

「情報処理」1年文I/IIクラス11-12 #4

久野 靖*

1995.11.6

今回も設定をちょっと手直ししましたので、

```
~kuno/setup mh
~kuno/setup emacs
```

を実行しておいてください。あと、授業でも言いましたがメールを Rmail、mh-e などの方法で読んでいる人は特に MH に乗り換える必要はありません。今日やるニュースの方も mnews 以外の方法でも読むのと投稿さえできれば何でもかまいません。

0 本日の目標

課題の提出状況を見ているとほとんどの人は自分でプログラムを書けているようで、とても結構だと思います(第1回に宣言したように、全く書けなければ単位はあげませんけど)。ただ、整数と実数の使い分けとか整数演算子が難しいようですね。今週も前回の復習から入って、次のことを目標とします。

- 整数と実数の使い分けについて納得する。
- 枝分かれのあるプログラムについて学ぶ。
- 日本語の入力。
- 電子ニュースの読み書き。

*筑波大学大学院経営システム科学専攻

1 復習: 整数と実数の使い分け/*

前回の演習3の問題を例にとって、整数と実数の使い分けについて考えてみる。何回もいうが、掲載しているPAD図やプログラムはあくまでも「例解」だからこの通りでなければ間違いということはない。

a 2つの数 x , y を入力し、 x を y で割った商と余りを出力。

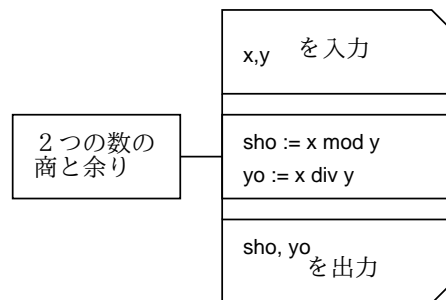


図 1: 商と余りを求める

```
program sam41(input, output);
var x, y, sho, yo: integer;
begin
  write('x> '); readln(x);
  write('y> '); readln(y);
  sho := x div y;
  yo := x mod y;
  writeln('sho = ', sho:5);
  writeln('amari = ', yo:5)
end.
```

この問題は、整数での演算ばかりなので変数はすべて `integer` にすればよい。

ところで、`writeln`の中に書く式:幅がよくわからんという人が多いので再度説明しておく。この値は整数を出力する場合は、出力のための取る幅を意味する。たとえば、上のは幅が5だったが、余りの値が7だとすると

```
amari =      7  ←左側に4文字のあきが取られている
```

のように、出力全体の幅が広くなる(見やすくなる)。たとえばもっと大きな数で余りが「1234」だとすると今度は出力が4桁あるので

amari = 1234 ←左側に 1 文字のあきが取られている

ということ。なお、必要とするよりも狭い幅を取った場合も、必要最小限の幅は取られるのでまあ大丈夫である。

b 2つの数 a、b、c を入力し、その平均を出力。

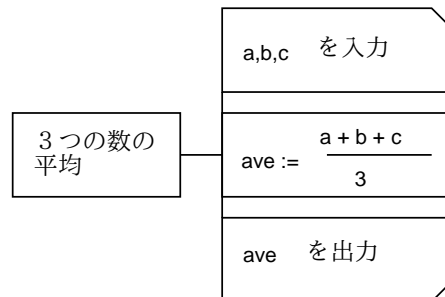


図 2: 3つの数の平均

```
program sam42(input, output);
var a, b, c, ave: real;
begin
  write('a> '); readln(a);
  write('b> '); readln(b);
  write('c> '); readln(c);
  ave := (a + b + c) / 3.0;
  writeln('ave = ', ave:8:2)
end.
```

こんどは、平均は実数 (小数点つきの数) で計算したいので、すべての変数を `real` にしている。平均の計算の「3」も `3.0` としてあるのに注意。実数と整数をまぜないようにしよう (トラブルのもとだから)。

さて、ここでも出力の「値:幅:桁数」がわからないという人がいると思う。こんどは上に加えて「桁数」が加わったが、これは小数点以下何桁出力するかを指定している。たとえば「123.456」というのを出力するときに

指定	出力	
ave:8:3	123.456	←ちゃんと 3 桁出ている。
ave:8:2	123.46	← 2 桁までで 4 捨 5 入

```

ave:8:1      123.5  ← 1桁までで4捨5入
ave:8:0      123   ← 小数点以下4捨5入
ave:8        1.2e+02 ← 指数形式!

```

のようになる。とくに「幅」しか指定しないと指数形式になって読みにくいので注意。

c cm 単位の長さを入力し、XmYcmZmm の形で出力。

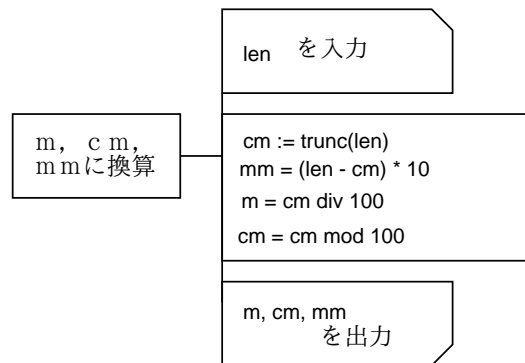


図 3: m/cm/mm 単位の換算

```

program sam43(input, output);
var len, mm: real;
    m, cm: integer;
begin
write('len> '); readln(len);
cm := trunc(len);           ←切り捨てたらセンチになる
mm := (len - cm) * 10.0;    ←センチ以上をなくしてから10
倍
m := cm div 100;           ←センチを100で割ればメー
ル
cm := cm mod 100;         ←その余りが本当のセンチ
writeln('len = ', m:1, 'm ', cm:1, 'cm ', mm:4:2, 'mm')
end.

```

こんどはもっと厄介である。つまり、最初の長さセンチとミリの長さは実数だが、センチとメートルは整数がよい(むずかしい!) なぜかよく考えてみよう。センチやメートルは切捨てたり剰余を取ったりしてきりよくまとめるが、ミリはその残りだから端数があるわけ。

2 判断と分岐/☆

さて、ここまで習ったのだとプログラムは上から順に処理して行って、下まで来ておしまい、ですね？ これでは大したバリエーションはできない。そこで次に、判断と分岐を習っておこう。例えば、次の問題を考える。

例題 数値 (実数) を入力し、その絶対値を出力する。

このような問題では、もらった値が「負かどうか」で処理を分けなければなりませんね？

PAD では、条件による処理の分岐は右側がぎざぎざの箱で表す。上の問題に対する PAD の例を示そう。このように、ぎざぎざの箱の中に条件

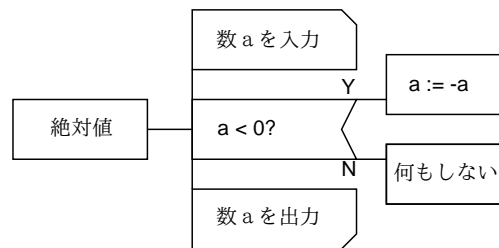


図 4: 「絶対値」の PAD

を書き、その右に条件が成り立った場合 (Y) と成り立たなかった場合 (N) の処理を分けて描く。上の場合には、 a の値が負だったら反転するので、もともと正の場合には何もすることがない。そこで「何もしない」と書いてあるわけだ。ところで、もう 1 つ別の考え方の PAD も示しておこう。これでもいいわけですね。あなたはどちらが好みか？

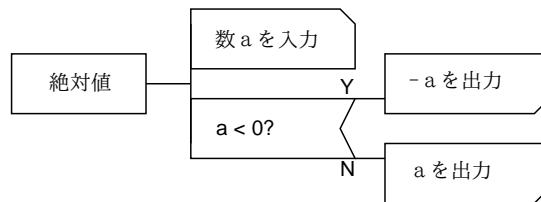


図 5: 「絶対値」の PAD(その 2)

さて、今度はこれらを Pascal にする。そのために、さっき習った Pascal の規則に次のことを追加する。

- 2 項演算子には次のもの (比較演算子) もある。

```
=      --- 等しい
<>    --- 等しくない
>      --- より大きい
>=    --- 以上である
<      --- より小さい
<=    --- 以下である
```

これらはすべて真偽値 (true=「はい」、false=「いいえ」のどちらかの値) を返す。

- 文の種類に次のもの (if 文) を追加する。

```
if 式 then 文
if 式 then 文 else 文
```

「式」は真偽値を表すものでないといけない。その働きは、「式」の値が true であれば then の次の文を実行し、そうでなければ (あれば) else の次の文を実行する。だから、PAD で片方が「何もしない」ような枝わかれは上、それ以外は下の形に対応する。枝の中には「文」が 1 つしか書けないことに注意せよ。複数書きたい場合には下記参照。

- 文の種類に次のもの (複合文) を追加する。

```
begin 文; … ; 文 end
```

つまり、沢山の文でもその全体を begin-end で囲むと 1 つの「文」として扱われるので、if の枝に書くことができる。

では、上の 2 つの PAD を Pascal にしたものを示しておく。

```
program sam3a(input, output);
var a: real;
begin
  write('A> '); readln(a);
  if a < 0 then a := -a;
  writeln('absolute value = ', a:8:5)
end.
```

```
program sam3b(input, output);
var a: real;
begin
```

```

write('A> '); readln(a);
if a < 0 then
    writeln('absolute value = ', -a:8:5)
else
    writeln('absolute value = ', a:8:5)
end.

```

実行結果はどっちも同じだから、片方だけ。

```

% pc sam3a.p
% a.out
A> -3.21
absolute value = 3.21000
% a.out
A> 1.235
absolute value = 1.23500
%

```

演習 1 ☆ 絶対値の例題の、好きな方を打ち込んで動かせ。

演習 2 ☆ 次の問題を解く PAD を設計せよ。

- a. 2つの数(実数)を読み込み、大きい方(両方同じ値ならその値)を打ち出す。
- b. 1つの数(整数)を読み込み、正、負、零に応じて1、-1、0を打ち出す。(ヒント: 枝わかれ先の中でまた枝わかれする。)
- c. 3つの数(実数)を読み込み、最大値(どれよりも小さくない値)を打ち出す。

演習 3 △ 演習 4 の PAD を Pascal にして動かせ。

3 日本語の入力/*

さて、これまで3回でだいぶキーボードになじんだと思うので、そろそろ日本語も打ってみよう。まず断わっておくが、皆様の端末についている「かな」キーは使用しない。ワープロでかな入力している人には泣いて頂くことになるので、ご勘弁頂きたい。以下すべてローマ字入力を前提とさせて頂く。¹

¹ここで説明するのは、「Wnn」(うんぬ)と呼ばれるかな漢字変換系と Mule 上で動作する「たまご」と呼ばれるインタフェースを組み合わせたものである。Unix ではほかに「sj3」と「かな」と呼ばれるシステムが広く使われているが、大筋はどれでも同様に操作できる。ただしカスタマイズなどの細かいところは違っているので注意。

3.1 ローマ字入力/☆

日本語を打つには、Muleの窓で「Control-\'」を打つ。すると、窓の左下すみに「あ」と現われるはずである。これは「ローマ字モードに入ったよ」ということを意味している。ローマ字モードで再度「Control-\'」を打つと元の英字モードに戻る。

ここで今度は「watasiha」と打ち込むと、画面には「わたしは」と出る。つまり、打ち込まれたキーから直ちにローマ字かな変換がなされて、変換結果のかなが画面に出るわけである。なお、ローマ字というのはまっかな嘘で、本当のローマ字なら「watasi wa」ですよ？ つまり、かなを打ち込むための便宜的手段としてローマ字「のようなもの」を打つというだけだと割り切って頂きたい。例えば次のような変な便宜がある。

n	→ ん (普通)
n'	→ ん (普通)
N	→ ん (うーん)
nn	→ ん (いやなら止めることができます。)
-	→ ー (長音の記号です。)
xa	→ あ
xi	→ い
...	
fa	→ ふあ
...	(あとは試してみてください。)

さて、打っている間その部分が「|」(たて棒)で囲まれているのにお気づきだろう。この状態を「フェンスモード」といい、まだ入力未完成を示す。打ち込みたいのがひらがなであれば、ここで[RET]キーを打つとフェンスが取れて(「確定」という)打ち込みが完了する。

3.2 漢字変換/☆

では、漢字にしたい場合は…？ その時は、フェンスモードの状態ですべての[SP](スペースバー)を押す。すると、かな漢字変換サーバと接続され、漢字への変換が起動される(最初だけちょっと遅い)。しばらく待つと、漢字への変換が行われ、「私は」になる。この原理は単純で、計算機の中に読みと漢字表記の対応を収録した国語辞典のようなもの(変換辞書)があって、この中から読みのあてはまるものを探してきて置き換えてくれるわけである。

最初に出てきたものでよければ[RET]で確定するか、単に次の言葉を打ち始めるだけでよい(自動確定)。しかし日本語には同音異語が沢山ある

ので、1発で成功しないことも多い。その時はさらに [SP] を押すと「渡しは」のように別の候補がでてくるので、次々に見て行けばよい。行きすぎた時は…? その時は^{^P}で前の候補に戻る。(実は [SP] の代わりに^{^N}でも同じ。)

ところで、場合によってはいくら候補を見て行っても目的の漢字が出て来ないことがある。それには以下の3つの場合がある。

1. 区切り損ない: 例えば「不自由」を出そうとして「ふじゆう」と打ったのにこれを計算機が「ふじ/ゆう」と間違っ分けてしまうと、「ふじゆう」→「富士-友」→「不治-夕」→…のように、いつまでも別の「ふじ」が出て来るだけになる。この場合には、区切り方を調整しないとイケない。それには

^{^O} --- 文節のぼし

^{^I} --- 文節縮め

を使う。例えば上の場合は^{^I}を1回打つと区切りが「ふ/じゆう」のように変わるので、あとはうまく行くはずである。

2. 後の節の選択: 上でぶじに「ふ/じゆう」になったあと「ふ」が「不」になったとして、「不-事由」になってしまったとしよう。「事由」を「自由」に直したいが、このまま [SP] を使っても先頭の「不」が変わってしまう。その場合は

^{^F} --- 先の区切りへ行く

^{^I} --- 前の区切りへ行く

を使う。この例では^{^F}でカーソルを「事由」の所へ移してから [SP] を使えばよいことになる。

3. 辞書にない: 自分の名前などは固有名詞だから、もともと変換辞書に入っていないかも知れない。その時は、(1) まずそれぞれの漢字を含む単語を考えて変換し、いらぬ字は削ってしまっるとにかく問題の漢字を作る。(2) 次回に備えてそれを辞書に登録してやる。

辞書への登録については少し後で説明する。

3.3 その他の字種/*

以上で「ひらかな」と「漢字」の入れ方はわかったが、では「カタカナ」は? それは、実は漢字変換すると最初にする「直前の」候補は変換内容を全部カタカナにしたものなので、一度 [SP] で変換してすぐ^{^P}で前の候

補を出すとカタカナになる。しかしそれでも不便だから、「コンピュータ」などよくあるカタカナ語は変換辞書に入っていて普通に変換するとでてくる。だから自分の使うカタカナ語で変換辞書にないものが見つかったらすぐ登録してしまうことをすすめる。

普通の英字や数字は？ それは、Control-\`でローマ字モードを抜ければ入れられますね？`

ところで、英字モードの字は画面上では漢字の半分に見えるが、ローマ字モードで記号などを打ち込むと漢字と同じ幅の記号が出てくる。実は英字モード (1 文字を 8 ビットで表現) で使っている英数字や記号すべてと同じものが日本語モード (1 文字 16 ビット) の文字セットにも含まれている。そして、Mule をや Kterm をはじめ多くのソフトが 16 ビットの文字を 8 ビットの文字の倍の幅に表示し、²たとえば「A」と「A」が実は同じ文字だということをちゃんと理解してくれない (別のものだと思って扱う)。人間が見るぶんにはまあどちらでも用は足りるが、計算機に食べさせる場合には注意すること。たとえば Pascal のプログラムでは文字列の内部以外に 16 ビットの文字を使ってはいけない!

演習 4 ☆ 以下の文章をできるところまで打ち込んでみよ。行かえとかは適当でよい。³

特別急行「富士」は、戦前のもっとも豪華な列車だった。

洋食堂車があり、一等寝台車があり、最後尾には展望車が連結されて、富士山をかたどったテール・マークが誇らしげに架けられていた。

展望車のデッキに立って、群れ集まった見送り人に鷹揚に会釈する高官、固い表情の軍人たち。腰にサーベルの警官や白手袋の駅長もいて、東京駅の 8 番線は物々しかった。

小学生の私は、それを遠くから眺めていた。「富士」を見たくて、幾度も東京駅へ行った。展望車には後光がさしているようで、眩しかった。

宮脇俊三、「シベリア鉄道 9400 キロ」より

²このため、8 ビットの文字を「半角」16 ビットの文字を「全角」と呼ぶ人が多いが、あくまでもソフトの勝手に半分の幅に表示しているだけなのだから本当はおかしい。

³日本語のかっこは【と】を打てば出る。「・」は「z/」で出る。

3.4 辞書登録/*

単語登録とは、先にも述べたように変換辞書に載っていない単語を追加することである。この場合、追加は各ユーザごとの個人辞書になされるので、他人と登録がぶつかる心配はない。一方、一度登録してしまえば南棟の Unix を使っている限りそれはずっと有効である。だから変換できない語やカタカナ語が見つかったらそのつど、すぐに登録することを勧める。

登録するには、まずその語を何とか別の方法で入力する。漢字については、その字を含んだ別の単語を考えてそれを変換して出し、いらない漢字は削ってしまうなどする。たとえば先の「鷹揚」だと、「三鷹」と「揚げる」から作るなどする。どうしても思いつかなければあまたある「ワープロ漢字辞典」の「JIS コード」が引けるものを購入して文字の JIS コードを調べ、

```
[ESC]xjis[RET]xxxx[RET]
```

で入力する (xxxx のところに JIS コードを打つ)。

問題の語がともかくできたら、まずその先頭にカーソルを持ってきて ^[SP] (Control+スペース) を押し (ここで Mark Set と表示される)、次に登録する語の直後まで (つまりその語の文字数ぶん) カーソルを移動する。この状態で、

```
[ESC]xtoroku-r[RET]
```

と打つと登録モードになり、最下行に語が表示され、読みを聞いてくる。読みはローマ字入力で普通に打ち込み、最後に [RET] する。次に登録辞書を聞いてくるがこれはただ [RET] する。その後品詞を聞いてくるので、自分の文法知識の範囲で正しく答えると、登録が完了するはずである。

演習 5 Δ 先の演習で出なかった語や自分/知り合いの名字、名前など変換できない語を最低 5 個以上探して登録せよ。

演習 6 Δ 自分の好きな日本語の歌を 1 つ選び、その歌詞を打ち込め。ただし、できる限りタッチタイプするように。

4 電子ニュース

4.1 電子ニュースの概要/Δ

電子メールが個人から個人への「お手紙」だとすれば、電子ニュースは「新聞」に相当する。ただ、普通の新聞と違うのは、誰でも (あなたでも) 記事を書けることと、新聞の数がやたら多い (話題が細分化されている)

ということである。電子ニュースの用語では各新聞のことを「ニュースグループ」と呼んでいる。

(Internet の) 電子ニュースには世界中の何万という計算機がつながっていて、そのどこから投稿した記事でも地球の反対側まで伝播して行って読めるようになっている。そんなことがどうしてできるのか不思議ですか? たとえば、あなたがここで投稿した記事を、ここから2箇所に伝達し、その2箇所がまたそれぞれ別の2箇所に伝達し…のようにしたとする。すると、それを20段繰り返しただけで百万箇所に伝達できることになるでしょう? 実際に起こっていることはこれとはちよっと違うが、原理はまあそんなものである。

さて、ニュースグループが沢山あるという話は上で出たが、それぞれのニュースグループの名前は例えば

```
komaba.lectures95.jousho.kuno-2
```

のように大分類→小分類の順で階層構造の形になっている。その一番上のカテゴリーの代表的なものとして次のものがある。

```
komaba    --- 駒場地区ローカルなもの
tecc      --- 教育用計算機センターのもの
u-tokyo   --- 東大全体
fj        --- 日本全体 (fj == From Japan)
comp      --- 世界中、計算機がらみの話題
rec       --- 世界中、趣味/娯楽の話題
soc       --- 世界中、社会的な話題
alt,misc  --- 世界中、一般の話題
```

ほかにも一杯あるが、とりあえずこれくらいで十分以上だろう (mnews を久野クラス設定で使う人はこれだけが見えるように設定してあります)。

4.2 ニュースを読む/*

ごたくはそれくらいにしてニュースを見て頂こう。ニュースを読むためのプログラム (ニュースリーダー) は「補遺」に説明されている gnus をはじめ多数あるが、ここでは比較的初心者むけの mnews というニュースリーダーを使用する。mnews を動かすには Kterm の窓で「menws[RET]」と打つ。すると、図6のような初期画面が出てくるはずである。

ここには、上で説明したようなカテゴリー (トップレベル) の一覧が出ている。その中でどの行へ進むかは、Mule と同様 ^N と ^P のキーでカーソルを動かして選択する (やってみよ)。(あと、記事やグループ一覧などで1



図 6: mnews の初期画面

画面に入り切らない時は[^]Fと[^]Bで1画面ずつ前後に移動することもできる。リストの先頭と最後に行くにはそれぞれ [ESC]<と [ESC]>である。)

ところで、この状態で「l」(小文字のエル)を押すと画面が書き変わり、「U」の印のついたカテゴリーが沢山出てくる。「U」は「見るつもりないから表示しない」という意味を持っている。カーソルをその行に持って行って「u」を押すと印が取れる。また印のないところで押すと印がつく。以下の課題等のため、「alt」を含め上に説明したカテゴリーは印なしにしておいて欲しい。再度「l」を打つと「U」のついたものは表示されなくなる。

次に、「komaba」のところへカーソルを持って行って「i」を押すと画面が書き変わり、「komaba. なんとか」を選ぶ一覧になる。ここで「komaba.forum」のところへカーソルを持って行って「i」を押すと、今度は図7のように記事のタイトル一覧になる。つまり、グループ名のところまで階層を降りてくると記事一覧、それより上の中間レベルではグループ名(ないしカテゴリー)一覧が表示され、カーソルでその中から選んで「i」を選ぶと階層を降りていくことができる。逆に「q」を打つと階層を上へ上がっていく。

記事一覧のところ「i」を打つと、その記事の内容が画面に表示されて読むことができるようになる。読み終わって戻るのもやはり「q」である。ところで、読み終わるとその記事のところに「R」の印がつくが、これは「この記事はもう読んだ」という意味である。そして、[^]Nと[^]Pの変わりに「n」と「p」を使うと読んだ記事は飛ばして移動してくれるので、まだ読んでいない記事だけ効率よく読むことができる。あと、タイトルだけ見て「読むに値しない」と思った時は「d」を打つとで読んだ印をつけられる。また、どこでも「c」を打つと、現在カーソルのあるカテゴリ/

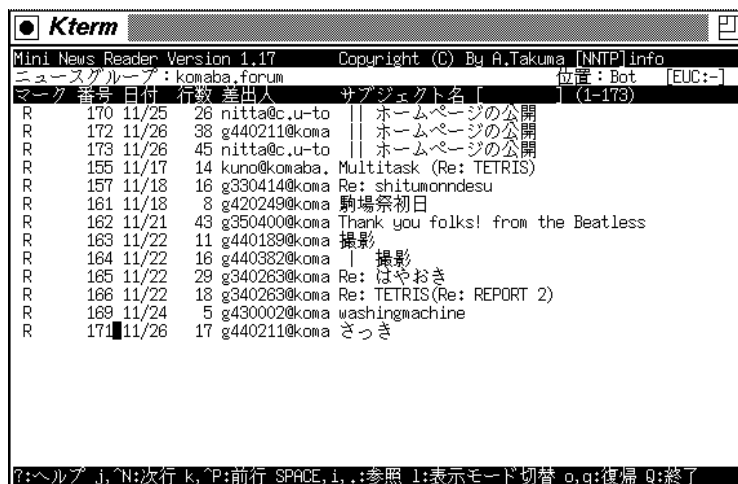


図 7: mnews の記事一覧画面

サブカテゴリ/グループの記事全部に印をつけられる。

mnews を終わらせるには、一番外側のカテゴリ一覧でさらに「q」を打つ。または、どこでも「Q」(大文字)を押すと一気に終わることができる。mnews は「どこまで読んだか」の情報を終わるときにファイル「.newsrc」に書くので、このどちらかの方法で正しく終わらせないと読んだものの情報が失われてしまう(つまり、次回 mnews を起動した時に読んだはずのものが「読んでいない」ことになってしまう)。

演習 7 ☆ komaba カテゴリの下にどんなグループがあるか探検し、記事をいくつか読んで見よ。

4.3 記事の投稿/☆

さて、読み方が分かったところで、次は投稿である。記事を投稿するには、まず正しい(話題にふさわしい)グループを選ばなければならない。最初は投稿の練習をするのだから、komaba.test(駒場地区用練習グループ)を選ぶことにして、その記事一覧画面へ行く。この状態で「a」を押すと、mnews はまず投稿記事の Subject:(タイトル)を聞いてくる。この状態での漢字入力はできないので、とりあえず英字のみを使って英語かローマ字で書き、最後に [RET] を打つ。次に Distribution:(配布範囲)を聞いて来るが、world[RET] と打つ。すると今度は Mule が起動され、本文を書くモードになる。

Mule の中では電子メールの comp と同様、記事の先頭部分も編集できるので、Subject:を漢字で書き換えたりしてもよい。肝心の本文は「text

follows this line」より下に書く。ここにはもちろん漢字を使って良いが、alt など海外が中心のグループでは漢字を読めない人が大部分だからやめておいた方がよい。書き終わったらいつも通り`^X^S`で保存し、`^X^C`で終了する。すると、もし自分が`.signature`ファイルを持っている場合にはそれをつけ加えるかどうか聞かれるので「y」か「n」で答える。次に完成したメッセージを見るモードになるので、スペースキーを使って最後まで見て行き、最後で再度スペースを押すと「投稿してもいいか」と聞いてくるので、ここで「y」を押すと投稿される。投稿した記事は一担 mnews を終わらせて再度起動した時に見えるようになっているはずである。

演習 8 ☆ komaba.test の下に記事を投稿し、確かに投稿できたかどうか確認してみよ。

以上は新規投稿だったが、メールの返信と同様、既にある記事に対して「フォロー」を投稿することもできる。その場合には、記事一覧の画面でフォローしたい記事にカーソルを置いた状態で「F」(大文字)を打つ。すると、Subject:はもとの記事から作られ、また本文ももと記事からの引用がなされた状態で Mule が動き出すので、ここで余分な引用などは削除して本文を組み立てる。(Mule でいらない行を消すには`^K`を連打する、というのを思い出しておくように。)

演習 9 △ komaba.test の下にある、自分以外の誰かが投稿した記事にフォローしてみよ。

だいぶ色々出たので、以下に mnews のコマンドキー一覧を載せておこう。

l	Unsubscribe したものを表示するかどうか切り替える
u	今いるカテゴリを Unsubscribe する、またはしていたのをやめる
^N	1つ下のカテゴリ/記事へ行く
^P	1つ上のカテゴリ/記事へ行く
^F	1画面先へいく
^B	1画面前へいく
[ESC]<	リストの先頭へいく
[ESC]>	リストの末尾へいく
i	そのカテゴリ/グループの中を表示
q	現在いるカテゴリ/グループから上へ戻る
Q	一気に mnews を終る
a	現在一覧の出ているグループに新規投稿
F	現在カーソルのある記事にフォローアップ

A 本日の課題 **4A**

本日の課題は、アンケートのニュースへの投稿のみです。グループとしては、必ず

「komaba.lectures95.jousho.kuno-2」を使用し、また Subject: は「Report 4A」とすること。アンケートの内容は次の通りです。

- Q1. プログラムにおける枝分かれについて、どのように感じましたか?
- Q2. 計算機における日本語入力はどれくらい簡単/大変ですか? どのようなところが?
- Q3. その他、感想、要望、質問があればどうぞ。

B 次回までの課題 **4B**

次回までの課題は、次の通りです。

- 演習 2 と 3 の b または c (できれば両方)。つまり、PAD とプログラムと実行例。
- ペンパルさんとどんなやりとりをしているかの簡単な紹介を、久野クラスのグループ (4A を投稿したところ) に各自出す。Subject: に「Report 4B」と必ずつけること。

後者についてはニュースを直接拝見しますから紙のレポートには記載不要です。レポート番号は **4B** です。アンケートは次の通り。

- Q1. プログラムにおける枝分かれについて理解しましたか? しなかったとしたら、どの辺に問題がありますか? したとしたら、どの辺が「要点」だと思いましたか?
- Q2. ニュースに投稿できるようになって、前に Mosaic 経由で見た時と違う感想を持ちましたか? 持ったとすればどんな?
- Q3. その他、感想、要望、質問があればどうぞ。