

テキストベースのゲノム配列ブラウザ gnm

gnm は Unix 上のテキストエディタである vi/vim に順じた操作感で、ゲノム配列をブラウズ・検索・染色体上の塩基位置を移動しながら確認・観測するためのツールです。

ダウンロード・コンパイル・インストール

Unix/Linux 端末上でダウンロードした gnm1.1.tar.gz を展開し、ディレクトリ gnm1.1 の中に入ります

```
$ tar xzvf gnm1.1.tar.gz
```

```
$ cd gnm1.1
```

make コマンドによりソースコードをコンパイルし実行形式ファイル gnm を得ます。

```
$ make
```

コンパイルには ncurses ライブラリが必要です。必要であれば Ubuntu 上では以下のようにしてインストールします。

```
$ sudo apt-get update
```

```
$ sudo apt-get install libncursesw5-dev
```

コンパイルあるいはダウンロードした実行形式の gnm をコマンドパスの通った場所にコピーしておきます。例えば

```
$ sudo cp gnm /usr/local/bin
```

以上で、fasta ファイルのある場所から gnm コマンドが起動できるようになります

起動と終了および画面表示

入力ファイルはシングル Fasta あるいはマルチ Fasta 形式のゲノム配列となっていて、入力ファイル input.fasta に対して以下のように起動します。

```
% gnm input.fasta
```

最上段の黒地に白文字の行の左側は、当該のファイルが 25 本の配列（染色体）からなるマルチファスタファイルで、そのうち 1 本目を表示中であることを表します。右側には Fasta ファイル中の当該配列のタイトル行が示されます。

画面中央の白地に黒文字の領域に塩基配列が表示されカーソル位置は黒地に白文字の反転表示となっています。

画面下から 3 行目の黒地に白文字の行の数値は、左から、カーソル位置の塩基位置、画面左上から右下の表示範囲（1-1672）、表示中の染色体の長さ（248387328）ファイル名（chm13v2.0.fasta）となっています。

単純に q あるいは :q を入力すると、gnm は終了します。



基本的なカーソル移動

カーソルを上下左右に動かすには、vi/vim と同様に 上:k 下:j 左:h 右:l キーを使います。vi/vim に不慣れであれば矢印のカーソルキーでも同様です。

整数値をそれぞれの向きの前に記述することで、移動量を指定できます。例えば 3 2 塩基上流（左）に移動したければ 32h あるいは 32← と入力します。数値を入力している間、指定する数値はコマンド入力行に表示され、kjhl あるいは矢印キーを入力した時点で数値表示は消えます。負の値は指定できません。

指定した配列（染色体）番号・塩基位置への移動

カーソルを指定した配列（染色体）番号の指定した塩基位置に移動する場合、二つの数値の間に : (コロン) を挟んで入力し、Return キーを押します。例えば 5 番目の配列（染色体）の 100 塩基目に移動したい場合

5:100 と入力した後、Return キーを押します。5:100 の部分はコマンド入力行に表示され Return の後消えます。

5: だけで Return キーを押すと 5 番目の配列の先頭（1 塩基目）に移動します。

:1000 だけで Return キーを押すと表示中の配列番号の 1000 番目の塩基位置に移動します。

文字列検索

vi/vim と同様に、/ (スラッシュ) の後に探索したい塩基配列を記述し Return キーを押します。ATGC なら /ATGC で Return キーを押すと (下流への) 次のヒット位置にカーソルが移動し一致した文字列が薄いブルーで、相補逆鎖での一致 (GCAT)がある場合には薄いピンクで表示されます。カーソル位置より上流へ逆方向に検索する場合にも vi/vim と同様に、? (クエスチョンマーク) の後に探索したい文字列を記述して Return キーを押します。ヒット位置が見つかった、あるいは、見つからなかったという結果がメッセージ表示行に表示されます。同じ文字列を続けて下流に向けて探索したい場合には、小文字の n を、上流に向けて続けて探索したい場合には大文字の N を押します。検索を / で始めたか、? で始めたかによらず、n は下流への、N は上流への検索となります。検索する文字列の長さにより、二つの検索モードが自動的に切り替わっています。

- a) 検索する文字列が9文字より短い場合：次に現れるヒットを逐次検索し、見つけ次第カーソルが移動します。
 - b) 検索する文字列が10文字以上の場合：全ての配列中のヒット位置を探索し、次に現れるヒット位置にカーソル移動します。検索は使用する計算機の全てのスレッドを用いて並列に行われるため、通常それほど時間はかかりません (数秒以内)。またメッセージ表示行に順方向・相補逆鎖・双方向のヒット位置の総数が表示されます。
- /あるいは?と検索文字列の間に1~9の整数値を挟むことで、ミスマッチを許容した検索が可能です。整数値は許容する最大のミスマッチ数で、ギャップを含まないハミング距離となります (ギャップを許容する曖昧検索 (レーベンシュタイン距離) については今後のバージョンで検討します)。例えば、/1ATGCATGC として Return すると、ATGCATGC の1文字が置き換わった文字列 (**T**ATGCATGC や ATGCAT**C**C など) にヒットし、不一致文字は通常のヒット位置より濃い色での反転表示となります。この場合にも検索文字列の長さにより、上記 a) と b) の検索モードの切り替えがあります。許容するミスマッチ数は最大9で b) で多くのミスマッチを許容すると、ヒット位置が膨大になりメモリが不足する場合があるので注意が必要です。

検索が当該配列の最下流に到達すると、次の配列の1塩基目からの検索となります。最後の配列の最後の塩基に到達すると、1つ目の配列の1塩基目からの検索となります。逆方向も同様です。

図では、塩基配列に大文字と小文字が混在しており、小文字は一般に低複雑性領域となっています。現時点では、大文字の文字列で検索すると、大文字・小文字の両方にヒットし、小文字の文字列で検索すると小文字の配列にのみヒットするようになっています。

それまでの検索結果のリセット (ブルーやピンクの表示の解除) には rs (reset search) または rf (reset find) とします。

マーク

vi/vim と同様に、あとで参照するために、カーソル位置をマークすることができます。マークしたいカーソル位置で、m に続けて a-z/A-Z のアルファベットを入力することで、2文字目のアルファベットでカーソル位置がマークされます。マークされた位置は反転表示となります。カーソル移動後、マークした位置に戻るためには、' (シングルクォート) のに続けて先ほどマークしたアルファベットを入力します。例えば、あるカーソル位置で ma とするとその位置が a という記号でマークされます。カーソルを移動した後に、'a (シングルクォートに続けて a) とすると、a でマークした位置にカーソルが移動します。

それまでのマーク位置 (反転表示も) のリセットには rm (reset mark) とします。

領域選択 ビジュアルモード

マークと類似の機能として、vi/vim のビジュアルモードの様に、配列上の領域を選択して記憶することができます。選択を開始したい塩基位置にカーソルを置いて、v と入力すると、画面の1行目左端に V の反転表示が現れ、ビジュアルモードに入ったことを示します。ここからカーソルを移動していくと最初の位置との間が薄いグレーの背景表示となります。選択を終了したい塩基位置にカーソルを置いたら、マークの場合と同様に、もう一度 v と入力し、a-z/A-Z のいずれかのアルファベットを入力することで当該領域がそのアルファベット記号で登録されます (ビジュアルモードは解除されます)。

例えば、ある塩基位置で v と入力し、10l として右に10塩基移動したのちに、vb とすると最初の塩基位置からの11塩基が b という記号で記憶されます (薄いグレーの背景表示)。

カーソル移動後、` (バッククォート) に続けて b とすると、b で登録したビジュアル領域の先頭にカーソルが移動します。

それまでのビジュアル領域選択 (薄いグレーの背景表示も) のリセットには rv (reset visual) とします。

選択された領域と同じ配列の検索

ビジュアルモードで選択した領域と同じ配列を検索することができます。上記の例である配列領域を b という記号で登録してある場合、/vb として Return すると、b の配列領域をカーソル位置から下流に検索します。通常の文字列検索と同様、2塩基のミスマッチ (ハミング距離) を許容して探索する場合には /v2b 等として Return します。

今後の予定 (To do list)

選択領域のファイルへの書き出し

ギャップを許容する検索

gff ファイルからの遺伝子アノテーションの読み込み (遺伝子名での塩基位置検索 等)

本プログラムの使用によって生じた、いかなる損失、損害についても責任は負いかねます。

プログラムライセンスは GPL に準拠することとします。

バグレポートおよび改善点のご提案は下記までご連絡ください。可能な限り対応します。

公立大学法人 前橋工科大学 工学部 情報生命工学群 情報システムプログラム 中村 建介
knakamura@maebashi-it.ac.jp (@は半角)