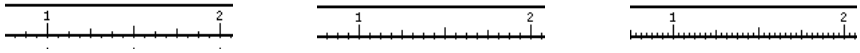


図 11　ルーラーはグリッド単位の大きさを常に表示している

1.4.5　ズーム

ズームインで表示拡大率を大きくすると細かい作業ができます、またズームアウトして拡大率を小さくすれば全体が見渡せるようになります。旨く使い分けて作業の効率をあげましょう。

1.4.6　キー入力コマンド

メニューをたどってコマンドを実行するのが煩わしくなってきたら、キー入力にコマンドが割り当てられていますから、それを使いましょう。基本的には Ctrl and/or Alt (tgif のメニューでは#と表記されている) キーと 1 文字を入力します。

表 1　Tgif のキー入力コマンド

キー入力	機能	キー入力	機能
C-n	新規ファイル	C-f	オブジェクト前面
C-o	ファイル開く	C-b	オブジェクト後面
C-s	ファイル保存	Alt-z	拡大
C-p	出力 (プリンタ、EPSF)	Alt-o	縮小
C-q	終了	Alt-:	デフォルトの拡大率
C-d	オブジェクト複写	Alt-p	挿入
C-x	オブジェクト削除	Alt-(	EPSF ファイルの挿入
C-a	全オブジェクト選択	C-Alt-g	スナップ (トグル)
C-u	Undo	C-Alt-s	別名でファイル保存
Alt-i	グリッドを粗くする	Alt-d	グリッドを細かくする

2　テクニック上達

写真とは異なり，図面は概念を的確に伝えるための強力な手段です。したがって，細かすぎず粗すぎず適当な模型化が必要です。そのためには，複雑な図形を単純な図形の組合せに分解する能力が必要です。そこで，まず基本的と思われる図形の描き方を十分に練習しましょう。

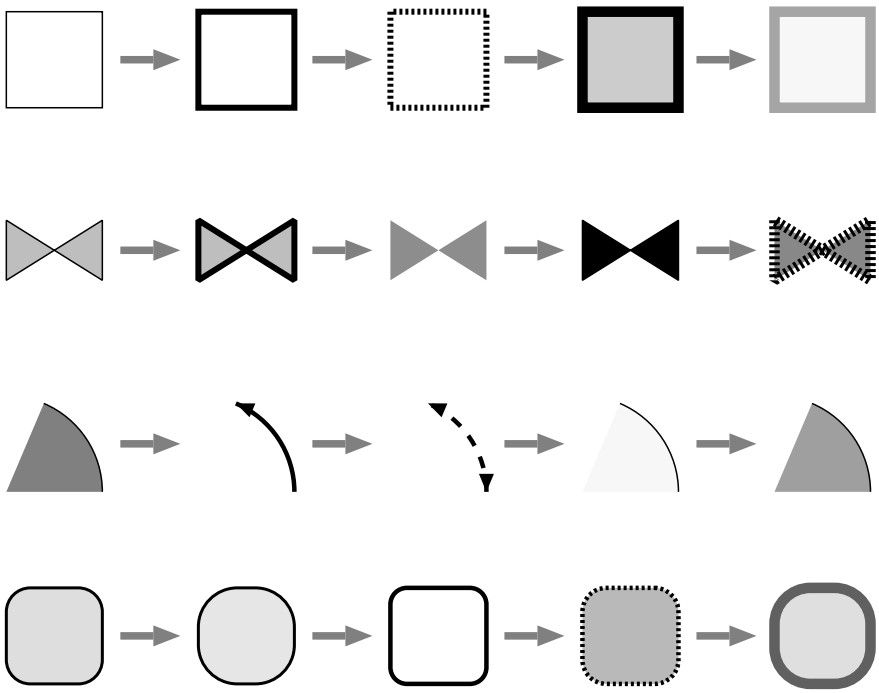
2.1　基本要素を描く

例題 1

メニューにある基本図形には，文字，四角，円 (楕円)，折れ線 (線分)，閉じた折れ線，扇形 (円弧)，丸四角，フリーハンド が用意されています。

- それぞれ適当に描いてみなさい。
- 線の太さや線の種類を変えてみなさい。
- 内部や外枠を変えてみなさい。

基本要素の描画実行例



2.2　四角，円

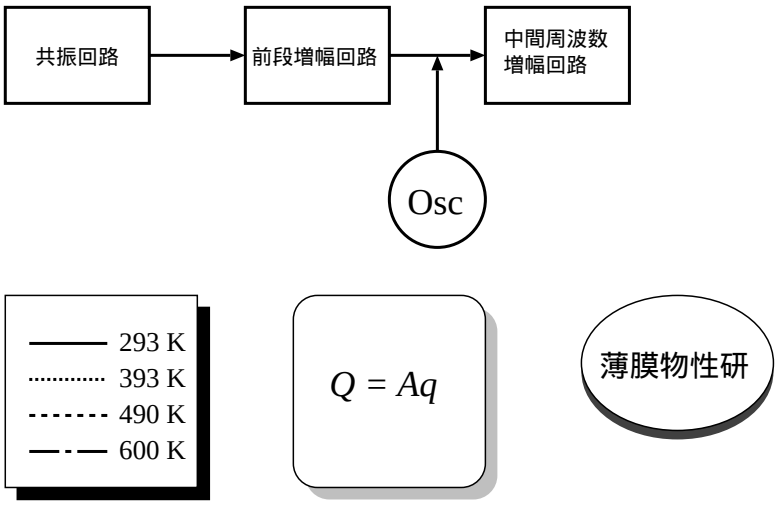
四角や円 (楕円) はメニューでも上位にあることからも判る通り，非常に多用される要素です。例えば，開いた箱を描くには，直観では折れ線ですが，四角を 2 つ (1 つは辺を消すための見えない四角) 組み合わせた方が後で修正が楽という場合があります。

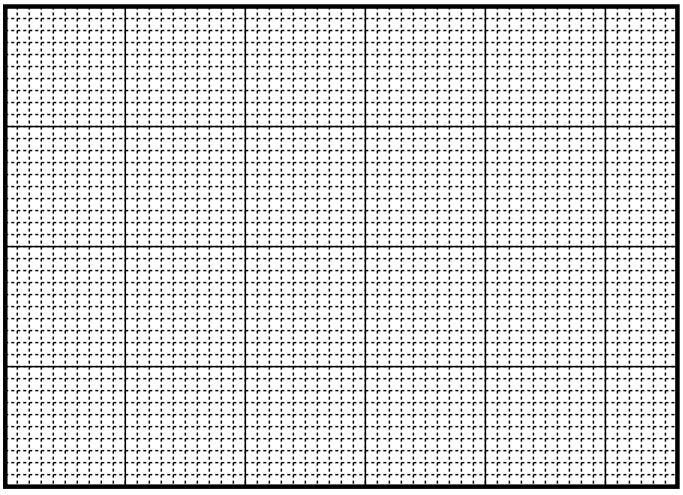
2.2.1　単純な組み合わせ

例題 2

四角や円を組み合わせてブロックダイアグラム・影付きのカード・グラフ用紙 *etc.* を書いてみなさい。
複写 : Ctrl – D を旨く活 用しなさい。

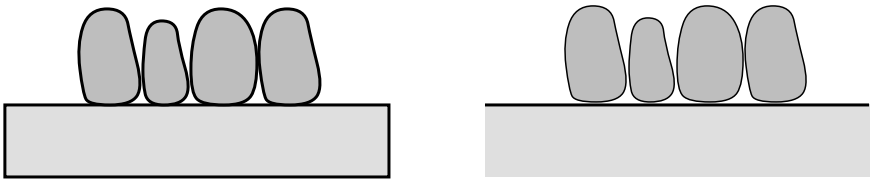
四角・円の実行例





2.2.2　Fill–White の応用

薄膜を作製する基板を描く場合を考えましょう。これは，四角でいいように思えますが，下の 2 つの図を比較してください。

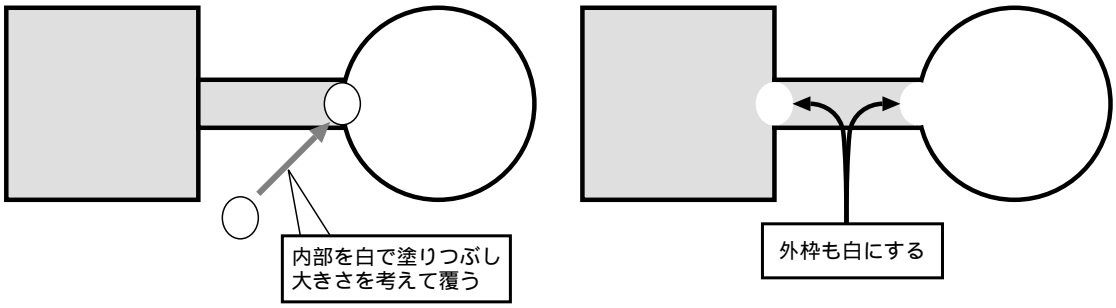


右側は基板内部を表すための外枠が透明で内部を塗りつぶした四角と表面を表わすための直線との組み合わせとなっています。外枠は黒という概念を捨てると応用がみえてきます。特に，外枠も内部も白のものは，下の図形を覆い隠すことができます。

例題 3

2 つの真空槽を管で接続した図を描くとしましょう。真空槽を四角や円で描くと管との接続部分が閉じられてしまいます。この部分に穴をあけてください。全てを折れ線で描くといった暴挙はいけません。解決方法は塗りつぶしです。

解決策



位置合わせや大きさの調整は最小グリッドの状態 **Grid Snap Off**（Ctrl–Alt–G によりトグルで変わる）で行ないましょう，それ以下の微妙な大きさは線の太さで調節するといいでしょう。

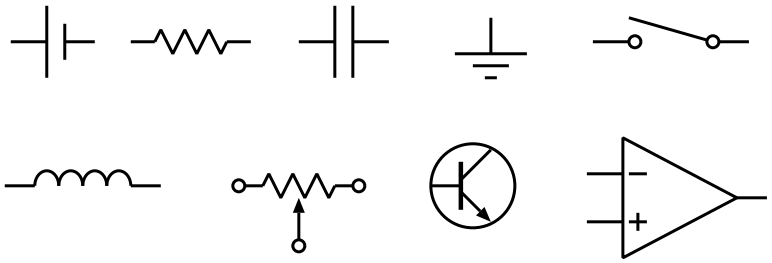
2.3　折れ線で描けるもの

折れ線は基本中の基本要素です。四角も折れ線で描くことが可能ですし，究極には曲線も折れ線で近似されます (フリーハンドはそうなっている)。しかし，四角は四角に任せて，折れ線ならではの簡単な図形を描きましょう。

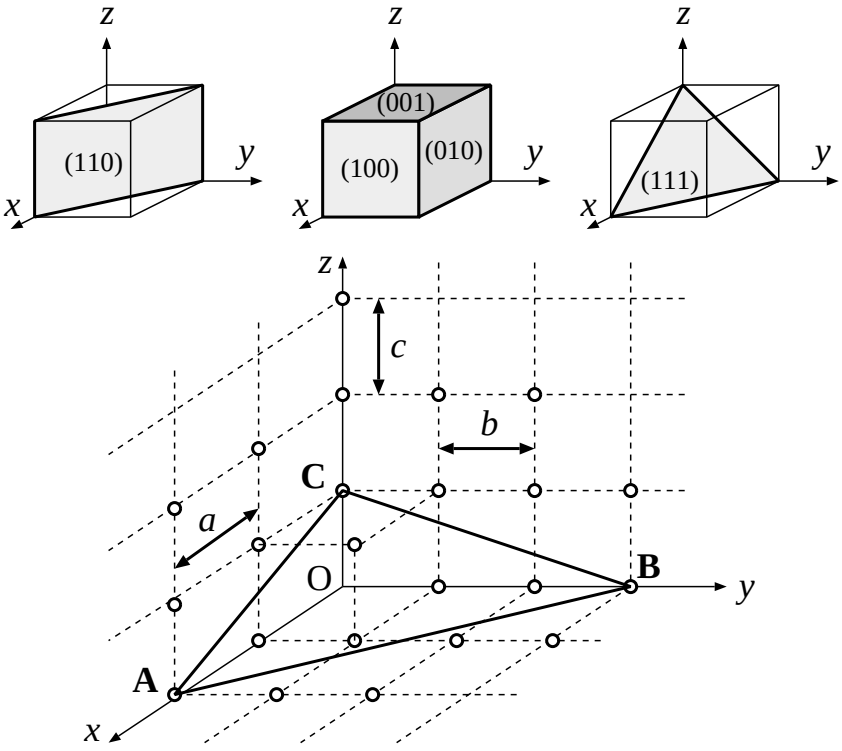
例題 4

折れ線を使って電子回路の部品記号を描きなさい。 電池，抵抗，コンデンサー，アー ス，スイッチ，コイル (インダクタンス)，可変抵抗，トランジス ター，オペアンプ $etc.$ 。 円と組み合わせなければならないものもありますが，比 較的簡単に描けるはずです。

折れ線の実行例



練習 1　ミラー指数の概念を表す次の図を描いてみなさい。



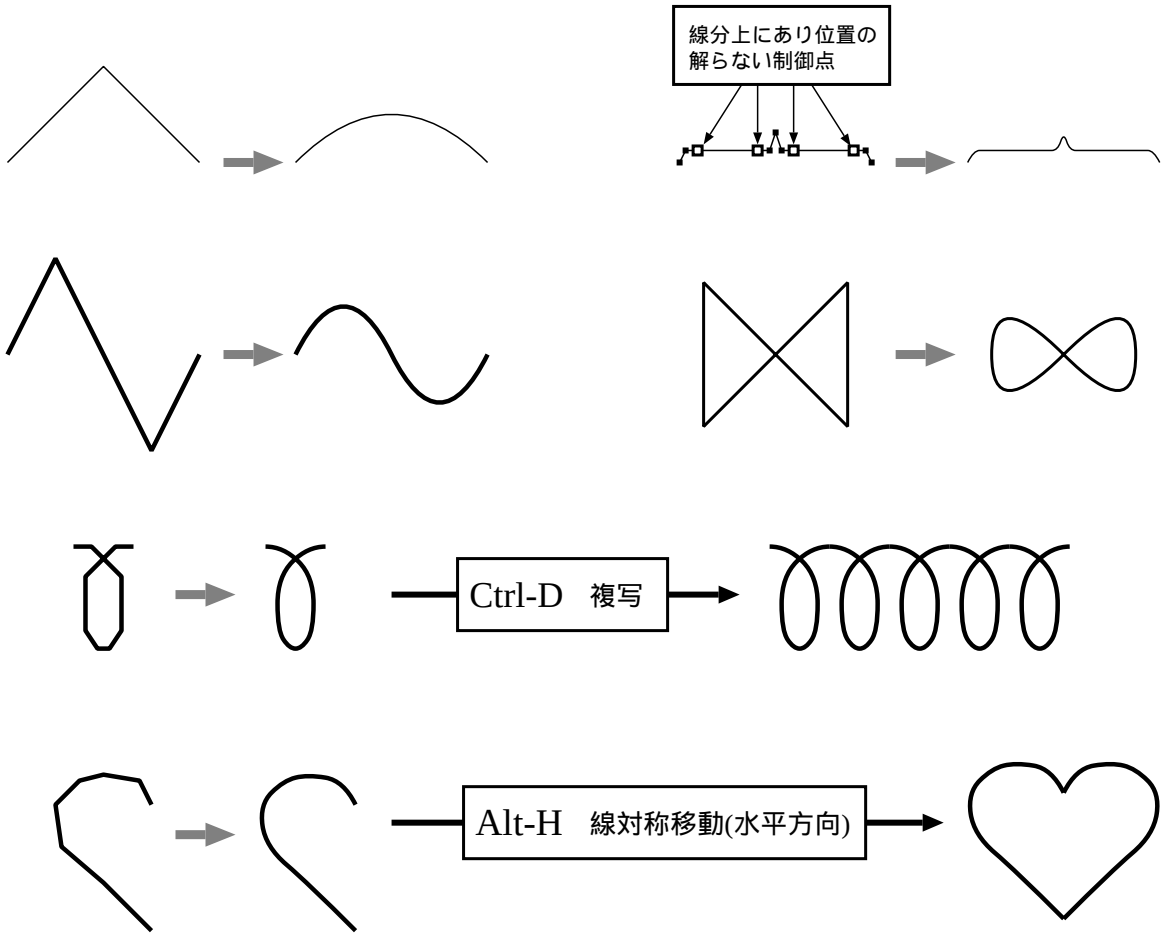
2.4　ベジエ曲線の応用

ばねや正弦波等の曲線をフリーハンドで書くのは，マウスでは非常に難しいです。フリーハンドでは短い線分 (直線) で近似することになり，見苦しい図が出来上がります。多数の短い線分を描くのではなく，代表する数点を制御してその間は計算によりある種の曲線で補間して全体を構成する方法を練習しましょう。直線上にない 3 点を滑らかに繋ぐ (補間する) 曲線には，2 種類用意されています。3 点を通過するように 3 次曲線で補間するスプライン (**spline**) と，端の 2 点は通過するが，中間へ向けての接線が滑らかになるように補間する (当然中間の点は通過しない) ベジエ (**bezier**) です。一見スプラインの方が使い易そうですが，実はスプラインは制御点を無理に通過させる宿命のため滑らかでなくなる場合が多々生じます。私は，ベジエ曲線になれることを勧めます。

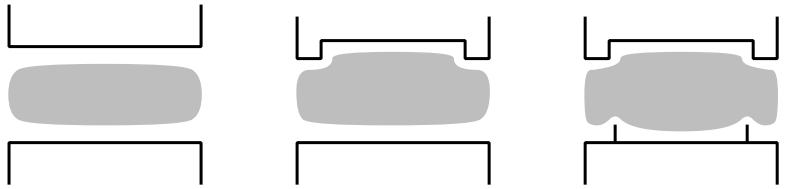
例題 5

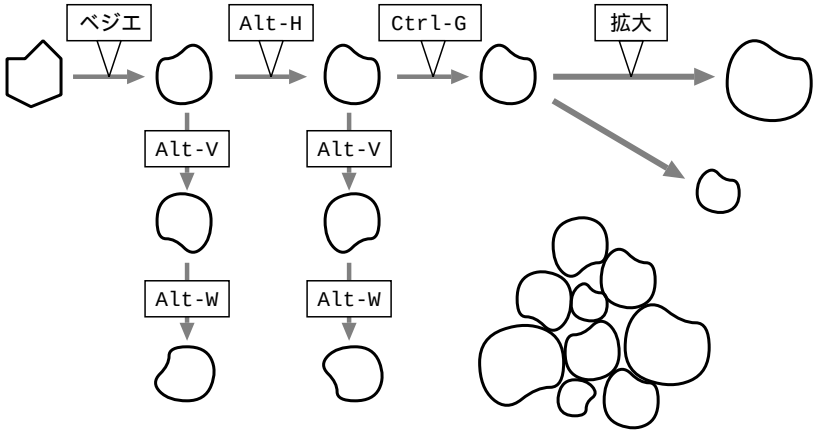
長い小括弧や中括弧，正弦波，ばね，ハート型 *etc.* をベジエ曲線を使って描いてみなさい。複写や線対称移動を旨く使うこと。また，曲率半径を小さくしたい場合 (鋭い角が欲しい場合) には，制御点を近づけると良い。

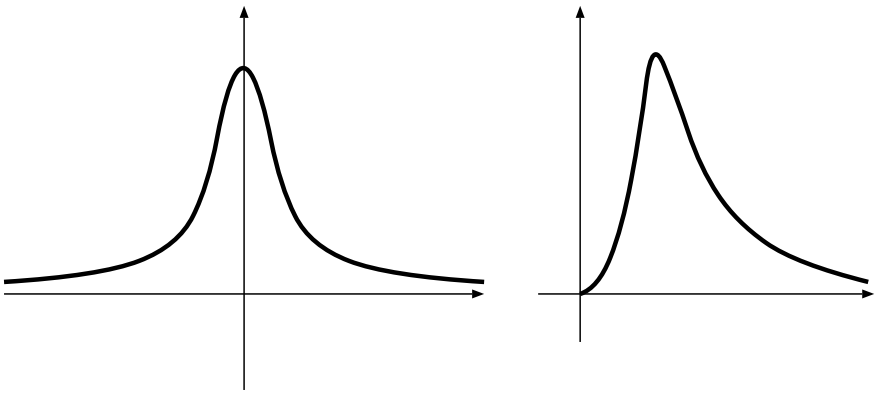
ベジエ曲線の実行例



練習 2　ベジエ曲線はとても便利です。スパッタリング中のプラズマの様子であるとか，薄膜の表面形態の模式とか，ガウス曲線とか色々と描いてみましょう。





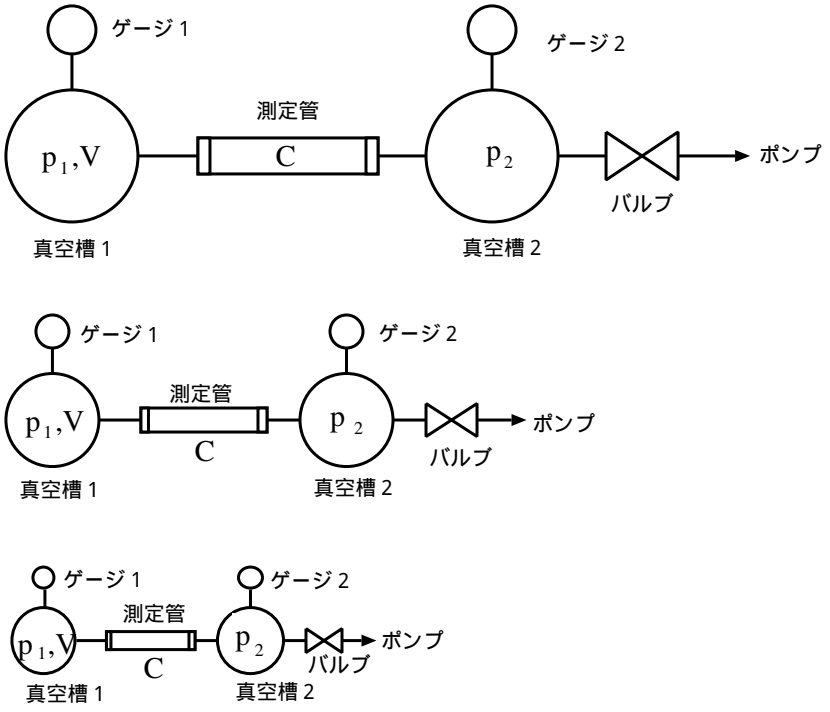


2.5　変形

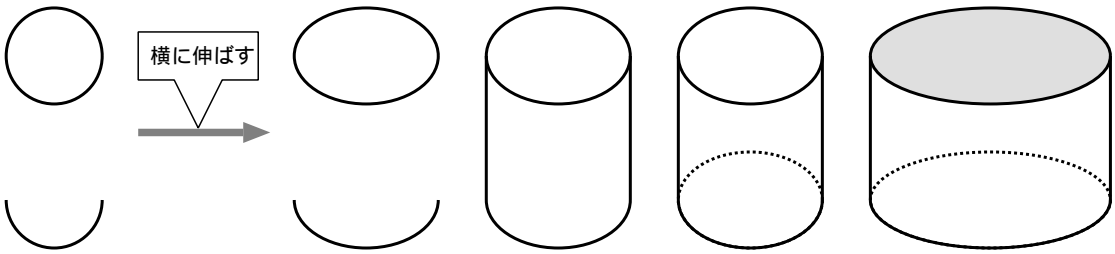
描いた図形はいつでも変形 (拡大・縮小・90 度回転) が可能です。図形を選択すると制御点が現れますから，制御点をマウスの左ボタンで選択してドラッグし，望みの位置で (ドラッグに連れて図形が変形しますから確認して) リリースします。変形はグループ化された図形に対してはグループ全体に作用します。ただし，文字サイズだけは (位置は動く) 変更になりませんから，後で調整が必要でしょう。

また，制御点自身に過不足が生ずる場合があります。例えば，ベジエ曲線を使っていてどうしても望みの図形にならないことがあります，そんな時は制御点の数自身を変更してみましょう。Edit メニューの DeletePonit (Ctrl – Alt – D)，AddPoint (Ctrl – Alt – A) を選択し，制御点の数を増減します。

変形の実行例：グループ化して複写・縮小・文字位置の修正



練習 3　円筒を斜めから見た図を描いてみなさい。問題は半分の楕円をどう作るかですが…，



半円を変形すればよいことに気づきましたか？

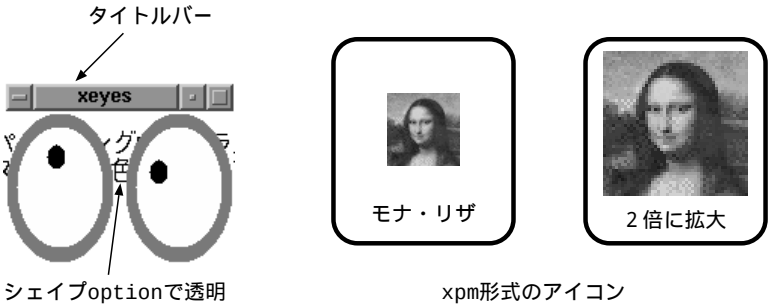
2.6　xbm,xpm,eps 画像の読み込み

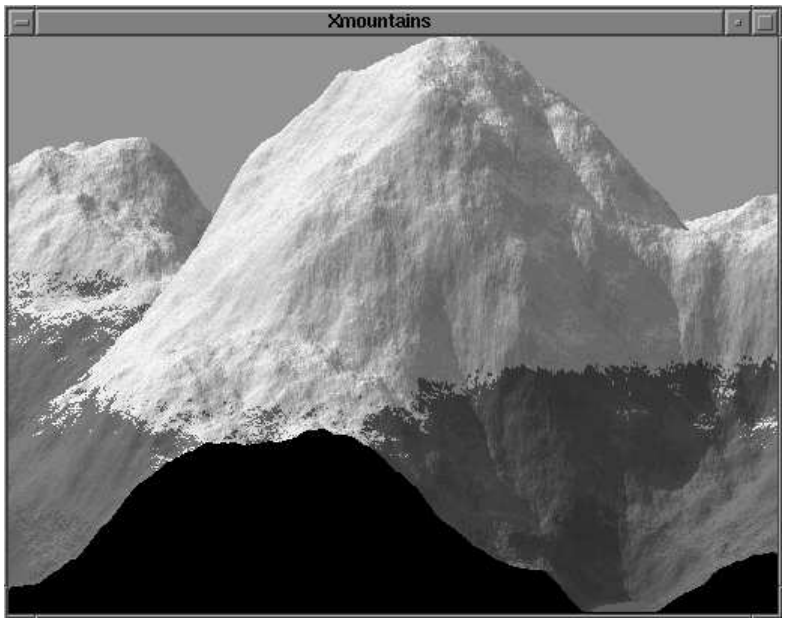
eps 形式を読み込むと外枠 (大きさ) しか表示されませんが，xbm,xpm 形式では内容が表示されます。したがって，画像の部分を矢印で指して説明を付け加える等の場合には，xbm,xpm 形式の画像が扱い易いでしょう。

例題 6

File メニューから **ImportXPixmap** を選択し，xpm 画像を読み込んでみなさい。ファイル選択ダイアログで，/usr/include/X11/pixmaps にディレクトリを移すといいでしょう。そこから任意の *.xpm を選びなさい。
このコマンドの呼出しショートカットキーは Ctrl – Alt – ; です。

xpm 画像を取り込んで説明を付け加えた例





xmountains の実行画面を xwpick で画面ダンプしたもの

2.6.1　xpm 形式で画面のダンプを取る方法 – xgrab –

xpm 形式の画面ダンプを取るツールとしては，**xgrab/xgrabsc** が使えます。xgrab は xgrabsc の GUI メニューとなっていて，option を覚えなくてもよいので使い易いでしょう。3 つの項目を選択し，ファイル名を入力します。*Input Option* には Click on Window ，*Output Format* には Bm/XPM3 ，*Output Options* には To File: を選択して出ファイル名を入力しましょう。**OK** をクリックすると，メニューが消えてカーソルが変化しますから画面ダンプしたい Window 上でクリックします。ダンプした後，再びメニューが現れますから，**Dismiss** で終了します。

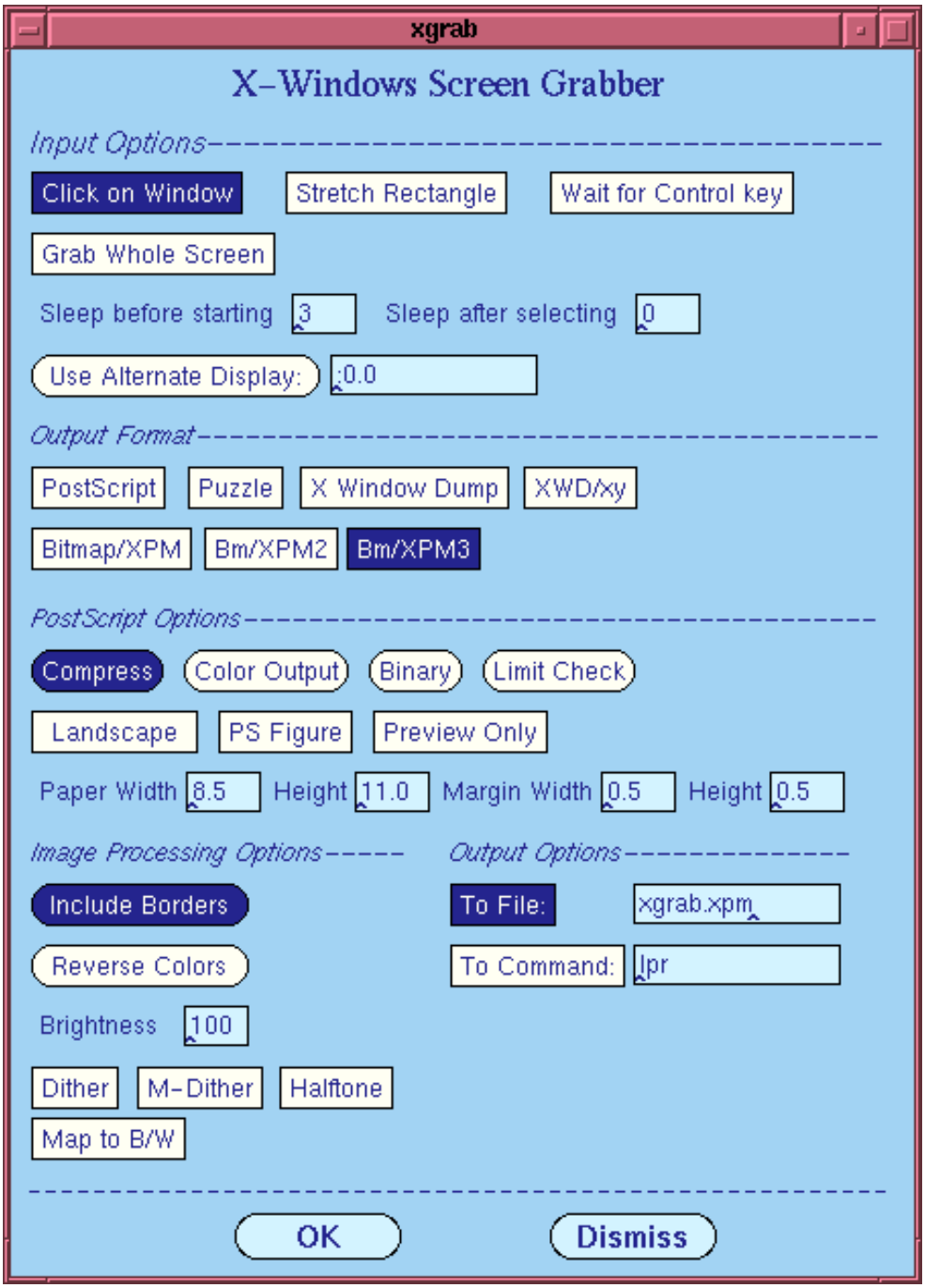


図 12　xgrab の実行画面

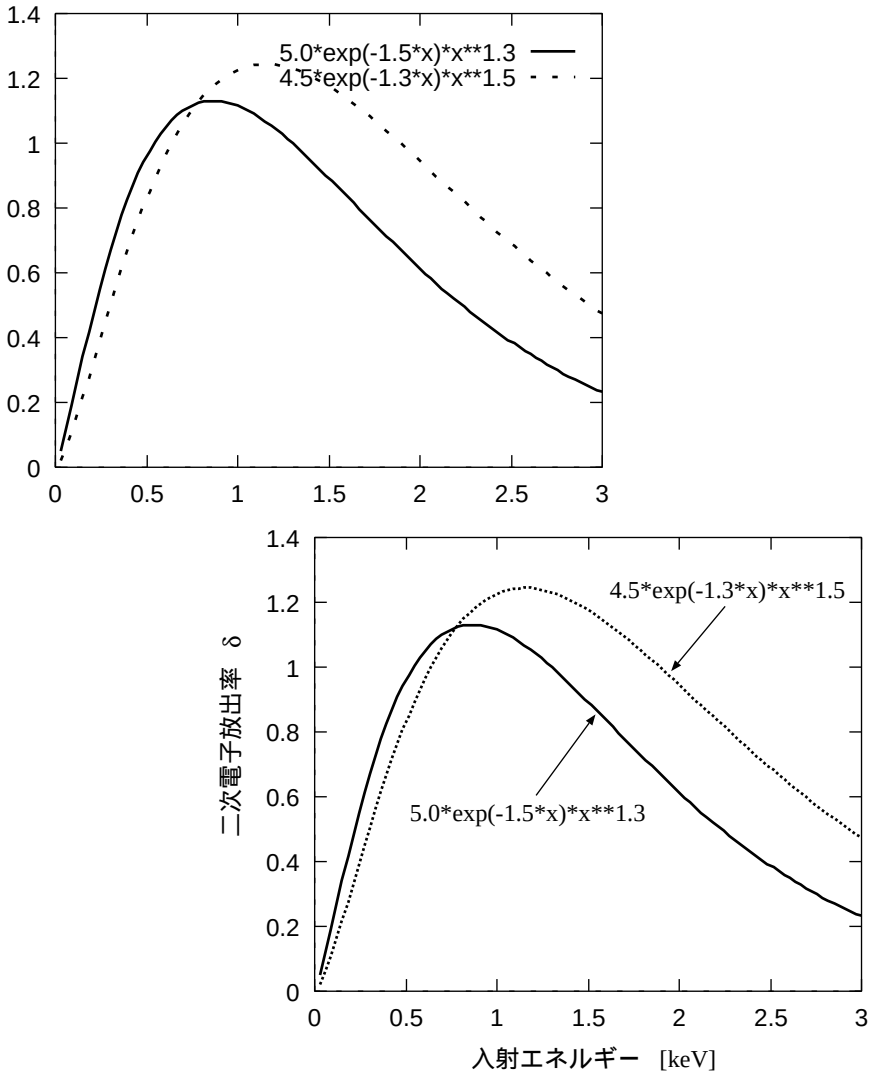
2.7　GNUPLOT からの Tgif-obj 形式出力

例題 7

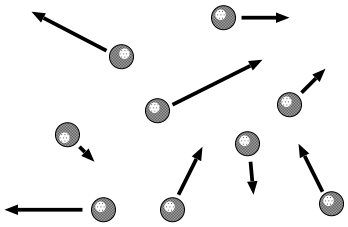
以下の GNUPLOT のスクリプトを実行して，'dionne.obj' を作り Tgif で編集してみなさい。GNUPLOT からの出力ではなぜか黄色で見づらいので，Ctrl–A により全図形を選択した後，色を黒に変換しなさい。

```
set output 'dionne.obj'
set term tgif
set size 0.7,0.5
set xrange[0:3]
plot 5.0*exp(-1.5*x)*x**1.3, 4.5*exp(-1.3*x)*x**1.5
```

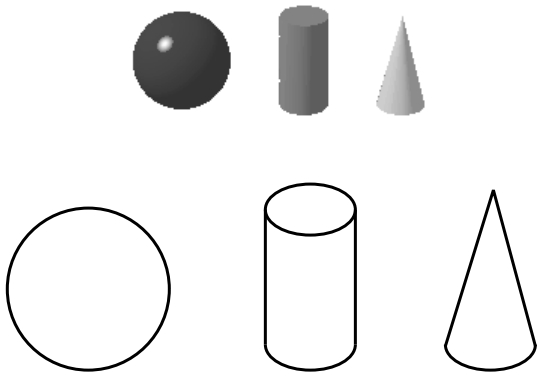
GNUPLOT からの出力を編集



2.8　写実的に見せる工夫



Tgif は 2D のドローイングツールですから，3D あるいは写実的・立体的に描くための機能は持っていません。どうしても写実的な表現をしたい場合には，影を付けるなどの工夫をしてみましょう。右図などはその例です。しかし，球以外の図形に関してはどうしたらいいのでしょうか。例えば，下図の図形に影を付けて，立体的にしてみてください。グラデーションがないと，ドロー系のツールで影に濃淡をつけるのは，非常に大変ですね。こういう場合はむしろペイント系のツール (**Xpaint** や **GIMP**) の方がずっと描き易い筈です。Tgif ではその中で描くのは諦めて，ペイント系のツールや POV-Ray で描いた立体的な画像を読み込んで活用しましょう。

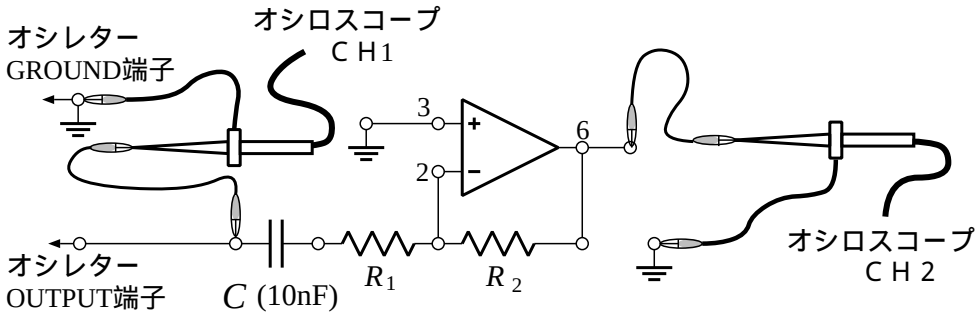


例えば，結晶構造の模形図などは POV-Ray で簡単に描けますから，それを読み込んで，Tgif で元素名や格子定数などを描き加えれば，見映えのよい図ができあがります。

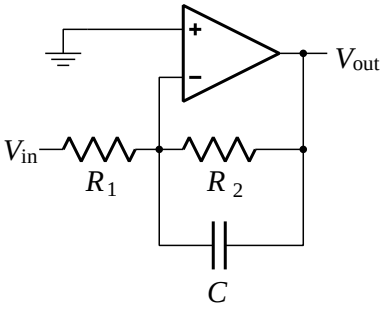
3　図面集

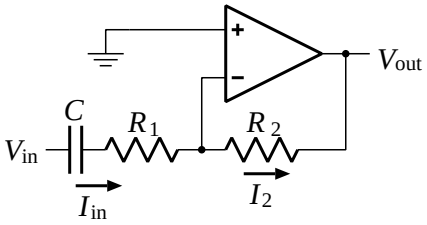
ここ数年，物理関連の実験用テキストを TEX+Tgif で作成してます。その中から面白そうなものを選んでみました。

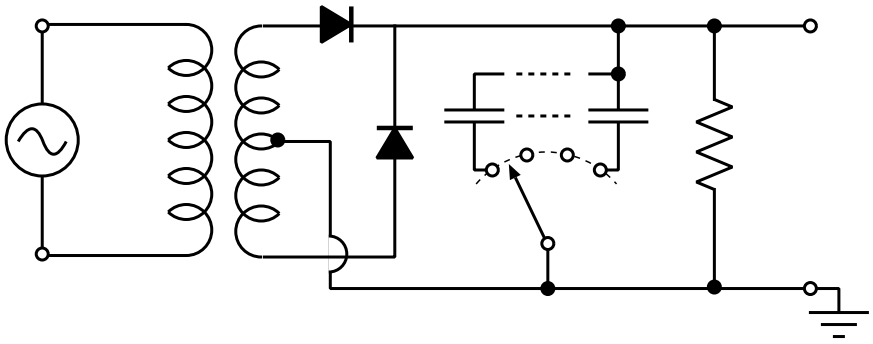
練習 4　次の図を描いてみなさい。実体配線の部分はベジエ曲線で描きましょう。ミノ虫クリップがなかなか面倒です。簡単に描くにはどうしたらいいでしょうか？



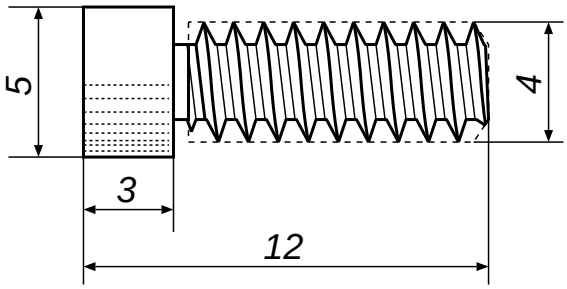
練習 5　次の回路図を描いてみなさい。コンデンサーや抵抗や OP アンプの記号を予め作成しておき，ライブラリとして使いましょう。



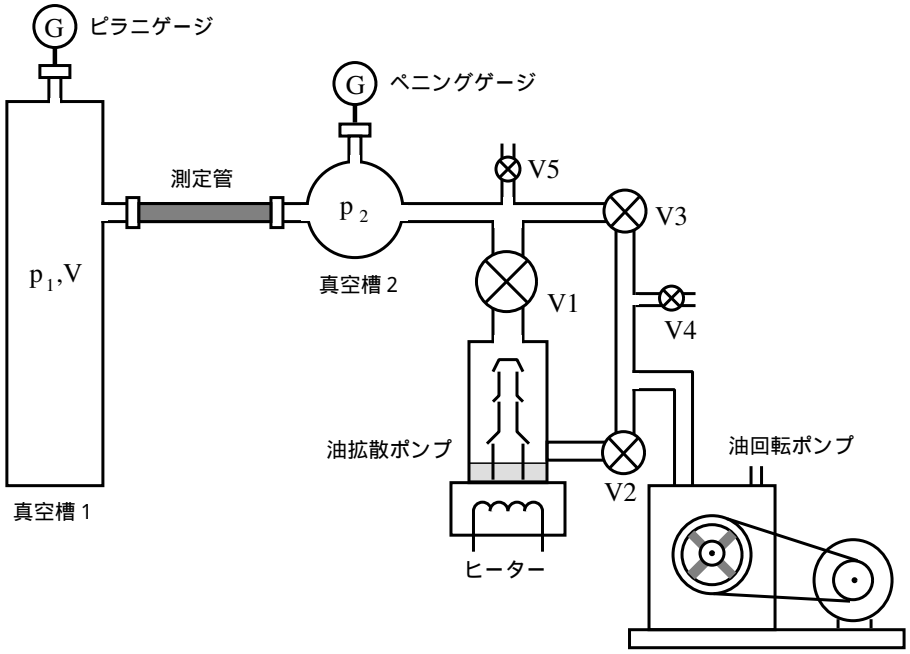


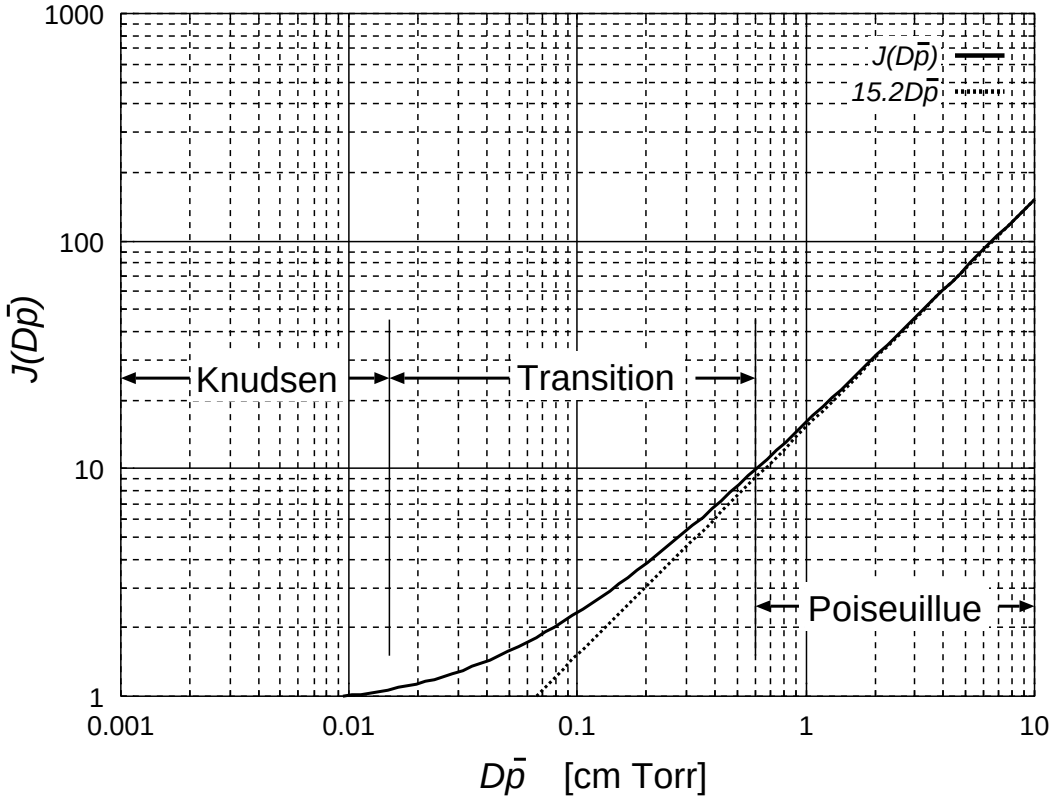


練習 6　次の図を描いてみなさい。これはかなり面倒です，特にネジの部分の立体感を出すためにベジエ曲線を使っています。これを直線で簡潔に表現してみてもいいでしょう。



練習 7　物質工学基礎実験の『P3. 真空技術』から抜粋しました。コンダクタンスの図は GNUPLOT を併用します。





練習 8　tgif で図面を作成し、LaTeX に組み込んでください。S 科『コンピュータ基礎および演習 II』で出題しました。

出力結果：EPS 図面の貼り付け

長さ l の片持ち梁の自由端に集中荷重 F を加えた場合の棒の位置 x と変位 yが満たす微分方程式は

$$\frac{d^2y}{dx^2} = -\frac{1}{R} = \frac{(l-x)F}{EI_z} \tag{1}$$

である、これを x について解き (積分する)、境界条件を考慮すると、棒の自由端の降下量 δ は

$$\delta = -\frac{F}{6EI_z}(3lx^2 - x^3)\Big|_{x=l} = -\frac{Fl^3}{3EI_z} \tag{2}$$

と求められる。

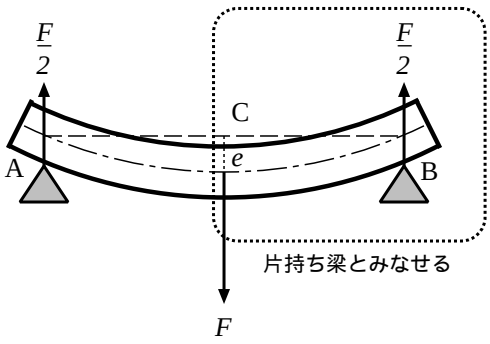


図 13　両端支持棒の中央集中荷重によるたわみ

ところで図 6 に示すように、実験では棒の両端を支えて棒の中央に集中荷重しており、中央から半分の部分は、長さ $\frac{l}{2}$, 荷重 $\frac{F}{2}$ の片持ち梁とみなすことができる。よって、式 (2) にこれらの値を代入して、次の結果を得る。

$$e = \frac{Fl^3}{48EI_z} \tag{3}$$

棒は円弧で描くといいでしょう。棒の外形は太くして、補助線や中央線は細くするのが普通です。日本語の入力時には **kinput2 -canna** で kinput2 が起動していること次のコマンドで確認してください。

```
ps -aux | grep kinput2
```

復習しますが、EPS 図形を貼るには以下のようにします。

EPS 図形の貼り込み方法

- `\documentstyle[a4j,epsbox]{jarticle}` 等とスタイルオプションで epsbox を指定する。
- 図を表示したい場所で

  \psbox[サイズ]{*filename.eps*}

  を行う。サイズには width=5cm, height=100pt, scale=0.7 等の指定が可能。