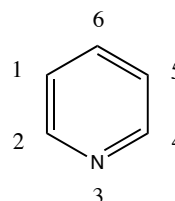


分子構造データ

molフォーマットと言うファイル形式ではたとえば右図のようなピリジンの分子構造を以下のように記述する（水素原子は省略されている）



pyridine.mol

```
-----
 6  6  0  0  0  0  0  0  0  0  0999 V2000
  -0.6135   0.5570   0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
  -0.6154  -0.7920   0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0  0  0
   0.5496  -1.4645   0.0000 N   0  0  0  0  0  0  0  0  0  0  0  0  0  0
   1.7175  -0.7909   0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0  0  0
   1.7128   0.5628   0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0  0  0
   0.5467   1.2298   0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0  0  0
  1  2  4  0  0  0  0
  3  4  4  0  0  0  0
  4  5  4  0  0  0  0
  2  3  4  0  0  0  0
  5  6  4  0  0  0  0
  6  1  4  0  0  0  0
-----
```

ピリジンはC₅NH₅で、右側の図の様な六角形の分子である。このデータでは炭素についた5個の水素原子が省略されている。

molファイル一行目の最初の6は原子数、二つ目の6は結合数を表す。3カラム目以降のエントリーは無視して構わない

その下の6行（原子情報行）が各原子についての情報で、最初の3つの実数は x , y , z それぞれの座標、次の C, N などの文字が 炭素(C) 窒素(N) 酸素(O) 水素(H)などの元素の種類を表す。4つ目以降のエントリーは無視して構わない。（ピリジンは真っ平らなので、 z の値が全て0.0になっている）

続く6行（結合情報行）は結合表で、その一行目の1 2 4では、1番目と2番目の原子がタイプ4の結合で結ばれていること、二行目以降、3-4, 4-5, 2-3, 5-6, 6-1 の原子間の結合が存在することを表している。三番目の数字（ここでは全て4）は結合次数(bond order)であり一重結合が1、二重結合が2、三重結合が3で、4は芳香族の共役結合を表している。今回は結合字数は特に必要としないので無視してもかまわない。

このようなファイルを読み込んで、データを構造体の配列に格納し、PostScriptファイルフォーマットで、分子構造を表示するプログラムを作成する。

原子情報 (atom_info) の構造体は読み込みデータとして、 x, y, z のそれぞれの座標を実数型として持つ。

ポストスクリプトファイルの出力

前回の演習で作成したプログラムを編集して、分子構造を簡易的にポストスクリプトフォーマットで表示するプログラムを作成する。

ここでは単純に(1)結合表にある原子を直線で結ぶ。(2)原子座標の、 x, y 軸のみの値を用いて、ポストスクリプト描画の平面上に投影する。という考え方で 前々回ダウンロードした .mol というサフィックスを持つmolファイルの内容を表示する。準備したmolfileの x, y 座標値は最大で $\pm 11\text{\AA}$ 程度であり、それぞれの座標値を20.0倍して245.0を加えれば、A4画面内に収まる描画とすることができる。

molファイルから読み込む必要があるのは、上記のピリジンの例で太字とした**原子数・結合数**、

各原子に付いて必要な情報は**x,y,z座標値**、各結合について必要な情報は**接続関係にある原子対の番号のリスト**である。

作成するプログラム名をmol drawとし、入力ファイルをmol1.molとすると、実行時に

```
mol draw mol1.mol
```

とすれば**mol1.mol.ps**というファイル名でポストスクリプトファイルが出力される様にする
argv[1]を入力ファイル名とし、

```
FILE *infile;  
infile = fopen(argv[1], "r");
```

と開き、出力ファイルは

```
char *psfile_name[132];  
sprintf(psfile_name, "%s.ps", argv[1]);
```

とoutfile_nameを出力ファイル名として作り、出力ファイルもfopenする

```
FILE *psfile;  
psfile = fopen(psfile_name, "w");
```

構造体定義などを含むヘッダファイル (mol draw.h)、メイン関数を含むソースファイル

(mol draw.c)、コンパイル・リンクに用いるmakefileの3点のファイル(関数を分割した場合は4点以上でも可)を012nakamura13mol draw1と自分の名前をつけたディレクトリに入れ、tar, gz してできた 012nakamura13mol draw1.tar.gz (自分の番号と氏名で置き換え)を添付ファイルとし、中村_13基本2(自分の氏名で置き換え)、をタイトルとしてメールで提出する。ポストスクリプトファイルの最初の3行と最後の2行は常に同じであるため、コード中にそのまま記述して出力する。

===== 最初の3行 =====

```
%!PS-Adobe-3.0  
<< /PageSize [592 842] >> setpagedevice  
newpath
```

=====

===== 最後の2行 =====

```
stroke  
showpage
```

=====

出力する文字列が %記号を含むために、

```
fprintf(psfile, "%!PS-Adobe-3.0\n");
```

としようとする、printfではダブルクォートの中の%を出力指示子と解釈するためにうまく行かない。
一つの解決方法としては、出力文字列のバッファを作って出力する行を記し、そのバッファをfprintfすることが考えられる。上記の1行は

```
char buff[512];  
strcpy(buff, "%!PS-Adobe-3.0\n");  
fprintf(psfile, buff);
```

とすれば最初の1行を書き込むことができる。

buffは一度定義しておけば再利用できるので、psfileへの出力は原則として 出力文字バッファを介することを推奨する。

```

===== moldraw.c =====
.....
.....
.....
char buff[512];
FILE *psfile;
// 出力ファイル開く //

.....
.....
strcpy(buff, "%!PS-Adobe-3.0");
fprintf(psfile, "%s\n", buff);
strcpy(buff, "<< /PageSize [592 842] >> setpagedevice");
fprintf(psfile, "%s\n", buff);
strcpy(buff, "newpath");
fprintf(psfile, "%s\n", buff);
.....
.....
=====

```

の様にして最初の3行を出力できる。

分子構造の描画のために、ポストスクリプト出力の、前3行と後2行の間に、結合の数だけ直線を描くループを挿入する

```

===== moldraw.c =====
.....
.....
float fromx, fromy;
float tox, toy;
.....
.....

// ポストスクリプト最初の3行を出力

for(i=0; i<n_bond; i++)
{
// 描く直線の両端の座標値 (fromx, fromy, tox, toy) を計算する

spritnf(buff, "%f %f moveto", fromx, fromy);
fprintf(output_file, "%s\n", buff);
spritnf(buff, "%f %f lineto", tox, toy);
fprintf(output_file, "%s\n", buff);
}

// ポストスクリプト最後の2行を出力
.....
.....
=====

```

ループのあと、最初の3行と同様に、最後の2行を出力する

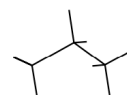
座標値としては、それぞれの座標値を約20倍し、PSの座標系の中央へスライドするために、x座標は +297.5、y座標は +421.0 しておけば、とりあえず表示はできるかと思います。

作成した(例えば) mol1.ps というポストスクリプトファイルについてはevinceというコマンドを用いて
evince mol1.ps

とするとpdfビューワーが起動して表示を確認することができる

propane.mol の出力例

----->



応用課題

12_2. A4の紙面の片隅に寄ったり、小さく表示されたりせずに、紙面中央に出来るだけ大きく分子の構造を表示できる様に改良する。すべての原子の質量を同じとした場合の重心点を中心とすることで画面中央に出来るだけ大きく表示ができる様にする。また入力ファイルの x, y 座標の最小値、最大値をそれぞれ求め、A4の枠内に描画が収まる様に全体のスケールを計算する。`moldraw2.c`, `mol.h`, `makefile`を含む `tar.gz` ファイル名は `012nakamura13moldraw2.tar.gz` メールタイトルは 中村13_応用2

12_3. 1を応用して、 xy 平面、 xz 平面、 yz 平面への投影図を並べた三面図を描画する
作成したcのソースコード`moldraw3.c` ヘッダファイル `moldraw.h` メイクファイル `makefile` (作成した関数のソースファイルがあればそれも) を収めて `tar, gz` したファイル
(`012nakamura13moldraw3.tar.gz`)をメールに添付しタイトルを中村13応用3。

その他、原子の位置に元素記号を表記したり、結合次数に応じて2重結合を見やすく2本線で表示するなどの工夫をしていただいたものができたら`012nakamura13moldraw4.tar.gz` 中村13応用4として送ってください