

次第立て言語《小朱唇》の設計思想

水谷 静夫 (東京女子大学 日本文学科)

0. 序

計算機が使えることが偉かった時代は過ぎたはずである。文科系——それも特に語文系では、本気のアマゲラマが育ってほしい、つまりマニア学生やパッケージの消費者的学生より、自分の問題を有ちそれを解く道具として計算機を使う学生が出てもらいたいのに、なかなかそういかない。原因は色々考えられるが、その一つに、既成のプログラミング言語が概して連系操作の為には使い勝手がよくないという事が挙げられよう。(Snobolが恐らく唯一の例外。) 小朱唇はこの不具合を軽減する目的で設計した教育用次第立て言語である。

語文系の計算機利用は大別して下記の二類三種になろうが、定型的処理が少ないのも特色と言えよう。

1) 資料調製 a) 言語材料の整理・探索・編集

b) 上記の結果の統計解析など(数値計算) ... 2の場合と有り

2) 言語・文学の理論上の仮説の検証(立証又は反証)

このうちaと2とでは連系操作が中核となる。特に2では、規模は小さくても思い着きをすぐ試してみることができなくては効果が落ちる。プログラムの実行効率など云々するよりずっと手前の、広義の計算機環境・プログラミング環境の所で、大方の者は気が滅入ってしまう。せめてプログラミング言語だけでも何とかして、注意を専ら問題解決の考え方に向けられるようにしたかった。待っていてもちろしの希望に叶うものが出そうもないから、文学士にあるまじき事ながら、「盲蛇ニ怖ガズ」で新言語の設計に手を出した次第である。

1. 連系操作における基本的操作

言語を計算機で扱うには、今のところ、連系という在り方を介するはか無い。従ってその基本的操作として何を選ぶかの問題は一考に価する。純理論的には、Snobol風の言い方で、型要素に連系変項を許した碇揚げ様式の置き換えという操作(と、その成否を検出して次の操作を選ぶ手段と)だけで済む[1]と考えられる。但しこれは操作としてかなり高水準での話になる。小朱唇の設計に当って次のものを基本操作と考えた:

(0.1) 対象が空連系か否かの判定

(0.2) 基本記号に関する同一性の判定

(1) 二連系の連ね演算

(2) 型合せ(連系変項を含み得る型との順方向分割で代表させて可)

(3) 連系の複写(入出力もこの一種と見る)

連系変項の存在が重要で、これにはSnobol同様 単純連系変項 長さ連系変項 釣合連系変項 の区別を立てる(但し釣合連系変項をSnobolのと同じとはしなかった)。序でながら、リスト構造は連系そのものではないが、括弧入れ表示形を連系と見て、リスト操作を釣合連系変項の利用でまねることはできる。数値計算が連系操作に還元できることも周知の通りである。しかし小朱唇での数値計算はそこまで落してはいない。

2. 小朱唇設計上の留意点

小朱唇は、東女大日文の学生に 90 の 2 や 2 の 計算機利用をする際の考え方を教える為の教具になる事を旨として、出発した。この授業の登録学生の過半数は計算機につかまるのが初めてであり、彼等らには Basic の表現でも jargon めいた恐ろしさを感じるらしい。そればかりが理由ではないが、

一、小朱唇のいわば地の部分は国語表現とする。

なお、計画の当初から処理系は Lisp に乗せる予定で、それ故 REDuction-to-List-Processing System の名を用意し、その略語に対応する「朱唇」をこの日本語次第立て言語の名前とした。処理系が発する事故通報も当然国語表現にしてあり、これが学生には殊のほか好評であった。

二、小朱唇構文法の大枠は 3 型 PSG で律せられるようにする。しかしその為に国語表現として許されないような形は採らない。国語を乱すな！

三、大筋の規定から細部の規定へと書き進められる組立てを採る。従って次第書きの一般的な姿は、

 標題行

 骨組部

 必要なだけの下請・函数の群れ

 オフリ行

四、構文解析などの仕事に必須ゆえ、下請も函数も頭山 (recursion) で呼べる。

五、流れの制御は「型合せの成否に基づく有無帰結句 仕訳句 (Lisp の cond に相当) 巡回句 (Lisp 1.9 の loop や for に相当) 及び脱出句 下請との往來の範囲でまかなう。

六、材料 (data) 型は設けないのが原則。実行上の事を考えて連系と数との区別はするが、次第書きの文脈環境から変換を要しかつその変換が可能な場合には処理系が面倒を見るという方針で臨む。(二進数だけは例外。)

特別な材料構造として 陣立て (array) を認める。一つの陣立ての要素に連系と数とが混在してよい。

七、朱唇固有語は片仮名、使用者が付ける名前は一ローマ字という風に用字範囲を変えることによって、所謂予約語は無しにする。

次第書きが所謂自由書式である事は、留意点に挙げるまでもあるまい。

連系の基本的操作との関係で言えば、従前のプログラミング言語の多くは強力な型合せ機能を持ち合わせないばかりか、連ね演算についてさえ実際上の不便が著しい。またその処理系でも、空連系が安心しては使えない——例えばクリアに閉じ零・空白・空連系のどれを割付けるかは本質的な違いなのに、この区別に無頓着である、等——というのが、実感である。連ね演算の単位元である空連系の扱いに信が置けないようでは、まともな連系操作など到底望めない。従って

八、空連系処理が連系理論の言う通りに確実に扱えるようにする

という、ごく初歩的な事に十分留意する。(序でに言っておけば、連系内の着目位置に関わるポインタは、通例の処理系では文字——基本記号を指すようであるが、字間を指す方がはるかに好都合である。文字位置を指す仕方では、1 字分の部分連系と空連系との区別が自然な仕方ではしかねる。) 連系操作で型合せが担う実際上の役割は極めて大きい。単一連系常項の型合せをすることとポインタ情報を与えることと程度の機能を具えるだけでは幼稚に過ぎる。そこで、割合によ

くいる型の種類を考えた上で、

九. ある程度まで複雑な構成をとる型による型合せが（ポインタ操作をあらわに処方することを要せず）一つの句で指図できるようにするが、これは既に Snobol に実績がある。これに関し

十. 連系変項という考えは不可欠。実用上、Snobolに見られた三種の連系変項は有用である。理論上、変項と常項との別を立てるのは当然として、名前を安易に変項と同一視するような態度を執るべきでない。なお、函数についても、引き項（の順序対）に対して一意的な（項の水準での）値を得ようとする働きとして見、この中で入出力を働かすとかその函数の外部の値に影響するとかの事が無いように努める。

以上に列挙した諸点には、別に創見を称し得るものが無い；どれもごく普通の事に思われる。（中には趣味の問題と言われそうなのもあるが、それは別にしよう。）その至って普通の事が全体として成就しているプログラミング言語（や処理系）が無かったことこそ、見逃してはならない問題であろう。——人文学者を代弁して敢えて言っておきたい：現在の計算機環境は人文学者にとってすこぶる使い勝手が悪く、苦痛を忍んで「足を靴に合わせ」なければ一步も歩いてはくれない；その点では欠陥製品とさえ思える。

3. 次第書きとその実行結果との一例

次ページの図1は、末子相統制の下での系図と同じ仕方で構文の木を描かせる小朱唇次第書きである。その出力例と入力材料帳の中身とは、図2の通り。

なお、現在の処理系は黒川利明氏に負うもので、LISP 1.9によるインタプリタになっている。Lispもインタプリタで走らせているから、実行速度は決して速いとは言えない（マシンは TOSBAC ACOS 600）。

図1の手続全体を他のもっと大きい処理の下請に作り変える事は、極めて簡単である。

発表当日は他の若干例も OHPでお目に掛ける。

4. 小朱唇を使わせられた学生の声

- 日本語めいたもので計算機が働いて、驚いた。初めは何だか気味悪かった。
 - こういうものでも動くのに、普通はどうして英語まがいなのでしょう？
- 事故通報が日本語で出るのが有難い。英語のは何だかよく分らない。
- 型合せが大概は一行で済むのが楽だ。考えた事の通りに書けていい。
- 国文法的には許される表現なのに、小朱唇文法に無い指図をしてしまって事故を出した。つい気を許してしまう。注意しなければ……
 - 「 $\alpha = \beta$ ガアレバ」をうっかり「 β ガ $\alpha = \text{アレバ}$ 」と入れて受け付けられなかったが、助詞まで添えるのだから、どちらでも通るようにしてほしい。
- いっそ漢字かな混り文だったら、もっと読みやすいのに！
 - キーボード入力の際シフトキーを頻繁に押さなければならないのが厄介。
- Lispの尻尾が切り離せませんか。あの“呪文”が苦手です。{筆者注：これは現在の処理系に向けられたもの。他にもそういうのがあったが、省く。}
- 小朱唇も習ってみた結果、本当にむずかしいのは解き方なのだと気づいた。
 - @タドレ{小朱唇tracer}の併用で頭山の働き方が目に見えてきた。

```

REI14L TDLカ"タノキ (マツシ サウソ"ワ ケイツ")ヲ カフ (HOH 860103) :
トリキメ: % コト"モ ノ アリカタ オヨヒ" ハンシフヨウ シ"ヤウカウ %
ANI トハ 「ア。」 OTO トハ 「オ。」 SUE トハ 「ス。」 HTR トハ 「ヒ。」,
SEN トハ 「ー」 KAG トハ 「キ」 CLN トハ 「:」 DEQ トハ 「==」
X トハ % イマ 60シ"フン ノ クウハク %
「
AT トハ 「@」 SP トハ 「」 SP3 トハ 「」 :シメ。
ヒナカ"タ P:-----** ショリ シ"カン ###.##"ウ **
「TDL サ"イレウチヤウ」ヲ カケ, FL ニ ヨメ, ヨミチヨウ [FL]ヲ ヒラケ, [FL]ノ アタマ ハ。
メク"リ
:. タンマツ カイキ"ヨウ。 「」ヲ K ニ ウツセ。
メク"リ
:. [FL] カラ キ"ヨウヲ G ニ ヨメ。 ナケレハ" [FL]ヲ トシ"ヨ, ヤメヨ。
G ノ スエ ニ 「キ」カ"アレハ" スケタ"セ, ナケレハ" K_Gヲ K ニ ウツセ。:.
K_Gヲ #()-KI_「キ」ニ ワカテ。
アレハ" シ"カン==>TK,
@TDL.KAKI, (シ"カン TK - 5 + 1000 /)=>TK
ヒナカ"タ P テ" TKヲ カケ;
ナケレハ" 「 ##ILL-DATUM## 」_K_Gヲ カケ。:.

*シタウク @TDL.KAKI : トウサ"ノ ナマエ ONO,ZAN。
O==>C。 % Cハ シュツリヨク セイキ"ヨ ヨウ %
KIヲ #ONO_#()-SAKI_#ZANニ ワカテ。
ナケレハ" KI テ" ATヲ DEQニ セヨ, KIヲ カケ, シタウクヲ スケタ"セ。
シワケ :. ZAN=カラ ナラ [ONO HTR SAKI]テ" @KO, シタウクヲ スケタ"セ。:.
<ONO>/ZIAKIヲ AKIニ ウツセ。
[AKI ANI SAKI]テ" @KO。
メク"リ :. ZANヲ #()-SAKI_#ZANニ ワカテ。
シワケ :. ZAN=カラ ナラ [ONO SUE SAKI]テ" @KO, スケタ"セ;
ホカハ [AKI OTO SAKI]テ" @KO。:. :シメ

*シタウク CUME K IEJ テ" @KO : トウサ"ノ ナマエ ONO,ZAN,L。
IEヲ #ONO_#()-SAKI_#ZANニ ワカテ。
ナケレハ" 1==>C, IE テ" ATヲ DEQニ セヨ,
シワケ :. K<SUE ナラ UME_KAG_IEヲ カケ;
ホカハ UME_SEN_IEヲ カケ。:. シタウクヲ スケタ"セ。
シワケ :. ZAN=カラ ナラ :. UMEノ スエ ニ SPカ"
アレハ" CUME_KAG_ONO HTR SAKI]テ" @KO
ナケレハ" CUME_SEN_ONO HTR SAKI]テ" @KO。
シタウクヲ スケタ"セ。:.:.
<ONO>/ZIAKIヲ AKニ ウツセ。 % LEADノ ヨウイ 4キ"ヤウ
シワケ :. C=1 ナラ @L.EDIT;
ホカハ シワケ :. K<HTR ナラ UME_SP3_AKヲ Lニ ウツセ;
ホカハ <UME>/ZIAKI_SP3_AKヲ Lニ ウツセ。:.:.
EL ANI SAKI]テ" @KO。 % フンク スル
メク"リ :. ZANヲ #()-SAKI_#ZANニ ワカテ。
シワケ
:. ZAN=カラ ナラ :. UMEノ スエ ニ SPカ"
アレハ" CUME_KAG_ONO SUE SAKI]テ" @KO
ナケレハ" CUME_SEN_ONO SUE SAKI]テ" @KO。
スケタ"セ。:.;
ホカハ EL OTO SAKI]テ" @KO。:. :シメ

*シタウク @L.EDIT :
UMEヲ #A_KAG_#Bニ ワカテ
ナケレハ" :. UMEヲ #A_SEN_#Bニ ワカテ
ナケレハ"
:. UMEニ CLNカ"
アレハ" シワケ :. K=OTO ナラ UME_CLN_AKヲ Lニ ウツセ;
ホカハ UME_SP3_AKヲ Lニ ウツセ。:.
ナケレハ" <UME>/ZIAKI_CLN_AKヲ Lニ ウツセ。
シタウクヲ スケタ"セ。:.:.
シワケ :. K=HTR ナラ A_CLN_<B>/ZIAKI_SP3_AKヲ Lニ ウツセ;
ホカハ A_SP3_<B>/ZIAKI_CLN_AKヲ Lニ ウツセ。:. :シメ

*カンスウ <Z>ノ ZIAKI : Xカラ #/<Z>/ナカ"サ/Yヲ トレ, Yカ"コタエ。 :シメ
オワリ

```

? [CSEYO 'S14L]
 === ショウシュン タ イ1.2ハン [86-01-25] === (02/13/86 13 12 . 45)
 TDL サ イレウチャウ
 ・ BUN

← 下線部が端末入力

```

      「 タイケン == ノ
      「 レンタイ == ノ
    「 Dソ - Tレンコ - タイケン == ハナ
  フン - Kワ - [クワン] == ヨ
      ** ショリ シ"カン 2.48^ウ **

      「 タイケン == ノ
      「 レンタイ == ノ
    「 Zソ - Tレンコ - タイケン == ハナ
    「 Zワ - Zシ" == E1
  フン - [ムスビ] - テンカ0 == ヨ
      ** ショリ シ"カン 3.32^ウ **

      「 Zソ - ケイヨウシ == アライ
      「 ウメコミワ - Zシ" == E3
      「 レンタイ == .
    「 Tレンコ - タイケン == ナナキ
    「 [カワ] == カ
    :      「 タイケン == カセ
    :      「 [カワ] == ニ
    「 Zソ - Yレンコ - ト"ウシ == ナヒ"イ
    「 Zワ - Zシ" == タ
    :      「 フテイシ" == カ シラ
  フン - [ムスビ] - テンカ0 == ネ
      ** ショリ シ"カン 8.10^ウ **

      「 タイケン == ト"テ
      「 レンタイ == ノ
    「 Tレンコ - タイケン == ナナキ
    :      「 カワシ"ヨ == E9
    「 [カワ] - カカリ == ハ
    「 Zソ - タイケン == カセ"マカセ
    「 Zワ - Zシ" == E4
    「 ク"ケツカ"フシ - [ヒラキ] == .
    :
    :      「 Zソ - ケイヨウシ == カハ<イ>イ
    :      「 ウメコミワ - Zシ" == E3
    :      「 レンタイ == .
    :      「 レンタイシ == アノ
    :      「 [レンタイ] == .
    :      「 Tレンコ - Tレンコ - タイケン == コ
    :      「 カワシ"ヨ == E9
    :      「 [カワ] - カカリ == ハ
    :      「 Zソ - タイケン == ワチマカセ
    「 ク"フカ"フ - Zワ - Zシ" == E1
    「 Zワ - Zシ" == E1
  フン - [ムスビ] == .
      ** ショリ シ"カン 20.88^ウ **

```

===== ス ミ ===== (13 13 . 49)
 ? [CALLSS 'LIST/ BUN]

```

(フン(Kワ(Dソ(Tレンコ (タイケン@ノ) (レンタイ@ノ) (タイケン@ハナ)) (クワン@ヨ))$
(フン(Zワ(Zソ(Tレンコ (タイケン@ノ) (レンタイ@ノ) (タイケン@ハナ)) (Zシ" @E1)) (ムスビ" (テンカ0@ヨ))$
(フン(Zワ(Zソ(Tレンコ (ウメコミワ(Zソ(ケイヨウシ@アライ)) (Zシ" @E3)) (レンタイ@ノ) (タイケン@ナナキ))
([カワ@カ]) (Yレンコ (タイケン@カセ)) ([カワ@ニ] (ト"ウシ@ナヒ"イ)) (Zシ" @タ)) (ムスビ"
(フテイシ" @カ シラ) (テンカ0@ネ))$
(フン(Zワ(ク"フカ"フ(Zワ(Zソ(Tレンコ (タイケン@ト"テ) (レンタイ@ノ) (タイケン@ナナキ)) ([カワ]
(カワシ" @E9) (カカリ@ハ)) (タイケン@カセ"マカセ)) (Zシ" @E4)) (ク"ケツカ"フシ([ヒラキ@ノ]) (Zワ(Zソ
(Tレンコ (ウメコミワ(Zソ(ケイヨウシ@カハ<イ>イ)) (Zシ" @E3)) (レンタイ@ノ) (Tレンコ (レンタイ@アノ)
(レンタイ@ノ) (タイケン@コ)) ([カワ] (カワシ" @E9) (カカリ@ハ)) (タイケン@ワチマカセ)) (Zシ" @E1)
)) (Zシ" @E1)) (ムスビ" @ノ))$

```

文献

- [1] 水谷静夫 (1973) 何が言語処理の原始的操作か. 『計量国語学』[67] 1-8.
 [2] — (1974) 連系操作プログラミングの一法. 『計量国語学』[70] 17-29.
 [3] — (1983) 《小朱唇》, 教育用の次第立て言語. 『計量国語学』[43] 87-112.
 [4] — (1986) 『次第立て言語小朱唇の手引』 東女大日本文学科.

附録: 小朱唇構文法 (改訂: 85-05-24; 第1.2版から)

下記引用では 凡例 を省略した.

I 語彙規則

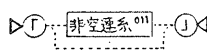
対象記号の部

対象記号として使ってよいのは, 下記四種類のどれかに属する記号である.

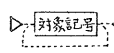
片仮名類 「キ」「エ」以外の清音片仮名及び小字の ャ ュ ョ ャ イ ク ス オ 並びに 濁音符・半濁音符・長音符
 ローマ字 「A」から「Z」までの立体的ローマ大文字
 数字 「0」から「9」までの立体的アラビア数字 (「0」以外を非零数字と謂ふ.)
 其他類 空白記号及び下記諸記号: —
 . , ; : . ! ? 「 」 ' () [] + - * / < = > ¥ ¢ # % @ ^ _

材料の部

010 引用形

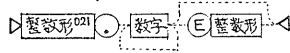


011 非空連系

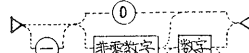


但しこの部分に^{xy}括弧や円記号を挟みには, その1記号をそれぞれ ¥「 ¥」 ¥¥ で表すことを要する.

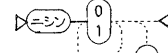
020 十進数項



021 整数形

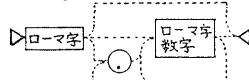


025 二進形

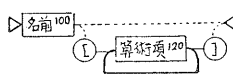


要素項の部

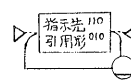
100 名前



110 指示先



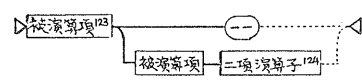
115 広義指示先



120 算術項 ::= 算術項⁰¹ | <算術項¹>

121 算術項⁰ ::= 整数形⁰²¹ | 十進数項⁰²⁰ | 指示先¹¹⁰ | 函数形¹⁴⁰

122 算術項¹

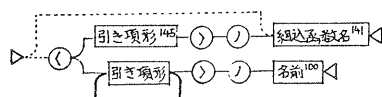


123 被演算項 ::= 算術項⁰ | 算術項¹

124 二項演算子

::= + | - | * | セイジヨ | アマリ | / | ** | & | ! | ¥

140 函数形



但し引き項形の数が, 組込函数の場合にはその所定数(すなわち組込函数の定数)に, その他の場合には 函数形のための 指定数に, 合ふことを要する.

141 組込函数名 ::= ヒツゲ | ジコク | ジカン |

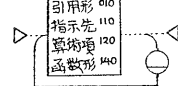
カタチ | コード | サカサ | ナガサ | セイスウ | ニシシカ | ヘイホウコン

(見掛け上は要素)

(事項)

145 引き項形 ::= 定まり型¹⁵⁰ | カラ

150 定まり型



(注) 意味論上の概念としては,

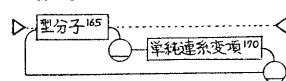
型 ::= 定まり型 | 動き型 | 拘束型

動き型 ::= 動き型⁰ | 算術連系変項¹⁷⁰ | 動き型¹ | 算術連系変項

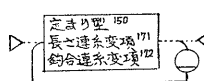
算術連系変項¹⁷⁰ | 動き型¹ | 算術連系変項

算術連系変項¹⁷⁰ | 算術連系変項

160 動き型⁰



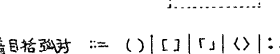
165 型分子



170 算術連系変項



171 長文連系変項



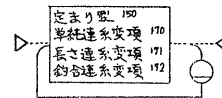
172 釣合連系変項



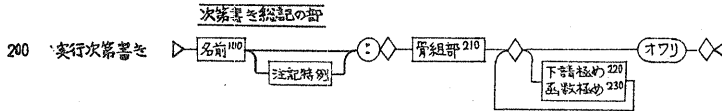
173 算術項¹²⁰ ::= () | [] | 「 」 | < > | : . . .

180 拘束型

これを狭く置き換へ句 (は 399) の中で先行するのが 動き型 0 の場合には、そこに現れてゐて指示先も有する連系変項 (170~172 の範囲でもよい) と全く同じ形のものでなければ拘束型の中に使つてはならないとの制限を課した上で、右に規定する姿のもの。 備考: 従つて定まり型は拘束型に拘束型である。



II 形成規則

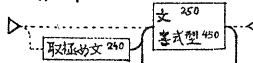


・注記について: 上記の注記特別を含め一切の注記は、本居表現に対してはメタ表現である。注記は次のどの姿もとらなければならない。

- 1) %注記表現% } 「%」の使用が許してあることを除くは任意の連系但し1行に収まることを要する(注記行を並べても可なり。)
- 2) 名注記変項% } 置ける位置は、横文法上空白が入れられる所ならどこでもよい。
- 3) 注記特別 } 「%」の使用も許してある注記表現。置ける位置は、規則 200 を見よ。

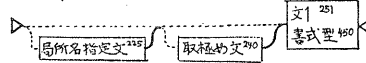
210 骨組部

下記で少なくとも一つの文は必要

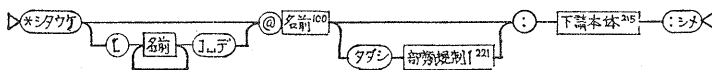


215 下請本体

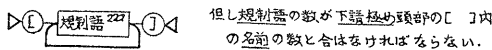
下記で少なくとも一つの文は必要



220 下請極め



221 奇数規則1



但し規則語の数が下請極め頭部の「」内の名前との数と合はなければならない。

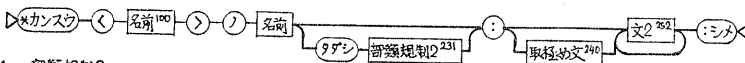
222 規則語

::= レンシ | カズ | ニシ | ニン

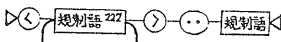
225 局所名指定文



230 函数極め

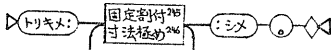


231 奇数規則2



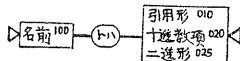
但し「」内の規則語の数が函数極め頭部の「」内の名前との数と合はなければならない。

240 取極め文

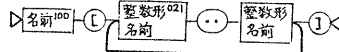


骨組部や下請本体の取極め文には極め極めも使つてよいが、特束禁止する予定なのでこゝには述べない。(ヒラゲ句で開く場合。)

245 固定割付

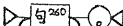


246 寸法極め



但し「」内の名前はそれに先立つ固定割付で数値を決めておくことを要する。

250 文



備考: 横文法上の文は取極め文も局所名指定文も含まない。

251 文1 左図の「句1」を「句1」に代へたもの
252 文2 左図の「句2」を「句2」に代へたもの

260 句 ::= 句0 | 入力句 | 出力句 | 振舞線句 | 終了句

261 句1 ::= 句 | 下請脱出句

262 句2 ::= 句0 | 函数値句

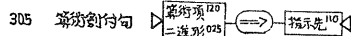
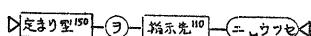
265 句0 ::= 割付句 | 算術割付句 | 有無帰結句 | 型合せ句 | 値取り句 | 置き換へ句 | 仕置句 | 巡回句 | 下請整注句 | 脱出句

「出力」に関係しない句の部

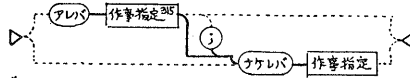
型印の閉き規則に於ける注意:

骨組部と下請本体と函数極め本体とは、従つてよい句の範囲にやゝ出入りがあるので、これを句1、句2 または、文1、文2 と略が分けた。後掲規則の中にはこの区別に応じた三通りを閉へる要のあるものがあるが、要をいって「文」「句」と記し一つにまとめる等にした。さういふ読み分けは規則には、番号の肩にも「1」「2」を附けておく。

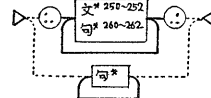
300 割付句



310* 有無帰結句

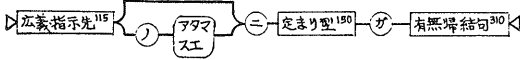


315* 作事指定

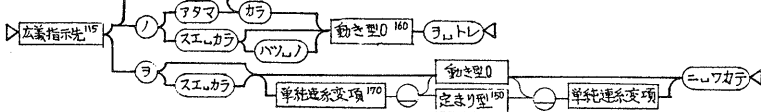


第ニ形は、ク句以上の時一般動詞には余り安全ではないが、略記として使ってもよい。それで生ずる模範の紛れは次第で手の費は。

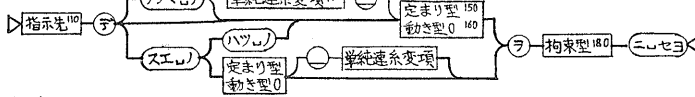
320* 型合せ句



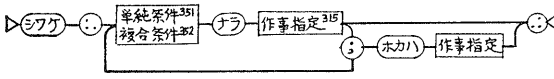
330 値取り句



340 置え換へ句

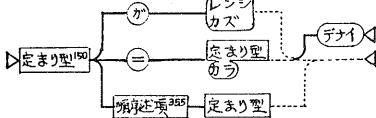


350* 仕訳句



但し作事指定から「.」に移る際、その作事指定も「.」で開けてあれば、間の空白は省いてよい。

351 単純条件

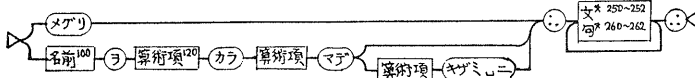


352 複合条件



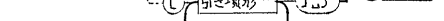
353 順序述項 ::= < | > | <= | >=

360* 巡回句



但し相違く「.」について、350の場合に同じ。

370 下請発注句



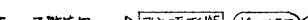
但し引き項形の数が、下請極めて指定した個数と合はなければならぬ。

380 脱出句



但し巡回句・下請本体の中でしか使ってはならない。

382 下請脱出句



但し下請本体の中でしか使ってはならない。

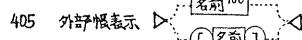
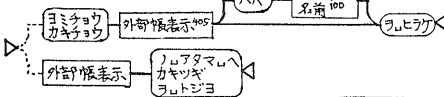
385 函数述句



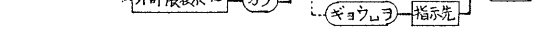
390 終了句 ::= ヤメモ

入出力の部

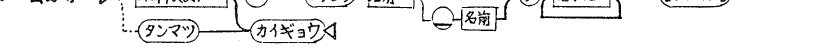
400 帳簿繰り句



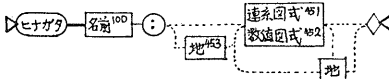
410 入力句



420 出力句



450 書式型



451 連系図式 「マ」かその直後に「L」「R」「C」「E」の中
の同じものを並べたかであるもの。

452 数値図式 一般形は下記の通り（「#...」は非制約述句）：
～#... ～#... ～#... ～#...

453 地 書式制御用の「マ」「#」「マ」「マ」を除く制約記号の並
び。但し「マ」は空白と同等；また連系図式の直後
には「L」「R」「C」「E」のどれも使ってはならない。