漢文訓読のアルゴリズムとカッコつき表示

An algorithm of Kanbun reading
and
its parentheses expression

060910/060927/061103/061226

島野達雄
SHIMANO Tatsuo

１．はじめに

１－１．おことわり

はじめにおことわりしておきますが、ここでは、漢文というさまざまな要素をもった記述用言語のなかの、「漢字の並べ方」だけを対象にした議論をおこないます。それぞれの漢字の意味や役割はもちろんのこと、音読み、訓読みの区別、句読点の位置なども対象にしていません。レ点や一二点（一、二、三…）、上下点（上下、上中下）などの返り点[*)]のついている漢文の漢字列が、訓読したときに「どのように並べ替えられるか」。それだけの議論をおこないます。ここでの漢字は、単に「集合の要素」という意味しかもっていません。

１－２．訓読アルゴリズムとは

「正しく返り点がついているｎ個の漢字の列（原文漢字列とよぶ）は、訓読という操作をおこなったとき、つねに一定の返り点のないｎ個の漢字の列（最終訓読漢字列とよぶ）に並び替わる」ことは、どなたにもご理解いただけると思います。つまり訓読とは、それがすべてではありませんが、たしかに「ｎ個の漢字の列を並び替える操作」をふくんでおり、その操作には、厳密に定められた手順（アルゴリズム）があります。この操作手順を「訓読アルゴリズム」とよびます。ふつう、漢文の教科書には、返り点を次のように説明しています。

> 「レ点」は、下の一字からすぐ上に置かれた一字に返って読むことを示す符号である。…「返り点」はその文字や語句の左下（返読する語が二字以上の熟語の場合は中間に添えられる）につくきまりがある[**)]ので、訓読の基本としてはまず左下に返り点の添えられていない文字を上から順に読む（再読文字の初読の場合は返り点に関係なくこれに準じる）ことであり、「レ」や「二」の符号のついた文字に出会った場合にはその指示に従ってすぐ下の語や「一」の符号のついた語から返って読むことになる。　遠藤哲夫「入門指導」（『高等学校国語科教育研究講座・第十一巻漢文』、有精堂出版、昭和四十九年）

しかし、以下に述べるように、ことはそう簡単ではありません。

１－３．原文漢字列の基本条件

[*)] 061103 版まで「訓点」と称していたが、訓点は返り点と送り仮名の総称であり、061226 版から「返り点」に統一する。本稿の「返り点」には、レ点、一二点、上下点などのほか、漢字連結記号（ハイフン、竪点（たててん）とも）も含める。

[**)] このような「きまり」を明文化したものは、０６年１２月現在まだ発見していない。

じつのところ、議論の前提となる「正しく返り点がついているｎ個の漢字の列（原文漢字列）」を定義することは、容易ではありません。おおまかに説明すると、原文漢字列とは、有限個の漢字と返り点とが入り混じって直線的に並んでいる列であって、

（ⅰ）列の先頭は、漢字から始まる。返り点から始まっていない。
（ⅱ）漢字だけが並んでいる列でもよい。返り点をふくんでいなくてもよい。
（ⅲ）一二三点のばあい、三点のあとに二点があらわれ、二点のあとに一点があらわれる。
（ⅳ）上下点、天地点、甲乙点なども上のルールにもとづいている。
（ⅴ）上下点と一二点が混在するとき、下点・二点・上点・一点のように交差しない。かならず下点・二 点・一点・上点のように「入れ子構造」になっている。他のばあいも同様。
（ⅵ）レ点が列の最後にこない。

などの条件を満たしているものになります。

１－４．訓読操作の開始と進行方向

原文漢字列のｎ個の漢字を並べ替える操作は、まず、「先頭の漢字をどうするか」、具体的には、「最終訓読漢字列の先頭におくのか、あるいは、後回しにするのか」という判断から始まります。つまり、訓読操作は、先頭の漢字から始まります。この判断は、実際は、先頭の漢字だけではできないので、かならず、その次に存在する漢字ないし返り点をあわせて対象にします。その次に存在する漢字ないし返り点をあわせて対象にしたあとは、そのまた次に存在する漢字ないし返り点をあわせて対象にします。
このように、訓読操作は先頭の漢字に始まり、次に、さらにその次に、と進行します。
ご覧のように説明は、左横書きに統一しているので、ここでは、訓読操作の対象は、もっとも左側にある漢字に始まり、つねに右の方向に進行します。
ただし、ふつうは縦書きである漢文をイメージしやすいように、「左右」という表現は避け、「前後」「手前」「次」などと書くことにします。
なお、以下では、漢字をＡ、Ｂ、Ｃ…であらわします。

２．返り点はどの漢字についているか

２－１．一二点は直前の漢字についている

本論にはいるまえに、漢字と返り点の関係を述べておきます。
たとえば、

Ａ$_{\text{二}}$ＢＣ$_{\text{一}}$　　　　　　　　　　文例１

の二点と一点はどの漢字についているでしょうか。いうまでもなく、二点は漢字Ａに、一点は漢字Ｃについています。とくにここでの一点は、列（文、文章といってもよい）の最後に位置しているので、直前の漢字Ｃに付属していることは明らかです。
上点、甲点なども列の最後に置かれることがあるので、上下点（上下、上中下）、甲乙点（甲乙、甲乙丙…）、天地点（天地、天地人）なども、直前の漢字に付属していると言うことができます。

２－２．一二点、上下点、天地点などを総称して、順序点とよぶ

このように直前の漢字に付属し、訓読の順序を規定する、一二点・上下点・甲乙点などの返り点

を総称して、「順序点」とよびます。
順序点は、一般に、ｎ点、…２点、１点と書くことができます。
順序点への書き換えは、一二点なら、対応する漢数字を算用数字に置き換えるだけです。
上下点なら、下点が２点、上点が１点。上中下点なら、下点が３点、中点が２点、上点が１点にあたります。甲乙点、天地点なども、同様に、ｎ点、…２点、１点に書き換えることができます。
以下、同じ種類の順序点で、１点までつづく一連の順序点のなかの最大のものを「最大点」と呼びます。同種の順序点では、原文漢字列の基本条件（ⅲ）（ⅳ）にあるように、最大点が最初にあらわれます。文例１では、二点が最大点になっています。
なお、実際には存在しませんが、順序点の１点がついている、

　A_{1}

だけでも、「最大点が１点」の列とみなし、漢字Ａと同一視します。

２－３．レ点はうしろの漢字についている

一方、レ点は、列の最後に置かれることはありません。

　$\mathrm{A}_{\text{レ}}\mathrm{B}$

のように、レ点のあとには、かならず漢字があります。つまり、レ点はうしろの漢字に付属していると考えられます。正確に言えば、「レ点はその後の漢字一字に付属している」と考えても、以降の議論に矛盾を生じません。
江戸時代でも「レ点は後の漢字につくもの」と考えていたようで、版本では、改行時にレ点を前行の末尾に置くことはなく、必ず次行の先頭に置いています。

２－４．レ点と順序点の両方をもつ漢字

漢字のほうからみると、漢字一字で手前にレ点、うしろに順序点をもつことができます。

　$\mathrm{A}_{\text{レ}}\mathrm{B}_{\text{二}}\mathrm{C\,D}_{\text{一}}$　　　　　　　　　　文例２

このＢは、レ点と順序点（一二点の二点）の両方をもっています。
このような一字で二役をはたす漢字を「レ点順序点漢字」とよぶことにします。
レ点順序点漢字は、最大点の順序点をもつ漢字とはかぎらず、

　$\mathrm{A}_{\text{三}}\mathrm{B\,C}_{\text{レ}}\mathrm{D}_{\text{二}}\mathrm{E\,F}_{\text{一}}$　　　　　　　　　　文例３

のＤのようなケースもあります。この場合、Ｄを読んだあと、レ点が機能してＣを読み、その次にＤの持っている順序点の２点（二点）の働きで、３点（三点）をもつＡに返って読む約束になっています。
レ点順序点漢字のもつ順序点が１点のばあい、たとえば、

　$\mathrm{A}_{\text{二}}\mathrm{B\,C}_{\text{レ}}\mathrm{D}_{\text{一}}$

の返り点のつけ方は、「間違い」とは言い切れませんが、次項の文例４のように順序点とレ点を連続してつけるのが一般的なようです。

２－５．順序点とレ点の連続

返り点のなかには、

　$\mathrm{A}_{\text{二}}\mathrm{B\,C}_{\text{一レ}}\mathrm{D}$　　　　　　　　　　文例４

の漢字Ｃと漢字Ｄのあいだにあるように、一点とレ点が複合したようにみえる返り点（縦書きでは「一の下にレ」）もあります。
このような返り点は、完全に合体しているわけではなく、それぞれが別々に前後の漢字に付属し

ていると考えられます。さきに述べたように、この文例の「一点は手前のＣに」、「レ点はうしろのＤに」ついています。
この文例の訓読操作を見てみると、（以下の「読む」は「最終訓読漢字列を配置する途上の、返り点や漢字の列、つまり訓読漢字列に置く」と同義です）
（１）二点をもつＡは、一点をもつＣより後に読む。つまり、ＣはＡよりさきに読む。
（２）次にあらわれるＢには返り点がないので、最初に読む。
（３）Ｄはレ点をもつので、直前のＣよりさきに読む。
です。この三つのことから、Ｂが最初、ＤはＣよりさき、ＣはＡよりさきに読む、すなわち最終的な漢字の列（最終訓読漢字列）はＢＤＣＡになります。一点とレ点が複合したようにみえる返り点が、べつべつに前後の漢字に付属している、と解釈できることがおわかりいただけると思います。
おなじように、上点とレ点の複合したようにみえる返り点（縦書きでは「上の下にレ」）も、「上点は直前の漢字に」ついており、「レ点は直後の漢字に」ついている、と解釈できます。
【補題１：順序点とレ点の連続は１点で発生する】***)
順序点とレ点の連続は、１点とレ点のときのみ発生する。
〔証明〕
なぜなら、ｋが２以上のとき、

$\cdots A_{k\text{レ}} B \cdots C_{k-1} \cdots$

と書いてあれば、Ａよりさきに読むのは、ＢなのかＣなのか、あいまいになる。（証明終わり）
この証明でのべた、

$\cdots A_{k\text{レ}} B \cdots C_{k-1} \cdots$

のようなケースは、次のように、Ｂをレ点順序点漢字として書くのが「正しい」ようです。

$\cdots A_{\text{レ}} B_{k} \cdots C_{k-1} \cdots$

２－６．熟語を示す漢字連結記号

なお、広い意味の返り点のひとつとして、熟語を示す「漢字連結記号」があります。漢字連結記号は、熟語や固有名詞などを示すために、二つの漢字を短い線分でつないだもので、英語のハイフンー　に相当します。たとえば、

Ａ－$_{\text{二}}$ＢＣＤ$_{\text{一}}$　　　　文例５

では、ＡとＢのあいだに二点と漢字連結記号がならんでおり、訓読のとき「ＡＢで一つの熟語である。Ａだけを返読するのではない。ＤのつぎはＡＢをまとめて読む」という意味がこめられています。つまり、漢字連結記号でつながれた漢字Ａと漢字Ｂは、一つの漢字のようにあつかえます。この文例では、Ａ－Ｂがひとまとまりで、順序点の二点をもっている、と解釈できます。
Ａ－Ｂ－Ｃのように、三つ、四つ…の漢字を漢字連結記号でつないだものも同様です。
漢字連結記号は広い意味での返り点のひとつですが、「二つの漢字を一つにみなす」だけの働きをしています。原文漢字列にＡ－Ｂがあるばあい、最終訓読漢字列はかならず（ＡとＢのあいだに他の漢字が入らない）ＡＢになります。
ちなみに、レ点とこの漢字連結記号とは共存せず、『漢字連結記号でつかう返り点は一二点とす

***) この補題１自体を、証明を要しない「正しい返り点のつけ方の約束事」つまり本稿の冒頭で述べた「原文漢字列の基本条件」としておいたほうがよいかもしれない。ただし、１点以外で順序点とレ点の連続がある例もある。たとえば、長澤規矩也編・和刻本正史の『三国志』呉書には、「…使$_{\text{二レ}}$ＡＢ$_{\text{一}}$ 」という使役形式の用例がしばしば登場し、「ＡをしてＢせしむ」または「ＡをＢせしむ」と読んでいる。この用例の読み下し文は、レ点を使わないで「…使$_{\text{二}}$ＡＢ$_{\text{一}}$ 」としても得ることができる。

る』ことが漢文の基本ルールとしてあるようです。Ａ㆑Ｂ－ＣはＡ㆓ＢＣ㆒と、Ａ－㆑ＢＣやＡ－Ｂ㆑Ｃは、Ａ－㆓ＢＣ㆒と書くのが「正しい」ようです。

２－７．漢字は四種類に分類できる

以上からわかるように、返り点とは、漢字連結記号でつながれた二つ以上の漢字を一つの漢字とみなせば****)、漢字のうしろにつく順序点と、漢字の前にあるレ点の二つしか存在しません。
つぎの文例には、一二点の順序点と二つのレ点があります。

　　Ａ㆑Ｂ㆓ＣＤ㆒㆑Ｅ　　　　　　　　　　文例６

ここで、レ点も順序点ももたない漢字を「無点漢字」、両方をもつ漢字を「レ点順序点漢字」、順序点だけをもつ漢字を「順序点漢字」、レ点だけをもつ漢字を「レ点漢字」とよぶことにすれば、正しく返り点がつけられた漢字の列（原文漢字列）は、

・レ点と順序点のどちらももたない漢字	（無点漢字）	上の文例６ではＡとＣ
・レ点と順序点の両方をもつ漢字	（レ点順序点漢字）	上の文例６ではＢ
・順序点だけをもつ漢字	（順序点漢字）	上の文例６ではＤ
・レ点だけをもつ漢字	（レ点漢字）	上の文例６ではＥ

の四種類の漢字で構成されています。
なお、以下では、付属するレ点や順序点をふくめて、レ点漢字、順序点漢字、レ点順序点漢字とよぶことがあります。

３．閉じた部分列と標準順序点列の返読

３－１．前方に開いている列

たとえば、すこし複雑ですが、

　　Ａ㆑Ｂ㆘Ｃ㆔ＤＥ㆓ＦＧ㆒Ｈ㆑ＩＪ㆑Ｋ㆗Ｌ㆑Ｍ㆓ＮＯ㆒Ｐ㆖　　　　文例７

という文例のなかの、無点漢字Ａだけを取り除いた、レ点順序点漢字Ｂから始まる、

　　㆑Ｂ㆘Ｃ㆔ＤＥ㆓ＦＧ㆒Ｈ㆑ＩＪ㆑Ｋ㆗Ｌ㆑Ｍ㆓ＮＯ㆒Ｐ㆖

という部分列は、これだけでは、「正しく返り点がついている漢字列（原文漢字列）」にはなりません。原文漢字列の基本条件の「（ｉ）列の先頭は、漢字から始まる。返り点から始まっていない」、つまり、原文漢字列はレ点から始らない、という条件に反するからです。
おなじく、後半の

　　㆑ＩＪ㆑Ｋ㆗Ｌ㆑Ｍ㆓ＮＯ㆒Ｐ㆖

も、レ点漢字Ｉのレ点からはじまるので、「正しく返り点がついている漢字列（原文漢字列）」とはみなせません。
レ点から始まる部分列は、「前方に開いている」部分列といえます。

３－２．後方に開いている部分列

同じ文例７の、先頭の漢字Ａとレ点との列、

****) 漢字Ａ一字を[Ａ]、漢字連結記号で結ばれたＡ－Ｂを[ＡＢ]、Ａ－Ｂ－Ｃを[ＡＢＣ]、Ａ－Ｂ－Ｃ－Ｄを[ＡＢＣＤ]などと[　]を使って表記すれば、漢字一字と漢字連結記号で結ばれた二字以上の漢字を一括して扱うことができる。漢字列をｆ、ｇとしたとき、ｇの前にレ点がなければ、[ｆ][ｇ]＝[ｆｇ]が成り立つ。コンピュータで扱う場合は、この[　]表示を取り入れたほうが便利かもしれない。

Ａ$_{\text{レ}}$

はレ点をもつ漢字が存在せず、また、

Ａ$_{\text{レ}}$Ｂ$_{\text{下}}$

は、レ点順序点漢字Ｂの下点（最大点）に対応する上点がないので、「正しく返り点がついている」とはいえません。

Ｃ$_{\text{三}}$ＤＥ$_{\text{二}}$Ｆ

も同様に、一点の漢字がないので、原文漢字列にはなりません。
以上三つの、返り点と漢字の列は、「後方に開いている」部分列といえます。

３－３．開いた部分列とは

このような、前方に開いている部分列、後方に開いている部分列、前方にも後方にも開いている部分列をまとめて、原文漢字列のなかの「開いた部分列」とよぶことにします。
開いた部分列は、正しく返り点のついた原文漢字列から切り取った、連続した返り点と漢字の列ですが、「単独では原文漢字列にならない列」とも言い換えられます。

３－４．閉じた部分列の定義

いっぽう、同じ文例７のなかの、

Ｃ$_{\text{三}}$ＤＥ$_{\text{二}}$ＦＧ$_{\text{一}}$

は、これだけで原文漢字列とみなせます。前にもうしろにも「開いていません」。
また、

Ｋ$_{\text{中}}$Ｌ$_{\text{レ}}$Ｍ$_{\text{二}}$ＮＯ$_{\text{一}}$Ｐ$_{\text{上}}$

の中点を順序点２点、上点を順序点１点とみなすと、

Ｋ$_{2}$Ｌ$_{\text{レ}}$Ｍ$_{\text{二}}$ＮＯ$_{\text{一}}$Ｐ$_{1}$

と書くことができ、やはり、前方にも後方にも開いていない、原文漢字列とみなせます。

Ｈ$_{\text{レ}}$Ｉ

には、レ点がありますが、これも原文漢字列とみなせます。
さらに、順序点の定義で、「最大点を１点とする漢字はその漢字と同一視する」としたので、

ＦＧ$_{\text{一}}$

は、漢字Ｆと、１点を最大点とする漢字Ｇ、からなる原文漢字列です。

Ｐ$_{\text{上}}$

も、１点を最大点とする漢字Ｐ一字、とみなせます。
このように、原文漢字列のミニチュアとも言える、「正しく返り点がついている」部分列を、「閉じた部分列」とよぶことにします。
閉じた部分列の定義は、

（１）原文漢字列のなかの、連続した、返り点や漢字の列である。（漢字一字でも可）
（２）レ点から始まっていない。（無点漢字か順序点漢字で始まる）
（３）列のなかに、順序点（レ点順序点漢字のもつ順序点でもよい）があるときは、対応する１点が列内にある。（1 点をもつ漢字だけからなる列でもよい）
（４）レ点が列の最後にこない。

になります。
閉じた部分列の先頭は、からなず無点漢字か順序点漢字のどちらかで、最後尾は、無点漢字、（レ点が前についている）レ点漢字、１点がついた順序点漢字のいずれかです。
２点以上の順序点をもつレ点順序点漢字が最後尾にあるとすると、【補題１：順序点とレ点の連

続は１点で発生する】により、対応する１点をもった順序点漢字が列内に存在しないので、部分列の定義（３）に反します。

３－５．標準順序点列の定義

なお、言うまでもありませんが、原文漢字列のなかにある、同じ種類の順序点で、最大点のｋ点（ｋ≧１。１点を最大点とすることができる）をもつ順序点漢字から、１点をもつ順序点漢字にいたる、

$P_{k}\cdots Q_{k-1}\cdots R_{1}$

という列は、閉じた部分列です。
以下では、このような「原文漢字列のなかにある、同じ種類の順序点で、最大点をもつ順序点漢字から、対応する１点をもつ順序点漢字にいたる」列を「標準順序点列」とよびます。ただし、ここでの順序点漢字は、漢字連結記号でむすばれた二字以上の漢字を漢字一字と見なしているものを除きます。
標準順序点列の内部に、別の種類の順序点をもつレ点順序点漢字があってもかまいませんが、大枠となる、同じ種類のｋ点から１点までの順序点をもっている漢字は、順序点漢字にかぎります。

３－６．閉じた部分列を結合しても閉じた部分列になる

これからは、閉じた部分列を、ｆ、ｇ、ｈ…とあらわすことにします。
【補題２：二つの閉じた部分列を結合しても閉じた部分列になる】
原文漢字列で、連続する二つの閉じた部分列をｆ、ｇとすると、結合した列ｆｇも閉じた部分列になる。
〔証明〕
ｆとｇは、それぞれさきほどのべた部分列の定義（１）～（４）を満たしている。
よって、ｆｇは、
（１）原文漢字列のなかの、連続した、返り点や漢字の列である。
（２）ｆがレ点から始まっていないので、レ点から始まっていない。
（３）ｆとｇはそれぞれ「列のなかに順序点があるときは、対応する１点が列内にある」ので、ｆｇの列のなかに順序点があるときは、対応する１点がｆｇの列内にある。
（４）ｇの最後は、レ点ではないので、レ点が列の最後にこない。
以上から、ｆｇは閉じた部分列である。（証明終わり）
この補題２の例を示すと、

$A_{\text{レ}}B_{\text{下}}C_{\text{三}}D\,E_{\text{二}}F\,G_{\text{一}}H_{\text{レ}}I\,J_{\text{レ}}K_{\text{中}}L_{\text{レ}}M_{\text{二}}N\,O_{\text{一}}P_{\text{上}}$　　文例７（再掲）

の、閉じた部分列

$C_{\text{三}}D\,E_{\text{二}}F\,G_{\text{一}}$

と、閉じた部分列

$H_{\text{レ}}I$

を結合した、

$C_{\text{三}}D\,E_{\text{二}}F\,G_{\text{一}}H_{\text{レ}}I$

も閉じた部分列です。

３－７．１つちがいの順序点漢字のあいだは閉じた部分列

ここで、「連続した同種の順序点をもつ、二つの順序点漢字のあいだにある部分列は、閉じた部

分列である」ことを証明します。
この補題は、要するに、標準順序点列、

$A_k \cdots B_i \cdots C_{i-1} \cdots D_1$ （ｉ≧２）

の、漢字Ｂと漢字Ｃのあいだにある「…」部分の、返り点や漢字の列は、閉じた部分列であることを言っています。
なお、標準順序点列の定義で述べたように、Ｂは漢字連結記号で結ばれた二字以上の漢字ではないので、ＢとＣのあいだの「…」部分には少なくとも一つ漢字が存在します。二字以上の漢字が漢字連結記号でつながれているばあい、

$E—_{三}FG_{二}\cdots$

のように、Ｅ－ＦとＧのあいだには漢字が存在しないことがあります。
【補題３：１つちがいの順序点漢字のあいだは閉じた部分列】
ｋ点（ｋ≧２。ここではｋ＝1のとき、つまり１点を最大点とするケースを論議の対象にしない）を最大点とし、同じ種類の１点まで、一連の順序点をもつ順序点漢字からなる、標準順序点列を、

$A_k \cdots B_i \alpha C_{i-1} \cdots D_1$ （ｉ≧２）

とする。
ここで順序点漢字Ｂと順序点漢字Ｃにはさまれた、連続した返り点や漢字の列αは、閉じた部分列である。
〔証明〕
αがレ点から始まっていないことは、Ｂの順序点が２点以上で、【補題１：順序点とレ点の連続は１点で発生する】ことから明らか。
αのなかに、ｋ点～１点以外の「別の種類の順序点」があるとすると、本稿の冒頭でのべた原文漢字列の条件の、

（ⅲ）一二三点のばあい、三点のあとに二点があらわれ、二点のあとに一点があらわれる。
（ⅳ）上下点、天地点、甲乙点なども上のルールにもとづいている。

という条件から、この標準順序点列（標準順序点列は閉じた部分列であるので、原文漢字列とみなせる）のなかには、「別の種類の順序点」の１点がかならず存在する。ここで、もしα内に「別の種類の順序点」の１点がない、とすると、その１点はＣよりあとに存在することになり、原文漢字列の条件の、

（ｖ）上下点と一二点が混在するとき、下点・二点・上点・一点のように交差しない。かならず下点・二 点・一点・上点のように「入れ子構造」になっている。他のばあいも同様。

に反する。
よって、「別の種類の順序点」がα内にあるとすると、その１点はα内にある。
また、αはレ点で終わっていない。
なぜなら、Ｃは標準順序点列の定義から順序点漢字であり、前方にレ点をもたない。
以上、閉じた部分列の定義（１）～（４）を満たしているので、αは閉じた部分列といえる。（証明終わり）

３－８．標準順序点列から先頭の順序点漢字をとりのぞく

たとえば、４点を最大点とする標準順序点列、

$A_4\, f\, B_3\, g\, C_2\, h\, D_1$

があるとします。ｆ、ｇ、ｈは、【補題３：１つちがいの順序点漢字のあいだは閉じた部分列】から、閉じた部分列です。
この列からA_4を取り除いた、

$f B_3 g C_2 h D_1$
さらに、$f B_3$を取り除いた、
$g C_2 h D_1$
さらに、$g C_2$を取り除いた、
$h D_1$
の、三つの部分列がすべて閉じた部分列であることを、次の補題は主張しています。
【補題４：標準順序点列から先頭の順序点漢字を取り除いた列は閉じた部分列】
最大点のｋ点（$k \geqq 2$）をもつ順序点漢字Ｐから始まり、対応する１点をもつ順序点漢字Ｒまでの、一連の順序点漢字からなる、標準順序点列、
$P_k f Q_{k-1} \cdots R_1$
から、P_kをのぞいた、
$f Q_{k-1} \cdots R_1$
は、閉じた部分列になる。
〔証明〕
なぜなら、【補題３：１つちがいの順序点漢字のあいだは閉じた部分列】から、ｆは閉じた部分列。また、
$Q_{k-1} \cdots R_1$
は、$k-1$点を最大点とする順序点漢字Ｑから始まり、対応する１点をもつ順序点漢字Ｒまでの、一連の順序点漢字からなっているので、標準順序点列。標準順序点列は、定義から、閉じた部分列である。
【補題２：二つの閉じた部分列を結合しても閉じた部分列になる】から、二つを結合した、
$f Q_{k-1}$・・・R_1
も閉じた部分列になる。（証明終わり）
この【補題４：標準順序点列から先頭の順序点漢字を取り除いた列は閉じた部分列】を繰り返して使うと、
$A_4 f B_3 g C_2 h D_1$
のA_4の次からD_1にいたる、
$f B_3 g C_2 h D_1$
も、さらに（ここから$f B_3$を取り除いた）B_3の次からD_1にいたる、
$g C_2 h D_1$
も、さらに（ここから$g C_2$を取り除いた）C_2の次からD_1にいたる、
$h D_1$
も、すべて、閉じた部分列になります。

３－９．閉じた部分列が意味するもの

閉じた部分列は、原文漢字列のミニチュアと言ってもよく、「正しく返り点がついている」漢字の列です。単独でも、訓読したとき、つねに一定の、返り点のない漢字の列に並び替ります。ひらたくいえば、「閉じた部分列は、原文漢字列のなかで、それだけで読むことができる」返り点や漢字の列です。
閉じた部分列は、いわば原文漢字列という大宇宙のなかにある小宇宙です。
その小宇宙のなかに、さらに閉じた部分列の微小宇宙がいくつも存在することもあります。
しかし、閉じた部分列は、なかにどんなに微小宇宙が存在しても、つねに一定の、返り点のない漢字の列に並び変わります。
いっぽう、開いた部分列は、単独で並べ替えることができません。前後にレ点をもっていたり、

一二点の二点で終わっていたりするので、部分列の前後にある漢字と順番が入れ替わります。
話を順序点漢字だけに限ると、【補題４：標準順序点列から先頭の順序点漢字を取り除いた列は閉じた部分列】の結果から、「ｉ点をもつ順序点漢字の次の漢字」に始まり、「最後の１点をもつ順序点漢字」までの漢字列は、閉じた部分列になります。
【補題４】は、その漢字列だけで、「原文漢字列と同じようにあつかうことができる」「読むことができる」「漢字の配置が可能である」ことを保証しています。

３－１０．Ｐはｍから返って読む

たとえば、４点を最大点とする、標準順序点列、

$A_4 f B_3 g C_2 h D_1$

の訓読操作の概略は、（以下の「読む」は「訓読途上の訓読漢字列に置く」と同義です）
（１）４点をもつ順序点漢字Ａよりさきに、$f B_3 g C_2 h D_1$という閉じた部分列を読む。
（２）３点をもつ順序点漢字Ｂよりさきに、$g C_2 h D_1$という閉じた部分列を読む。
（３）２点をもつ順序点漢字Ｃよりさきに、ｈＤという閉じた部分列を読む。（D_1はＤと同一視）です。
ここで、「ｉ点（ｉ≧２）をもつ順序点漢字Ｐから始まる標準順序点列において、Ｐの後方にある、対応する１点をもつ順序点漢字までの閉じた部分列をｍとするとき、もとの標準順序点列を

Ｐ（ｍ）

と、カッコをつけて書く」と定義します。
すると、上の（１）（２）（３）からわかるように、「訓読操作をおこなう」とは、要は「順序点漢字Ｐより先に閉じた部分列ｍを読む」ことなので、Ｐ（ｍ）は、
新しい漢字返り点の列

ｍＰ

となります。つまり、

Ｐ（ｍ）＝ｍＰ

という式が成り立ちます。
この表記と式を繰り返し適用すると、まず（１）は、

$A(f B_3 g C_2 h D_1)$

とあらわせます。このカッコ内にある、つぎの（２）の

$B_3 g C_2 h D_1$

は、

$B(g C_2 h D_1)$

とあらわせます。さらに、このカッコ内にある、つぎの（３）

$C_2 h D_1$

は、

$C(h D)$　　　　（D_1はＤと同一視）

とあらわせます。
これらを、順に代入すると、

$A_4 f B_3 g C_2 h D_1$
$=A(f B_3 g C_2 h D_1)$
$=A(f B(g C_2 h D_1))$
$=A(f B(g C(h D)))$

とあらわせます。
要は、すべての順序点の位置に、「(　」をつけ、対応する「　)」は、それらに対応する順序点

１点の位置につけることになります。
さらに、

Ｐ（ｍ）＝ｍＰ

を利用し、カッコをはずしてゆくことができます。

$A_4 f B_3 g C_2 h D_1$
＝Ａ（ｆＢ（ｇＣ（ｈＤ）)）　　カッコつき表示
＝ｆＢ（ｇＣ（ｈＤ））Ａ　　ＡをＰ、ｆＢ（ｇＣ（ｈＤ ）をｍとみなす。
＝ｆｇＣ（ｈＤ ）ＢＡ　　ＢをＰ、ｇＣ（ｈＤ ）をｍとみなす。
＝ｆｇｈＤＣＢＡ　　ＣをＰ，ｈＤをｍとみなす。

３－１１．標準順序点列の返読定理

【定理１：標準順序点列の返読定理】
標準順序点列、

$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$　（ｎ≧２）

において、順序点漢字Ｐと順序点ｎを取り除いた、

$f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$

をｍとおき、この標準順序点列をＰ（ｍ）と表記する。

Ｐ（ｍ）＝ｍＰ

と定義すると、

$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$
＝ｆｇ…ｈＴＳ…ＲＱＰ

が成り立つ。
〔証明〕
もとの標準順序点列

$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$

の部分列で、順序点漢字を先頭にもつ、

$Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$
$R_{n-2} \cdots\cdots S_2 h T_1$
$S_2 h T_1$

などは、すべて標準順序点列になっているので、

Ｐ（ｍ）＝ｍＰ

の、Ｐ（ｍ）という表記および式を繰り返して適用することができる。よって、

$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$
$= P(f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1)$　　ｍの定義による。
$= P(f Q(g R_{n-2} \cdots\cdots S_2 h T_1))$　　Ｑ以下にＰ（ｍ）の表記を適用。
$= P(f Q(g R(\cdots\cdots S_2 h T_1)))$　　Ｒ以下にＰ（ｍ）の表記を適用。
＝Ｐ（ｆＱ（ｇＲ（……Ｓ（ｈＴ））））　　Ｓ以下にＰ(ｍ)の表記を適用。T_1はＴと同一視。
＝ｆＱ（ｇＲ（……Ｓ（ｈＴ）））Ｐ　　Ｐ（ｍ）＝ｍＰの定義による。
＝ｆｇＲ（……Ｓ（ｈＴ））ＱＰ　　Ｑ（ｉ）＝ｉＱより。
＝ｆｇ…Ｓ（ｈＴ）…ＲＱＰ　　Ｒ（ｋ）＝ｋＲより。
＝ｆｇ…ｈＴＳ…ＲＱＰ　　Ｓ（ｊ）＝ｊＳより。

（証明終わり）（注：厳密には、この証明は数学的帰納法でおこなう）

３－１２．標準順序点列の返読定理の意味

この定理は、標準順序点列、

$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$　（$n \geqq 2$）

の最大点である順序点ｎ点のところに「(　」を、順序点１点のうしろに「　)」をつけた

$P(f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1)$

というカッコつき表示を繰り返し適用すると、もとの標準順序点列は、順序点のない

$P(fQ(gR(\cdots\cdots S(hT))))$

というカッコつき表示に置き換わり、

$P(m) = mP$

という式をくりかえし使えば、最後に

$fg \cdots hTS \cdots RQP$

という列に置き換わることを示しています。

この列は、まさしくもとの標準順序点列を「訓読した列」になっています。閉じた部分列ｆ、ｇ、ｈはレ点をふくんだり、入れ子構造の下位に属するほかの標準順序点列などをふくんだりすることもあるので、この列は最終訓読漢字列ではありませんが、訓読途上の訓読漢字列にはなっています。

さらに言い換えれば、標準順序点列では、

$P(m) = mP$

つまり、「Ｐはｍから返って読む」「Ｐはｍのあとに配置する」という、カッコつき表示を取り入れるだけで、一二点、上下点、甲乙点などの、すべての順序点をカッコつき表示に置き換えることができ、式の変形という機械的操作で訓読漢字列が得られることを、この標準順序点列の返読定理は示しています。

４．一般順序点列の返読

４－１．順序点の概念を拡張する

無点漢字Ａに、レ点漢字Ｂだけがつづいている、

$A_{レ}B$

は、

$A_2 B_1$

と番号を振ることが可能です。

さらに、レ点漢字Ｂにレ点漢字Ｃがつづいている、

$A_{レ}B_{レ}C$

には、

$A_3 B_2 C_1$

と番号をあたえることができます。

一般に、無点漢字Ａから始まり、レ点漢字が連続している漢字列（無点レ点連続列とよぶ）、

$A_{レ}B_{レ}C \cdots_{レ}D_{レ}E$　　　　　　　　　　　文例８

は、最後尾にあるレ点漢字（ここではＥ）から順に返って読むので、

$A_n B_{n-1} C_{n-2} \cdots D_2 E_1$

と、漢字のうしろに番号を振ることができます。１以外の番号は、ちょうどもとのレ点の位置にあることをここで注意しておきます（注：のちほどのべる「一般順序点列のカッコのつけ方」を参照）。

これまで、順序点は「一二点、上下点、甲乙点などを総称したもの」としてきましたが、これからは、このような無点レ点連続列につけた番号も、新しい順序点としてあつかうことにします。

４－２．あらゆるレ点は順序点とみなせる

レ点がついているレ点漢字やレ点順序点漢字は、その前にある漢字よりもさきに読む約束なので、無点レ点連続列で示したように、レ点の前後の漢字に番号を振ることができます。
レ点漢字の前に無点漢字があるケースが、無点レ点連続列です。
レ点順序点漢字Ｂの前に無点漢字Ａがある、
　$A_{\text{レ}}B_{\text{二}}CD_{\text{一}}$　　文例２（再掲）
では、ＢとＤの順序点をそのままにして、Ａに新しく順序点３をあたえ、
　$A_{3}B_{2}CD_{1}$
と書き換えることができます。
一般に、レ点の前に無点漢字Ａがある、
　$\cdots A_{\text{レ}}B\cdots$
のばあい、Ｂの新しい順序点がｉ点なら、Ａには新しい順序点１＋ｉ点をあたえます。
レ点の前がレ点漢字である、
　$\cdots_{\text{レ}}A_{\text{レ}}B\cdots$
も同様に、Ｂの新しい順序点がｉ点であれば、Ａの新しい順序点は１＋ｉ点になります。
レ点の前に順序点（【補題１：順序点とレ点の連続は１点で発生する】から、この順序点は１点です）があっても同様に番号をつけ、新しい順序点をあたえることができます。。
たとえば、
　$A_{\text{三}}\cdots B_{\text{二}}\cdots C_{\text{一レ}}D_{\text{レ}}E$
のＥに１、Ｄに２、Ｃに１＋２＝３、Ｂに２＋２＝４、Ａに３＋２＝５と、番号をあたえ、新しい順序点をつけることができます。結果は次のようになります。
　$A_{5}\cdots B_{4}\cdots C_{3}D_{2}E_{1}$
この例のように、レ点の前に１点をもつ順序点漢字があれば、対応する最大点までの順序点をもつ漢字の順序点もつけかえます。
一般に、標準順序点列のうしろにレ点がある、
　$P_{k}\cdots Q_{1\text{レ}}A\cdots$
のようなばあい、Ａの番号がｉであれば、Ｑの順序点は１＋ｉ点、Ｐの順序点はｋ＋ｉ点につけかえることができます。つまり、
　$P_{k+i}\cdots Q_{1+i}A_{i}\cdots$
になります。
以上、レ点漢字にせよレ点順序点漢字にせよ、レ点の前にあるのは無点漢字か、レ点漢字か、１点の順序点しかないので、『あらゆるレ点は順序点とみなせる』ことがわかります。
このようにして、レ点漢字やレ点順序点漢字はもちろん、無点漢字、順序点漢字に新しくつけかえた順序点を、「一般順序点」とよぶことにします。
一般順序点も、これまでの順序点と同じように、ｎ点、ｎ－１点、…２点、１点とあらわせます。
【補題５：原文漢字列のあらゆるレ点と順序点は一般順序点におきかえることができる】
〔証明〕（省略）
なお、一般順序点のなかにも、「レ点だけを置き換えたもの」「一二点だけのもの」「一二点とレ点の結合を置き換えたもの」「上下点だけのもの」「上下点とレ点の結合を置き換えたもの」といった、異なった種類の一般順序点があることを注意しておきます。

４－３．一般順序点の例

念のため、レ点や順序点を一般順序点につけかえた例をいくつかご紹介します。
（１）
標準順序点列のうしろに、レ点漢字Ａ、そのうしろにレ点漢字Ｂ、そのうしろにレ点漢字Ｃ…とレ点漢字がｎ個つづいている、

$$P_{k}\cdots Q_{1\text{レ}}A_{\text{レ}}B_{\text{レ}}C\cdots_{\text{レ}}D$$

は、

$$P_{k+n}\cdots Q_{1+n}A_{n}B_{n-1}C_{n-2}\cdots D_{1}$$

と、もとの標準順序点列の順序点をつけかえ、レ点漢字Ａ、Ｂ、Ｃ、…Ｄには、ｎ点、ｎ－１点、ｎ－２点、…１点をあたえると、一般順序点をつけることができます。
漢字Ｑと漢字Ａにはさまれていた、もとの１点とレ点は、同じ位置に、新しい一般順序点１＋ｎ点になっています。１点とレ点が複合したようにみえる返り点は、一般順序点に書き換えたとき、一つの点になることをここで注意しておきます（注：のちほど述べる「一般順序点列のカッコのつけ方」を参照）。
（２）
標準順序点列のうしろに、レ点順序点漢字がつくこともあります。

$$P_{k}\cdots Q_{1\text{レ}}A_{j}\cdots B_{1}$$

この漢字列では、Ａが最大点ｊ点をもつ、レ点順序点漢字です。
この場合も、Ｐの順序点をｋ＋ｊ点、Ｑの順序点を１＋ｊ点とつけかえれば、

$$P_{k+j}\cdots Q_{1+j}A_{j}\cdots B_{1}$$

と一般順序点をもつ漢字列に生まれ変わります。
ここでも、漢字Ｑと漢字Ａにはさまれていた、もとの１点とレ点は、同じ位置の、新しい一般順序点１＋ｉ点になっています。
（３）
無点漢字Ｐで始まり、レ点漢字Ｑで終わる漢字列、つまりｋ個のレ点漢字をもつ、無点レ点連続列のうしろに、レ点順序点漢字がつづくケースもあります。下のＡが最大点ｊ点をもつレ点順序点漢字です

$$P\cdots_{\text{レ}}Q_{\text{レ}}A_{j}\cdots B_{1}$$

これも、これまでとまったく同様に、一般順序点をあたえることができます。

$$P_{k+j}\cdots Q_{1+j}A_{j}\cdots B_{1}$$

このケースでは、もとのレ点や順序点と同じ位置に、新しい一般順序点がつくことを注意しておきます（注：のちほど述べる「カッコのつけ方」を参照）。
（４）
なお、レ点漢字やレ点順序点漢字を組み合わせると、

$$P_{k}\cdots P_{1\text{レ}}Q_{\text{レ}}R_{\text{レ}}S_{n}\cdots S_{1\text{レ}}T_{m}\cdots T_{1\text{レ}}\cdots$$

のように、えんえんと後方につづく漢字列が考えられますが、『原文漢字列の漢字は有限個』なので、いつかは「うしろにレ点がつづかない漢字」で終わります。この漢字に一般順序点１点を割り当てれば、すべての漢字に一般順序点を付加することができます。

４－４．一般順序点列の定義

【補題５：原文漢字列のあらゆるレ点と順序点は一般順序点におきかえることができる】により、原文漢字列にふくまれる、あらゆるレ点と順序点を一般順序点におきかえます。すると、すべてのレ点が解消されたわけですから、原文漢字列には、一般順序点をもつ漢字（いうなれば「一般

順序点漢字」）と無点漢字の二種類の漢字しかありません。
レ点から一般順序点におきかわった部分では、一般順序点ｉ点をもつ漢字と、一般順序点ｉ－１点をもつ漢字が連続しています。
そこで、標準順序点列の概念を拡張し、

$$P_k f Q_{k-1} g R_{k-2} \cdots S_3 \ h T_1 \qquad (k \geqq 2)$$

のｆ、ｇ、ｈなどは、「返り点（ここでは一般順序点のみ）や漢字の存在しない空（くう）の部分列φ（ファイ）であってもよい」と解釈します。
空の部分列φは、訓読操作では「何も置かないで、つぎの訓読操作に進む」ことを意味しています。何もしないことは、広い意味では閉じた部分列とみなせます。
このように、「連続する一般順序点をもつ二つの漢字のあいだに空の部分列φがあってもよい、と認めた、同じ種類の一般順序点の、最大点ｋ点をもつ漢字に始まり、１点をもつ漢字にいたる、一連の漢字列」を、「一般順序点列」とよぶことにします。
なお、標準順序点列の定義で注意しましたが、

$$A - {}_{二} B C_{一}$$

も、漢字連結記号で結ばれた二点をもつＡ－Ｂおよび一点をもつ漢字Ｃのあいだに、空の部分列φが存在していると解釈できるので、このような列も一般順序点列にふくまれます。

４－５．一般順序点列の返読定理

【定理２：一般順序点列の返読定理】
ｆ、ｇ、ｈが空の部分列φであることもある、一般順序点列、

$$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1 \qquad (n \geqq 2)$$

において、漢字Ｐと一般順序点ｎを取り除いた、

$$f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$$

をｍとおき、この標準順序点列をＰ（ｍ）と表記する。

$$P(m) = mP$$

と定義すると、

$$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$$
$$= f\ g \cdots h T S \cdots R Q P$$

が成り立つ。
〔証明〕
もとの一般順序点列

$$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$$

の部分列で、一般順序点をもつ漢字から始まる、

$$Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$$
$$R_{n-2} \cdots\cdots S_2 h T_1$$
$$S_2 h T_1$$

などは、すべて一般順序点列になっているので、

$$P(m) = mP$$

の、Ｐ（ｍ）という表記および式を繰り返して適用することができる。よって、

$$P_n f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1$$
$= P(f Q_{n-1} g R_{n-2} \cdots\cdots S_2 h T_1)$　　ｍの定義による。
$= P(f Q(g R_{n-2} \cdots\cdots S_2 h T_1))$　　Ｑ以下にＰ（ｍ）の表記を適用。
$= P(f Q(g R(\cdots\cdots S_2 h T_1)))$　　Ｒ以下にＰ（ｍ）の表記を適用。
$= P(f Q(g R(\cdots\cdots S(h T))))$　　Ｓ以下にＰ（ｍ）の表記を適用。T_1はＴと同一視。

　＝ｆＱ（ｇＲ（……Ｓ（ｈＴ)））Ｐ　　　Ｐ（ｍ）＝ｍＰの定義による。
　＝ｆｇＲ（……Ｓ（ｈＴ)）ＱＰ　　　　Ｑ（ｉ）＝ｉＱより。
　＝ｆｇ…Ｓ（ｈＴ）…ＲＱＰ　　　　　Ｒ（ｋ）＝ｋＲより。
　＝ｆｇ…ｈＴＳ…ＲＱＰ　　　　　　　Ｓ（ｊ）＝ｊＳより。

（証明終わり）（注：厳密には、この証明は数学的帰納法でおこなう）
この定理２は、【定理１：標準順序点列の返読定理】の「標準順序点列」を「一般順序点列」に、「順序点」を「一般順序点」におきかえただけのものですが、あらゆるレ点をふくむ漢字列にも、カッコつき表示が適用でき、機械的な式の変形で訓読漢字列が得られることを示しています。
一般順序点には、すべての順序点とレ点がふくまれています。この定理２を繰り返し適用すると、すべての順序点とレ点をもった漢字列をカッコつき表示に置き換えることができ、漢字だけの最終訓読漢字列に置き換えることができます。
熟語を示す漢字連結記号でつながれた二字以上の漢字は、カッコつき表示のカッコをすべてはずしたあと、漢字連結記号をとりのぞきます。

４－６．一般順序点列のカッコつき表示

一般順序点列は、いわば標準順序点列の１つちがいの順序点のあいだが空の部分列φであってもよい、と認めたものですから、カッコつき表示は、標準順序点列とまったく同じものになります。

$$P_k\, f\, Q_{k-1}\, g\, R_{k-2} \cdots\cdots S_3\ \ h\, T_1$$

　＝Ｐ（ｆＱ（ｇＲ（……Ｓ（ｈＴ))))

ここで、たとえばｆが空でなく、ｇ、ｈが空の部分列φであれば、

　Ｐ（ｆＱ（Ｒ（……Ｓ（Ｔ))))

と、ｇとｈを取り除いた、カッコつき表示になるだけです。

４－７．訓読アルゴリズムの基本定理

以上で、「正しく返り点がついているｎ個の漢字の列、原文漢字列は、カッコつき表示を適用すれば、機械的な置き換えや計算で、つねに一定の返り点のないｎ個の漢字の列、最終訓読漢字列に並び替えられる」ことが証明できました。
最後に「訓読アルゴリズムの基本定理」として、定式化しておきます。
【定理３：訓読アルゴリズムの基本定理】
任意の原文漢字列は、カッコつき表示を適用すると、最終訓読漢字列に変形できる。
〔証明〕
原文漢字列にふくまれる返り点は、一二点、上下点、甲乙点などの順序点と、レ点の、返読を指示する二種類の返り点、および熟語を示す漢字連結記号しかない。【定理２：一般順序点列の返読定理】を、原文漢字列にある、順序点およびレ点をもった、あらゆる部分列に繰り返し適用すると、カッコつき表示により、すべての順序点とレ点は解消され、漢字と漢字連結記号だけの列になることは、明らか。ここからさらに、漢字連結記号を取り除けば、漢字だけの最終訓読漢字列が得られる。（証明終わり）

４－８．一般順序点列のカッコのつけ方

一般順序点列のカッコのつけ方は、標準順序点列と同じように、同じ種類の、２点以上のすべての一般順序点のところに「(　」を、対応する一般順序点１点のうしろに「　)」をつけます。完成したカッコつき表示からわかるように、一般順序点の１点をもつ漢字は、

$T_1 = T$

のように置き換えるので、この「順序点１点のうしろに「　)」をつけます」は、「順序点１点のところに「　)」をつけます」と言ってもかまいません。

「「(　」をつける、すべての一般順序点」とは、原文漢字列でいえば、上中下点なら下点と中点、一二三…点なら一点以外の二、三…点、および、それぞれのレ点になります。４－１で注意したように、それぞれのレ点の位置に２点以上の一般順序点があらわれます。

また、４－３で注意したように、１点とレ点の複合は、その位置に一般順序点がひとつ置かれるので、返読の開始をあらわす「(　」をひとつだけつけます。

対応する一般順序点１点とは、返読がスタートするもっともうしろの漢字につけられた一般順序点という意味です。

本稿の文例のなかで、もっとも複雑な文例７のカッコつき表示はつぎのようになります。

$A_{\text{レ}}B_{\text{下}}C_{\text{三}}DE_{\text{二}}FG_{\text{一}}H_{\text{レ}}IJ_{\text{レ}}K_{\text{中}}L_{\text{レ}}M_{\text{二}}NO_{\text{一}}P_{\text{上}}$　　文例７（再掲）

＝Ａ（Ｂ（Ｃ（ＤＥ（ＦＧ））Ｈ（Ｉ）Ｊ（Ｋ（Ｌ（Ｍ（ＮＯ））Ｐ））））

このカッコつき表示を解説すると、ここでは、

$A_{\text{レ}}B_{\text{下}}$

および

$J_{\text{レ}}K_{\text{中}}$

の２つのレ点、下点、中点の位置に「(　」があり、対応する１点つまり

$P_{\text{上}}$

のうしろに、四つの「　)」がついています。

三点をもつ漢字Ｃと、二点をもつ漢字Ｅには、それぞれ「(　」がつき、対応する一点をもつＧのうしろに、二つの「　)」があります。

閉じた部分列、

$H_{\text{レ}}I$

と、同じく閉じた部分列、

$L_{\text{レ}}M_{\text{二}}NO_{\text{一}}$

には、一般順序点をつけたものと、カッコをつけたものを示します。

$H_{\text{レ}}I = H_2 I_1 =$ Ｈ（Ｉ）

$L_{\text{レ}}M_{\text{二}}NO_{\text{一}} = L_3 M_2 NO_1 =$ Ｌ（Ｍ（ＮＯ））

最後に、文例７のカッコを外側からはずしてゆき、最終訓読漢字列を示しておきます。

$A_{\text{レ}}B_{\text{下}}C_{\text{三}}DE_{\text{二}}FG_{\text{一}}H_{\text{レ}}IJ_{\text{レ}}K_{\text{中}}L_{\text{レ}}M_{\text{二}}NO_{\text{一}}P_{\text{上}}$

＝Ａ（Ｂ（Ｃ（ＤＥ（ＦＧ））Ｈ（Ｉ）Ｊ（Ｋ（Ｌ（Ｍ（ＮＯ））Ｐ））））

＝Ｂ（Ｃ（ＤＥ（ＦＧ）Ｈ（Ｉ）Ｊ（Ｋ（Ｌ（Ｍ（ＮＯ））Ｐ）））Ａ

＝Ｃ（ＤＥ（ＦＧ））Ｈ（Ｉ）Ｊ（Ｋ（Ｌ（Ｍ（ＮＯ））Ｐ））ＢＡ

＝ＤＥ（ＦＧ）ＣＩＨＫ（Ｌ（Ｍ（ＮＯ））Ｐ）ＪＢＡ

＝ＤＦＧＥＣＩＨＬ（Ｍ（ＮＯ））ＰＫＪＢＡ

＝ＤＦＧＥＣＩＨＭ（ＮＯ）ＬＰＫＪＢＡ

＝ＤＦＧＥＣＩＨＮＯＭＬＰＫＪＢＡ

５．カッコつき表示の実例

以下に、基本となる原文漢字列の例をあげ、カッコつき表示の適用から、最終訓読漢字列を導くまでを示します。一般順序点列での、返読の定義

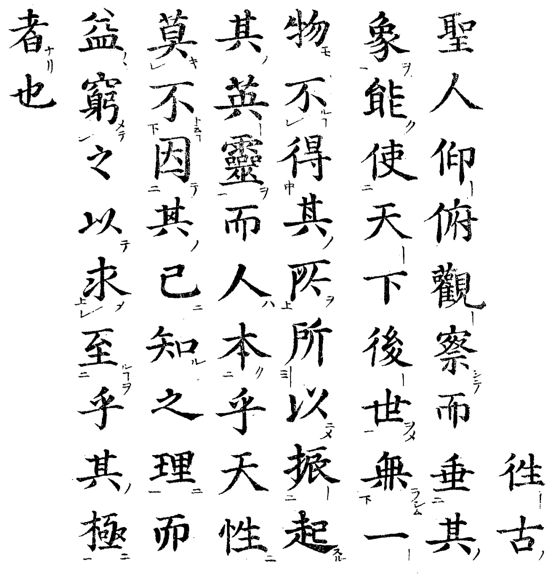
Ｐ（ｍ）＝ｍＰ

のＰやｍの表記は以下の説明でそのまま利用します。

［例１：レ点の返読］
原文漢字列Ａ㆑Ｂは、カッコつき表示でＡ（Ｂ）になり、最終訓読漢字列はＢＡです。
　　Ａ㆑Ｂ＝Ａ（Ｂ）＝ＢＡ
このとき、返読される閉じた部分列ｍは、漢字Ｂの一字だけです。
［例２：ふたつのレ点の返読］
Ａ㆑Ｂ㆑Ｃはカッコを二重につかって、
　　Ａ㆑Ｂ㆑Ｃ＝Ａ（Ｂ（Ｃ））＝Ｂ（Ｃ）Ａ＝ＣＢＡ
です。Ａ（Ｂ（Ｃ））＝Ｂ（Ｃ）Ａは、ｍをＢ（Ｃ）とみなし、Ｂ（Ｃ）Ａ＝ＣＢＡはｍがＣだけになっています。
これは外側からカッコをはずしていますが、内側のカッコからはずしても同じ結果になります。
　　Ａ㆑Ｂ㆑Ｃ＝Ａ（Ｂ（Ｃ））＝Ａ（ＣＢ）＝ＣＢＡ
一般に「カッコはどこからはずしてもよい」という定理が成り立ちます。証明は、ＡとＢを漢字、ｆとｇを任意の閉じた部分列（空の部分列でもよい）として、
　　Ａ（ｆＢ（ｇ））＝ｆＢ（ｇ）Ａ＝ｆｇＢＡ　　外側のカッコから処理
　　Ａ（ｆＢ（ｇ））＝Ａ（ｆｇＢ）＝ｆｇＢＡ　　内側のカッコから処理
となり、どちらからカッコをはずしても同じ訓読漢字列が得られるからです。
この実例集では、原則として外側からカッコをはずすよう統一していますが、さきにのべた文例７のように複雑な入れ子構造をもつものは、内側から処理したほうが良いかもしれません。
［例３：一二三点］
一点から二点、三点まで返るケースは、カッコを外側から処理すると
　　Ａ㆔ＢＣ㆓ＤＥ㆒＝Ａ（ＢＣ（ＤＥ））＝ＢＣ（ＤＥ）Ａ＝ＢＤＥＣＡ
［例４：一二点のなかにレ点］
　　Ａ㆓Ｂ㆑ＣＤ㆒＝Ａ（Ｂ（Ｃ）Ｄ）＝Ｂ（Ｃ）ＤＡ＝ＣＢＤＡ
［例５：レ点と一二点の結合］
　　Ａ㆑Ｂ㆓ＣＤ㆒＝Ａ（Ｂ（ＣＤ））＝Ｂ（ＣＤ）Ａ＝ＣＤＢＡ
［例６：さらに複雑な一二点とレ点の複合］
　　Ａ㆓ＢＣ㆒㆑Ｄ㆓ＥＦ㆒
　　＝Ａ（ＢＣ（Ｄ（ＥＦ）））
　　＝ＢＣ（Ｄ（ＥＦ））Ａ
　　＝ＢＤ（ＥＦ）ＣＡ
　　＝ＢＥＦＤＣＡ
［例７：熟語を示す漢字連結記号と一二点］
ＡとＢのあいだに、ＡＢで一つの熟語や固有名詞であることを示す漢字連結記号があるばあいです。
　　Ａ－㆓ＢＣ㆒＝Ａ－Ｂ（Ｃ）＝ＣＡ－Ｂ＝ＣＡＢ
漢字連結記号は最後の段階ではずします。
［例８：複数の漢字連結記号］
原文漢字列を省略し、複数の漢字連結記号が存在するカッコつき表示と、その最終訓読漢字列を示します。
　　Ａ－Ｂ（Ｃ－Ｄ－Ｅ（Ｆ））
　　＝Ｃ－Ｄ－Ｅ（Ｆ）Ａ－Ｂ
　　＝ＦＣ－Ｄ－ＥＡ－Ｂ
　　＝ＦＣＤＥＡＢ

６．『量地指南』穂積英の序

上下点、一二点、レ点、漢字連結記号が複雑に入り混じった原文漢字列の例として、測量家の村井昌弘が著した『量地指南』の冒頭にある穂積英の序（一部）（1732)をとりあげ、カッコつき表示から、最終訓読漢字列を導く過程を示します。さらに読点と、原文の送り仮名を付加し、黙字を除去。こうした機械的な操作で、人力による読み下し文とほとんど変わらない訓読が得られることを示します。
なお、ここでは内側からカッコをはずしています。
「１次の変換」は、重なっているカッコの、もっとも内側にあるカッコをはずしたことを示しています。カッコが重なっていない場合は、この１次の変換でカッコがはずれます。
「２次の変換」は、１次の変換で得たカッコつき表示の、重なっているカッコの、もっとも内側にあるカッコをはずしています。
以下、同様です。


徃古聖人仰俯觀察而垂其象能使天下後世無一物不得其所所以振起其英靈而人本乎天性莫不因其已知之理而益窮之以求至乎其極者也

徃古
聖人仰俯觀察而垂二其
象一能使二天下後世一無下一
物不レ得中其所上所－三以振－二起
其英靈一而人本二乎天性一
莫レ不下因二其已知之理一而
益窮レ之以求上レ至二乎其極一
者也

［注１：「天性」のあとの「一」点は原文にないが、明らかな間違いなので補った］

〔返り点〕（横書き）
徃古聖人仰俯觀察而垂二其象一能使二天下後世一無下一物不レ得中其所上所－三以振－二起其英靈一而人本二乎天性一莫レ不下因二其已知之理一而益窮レ之以求上レ至二乎其極一者也

〔カッコ付き表示〕
徃古聖人仰俯觀察而垂（其象）能使（天下後世）無（一物不（得（其所）））所－以（振－起（其英靈））而人本（乎天性）莫（不（因（其已知之理）而益窮（之）以求（至（乎其極））））者也

〔１次の変換〕（読みやすさを考慮し、はずした（）をそのまま残している）
徃古聖人仰俯觀察而其象垂（）能天下後世使（）無（一物不（其所得（）））所－以（其英靈振－起（））而人乎天性本（）莫（不（其已知之理因（）而益之窮（）以求（乎其極至（））））者也

〔２次の変換〕
徃古聖人仰俯觀察而其象垂（）能天下後世使（）無（一物其所得（）不（））其英靈振－起（）所－以（）而人乎天性本（）莫（不（其已知之理因（）而益之窮（）以乎其極至（）求（）））者也

〔３次の変換〕
徃古聖人仰俯觀察而其象垂（）能天下後世使（）一物其所得（）不（）無（）其英靈振－起（）所－以（）而人乎天性本（）莫（其已知之理因（）而益之窮（）以乎其極至（）求（）不（））者也

〔４次の変換〕
徃古聖人仰俯觀察而其象垂（）能天下後世使（）一物其所得（）不（）無（）其英靈振－起（）所－以（）而人乎天性本（）其已知之理因（）而益之窮（）以乎其極至（）求（）不（）莫（）者也

〔漢字連結記号を除去〕
徃古聖人仰俯觀察而其象垂（）能天下後世使（）一物其所得（）不（）無（）其英靈振起（）所以（）而人乎天性本（）其已知之理因（）而益之窮（）以乎其極至（）求（）不（）莫（）者也

〔（）を読点に変換〕（これは返読操作をおこなうたびに読点を付加することに等しい）
徃古聖人仰俯觀察而其象垂、能天下後世使、一物其所得、不、無、其英靈振－起、所－以、而人乎天性本、其已知之理因、而益之窮、以乎其極至、求、不、莫、者也

〔原文の仮名を付加〕（カタカナ合字は「コト」「シテ」等に読み替えている）
徃古ノ聖人仰俯觀察シテ而其ノ象ヲ垂、能ク天下後世ヲシテ使、一物モ其ノ所ヲ得、不ルコト、無ラシム、其ノ英靈ヲ振－起スル、所－以ニシテ、而人ハ乎天性ニ本ク、其ノ已ニ知ル之理ニ因テ、而益〱［注２］之ヲ［注３］窮メテ、以テ乎其ノ極ニ至ルコトヲ、求メ、不トイフコト、莫キ、者也
［注２：「益」の送り仮名は縦書きの繰り返し記号「〱」を用いている］
［注３：「之」の「ヲ」は原文にないが、「これ」と読んでいるのは明らかなので補った］

〔黙字を除去〕
徃古ノ聖人仰俯觀察シテ其ノ象ヲ垂、能ク天下後世ヲシテ、一物モ其ノ所ヲ得、不ルコト、無ラシム、其ノ英靈ヲ振起スル、所以ニシテ、人ハ天性ニ本ク、其ノ已ニ知ル之理ニ因テ、益〱之ヲ窮メテ、以テ其ノ極ニ至ルコトヲ、求メ、不トイフコト、莫キ、者也

〔訓読〕（人力による）
徃古ノ聖人、仰俯觀察シテ其ノ象ヲ垂レ、能ク天下後世ヲシテ、一物モ其ノ所ヲ得ザルコト、無ラシム。其ノ英靈ヲ振起スル、所以ニシテ、人ハ天性ニ本ク、其ノ已ニ知ルノ理ニ因テ、益々之ヲ窮メテ、以テ其ノ極ニ至ルコトヲ、求メザルトイフコト、莫キモノ也。

この例からは、次のことがわかります。
○最後の〔訓読〕以外は、機械的におこなえる。
○この例文には、「天性」の一点を除いて、訓読アルゴリズムの理論上の問題はないが、訓読操

作のすべてを機械的におこなう際の問題が送り仮名の表記などにある。
・「これ」と読む「之」のあとの「ヲ」が原文にない。省略されるときがままある。
・「不ルコト」「不トイフコト」と「不」の送り仮名が二通りある。
・「益」の送り仮名に縦書きの繰り返し記号「く」を用いている。

７．まとめ

以上、あらゆる返り点つき漢文の、すべての訓読操作をカバーしていることが、おわかりいただけると思います。
二つのカッコ記号（　と　）、およびハイフン－　の三つだけを使った、カッコつき表示は、式の変形をおこなうだけで、最終訓読漢字列をみちびくことができます。
カッコつき表示は、ひと目で返読が入れ子になっている回数がわかり、「漢文の返読度（？）」が簡単に計算できます。
難点をいえば、再読文字をふくむ漢字列、たとえば、

　未レ発

の最終訓読漢字列「発未」からは、「いまだ発せず」となかなか読みにくい、という点です。
現在、再読文字の左右の傍訓の問題は、送り仮名ならぬ「迎え仮名」の概念を取り入れれば、解決できるであろう、と思っていますが、冒頭に述べたように、本稿は、『それぞれの漢字の意味や役割はもちろんのこと、音読み、訓読みの区別、句読点の位置なども対象にしていません』。
なにぶん、『ここでの漢字は単に「集合の要素」という意味しかもっていない』のですから。

（以上）

（付記）
ここでは、どちらかと言えば、先に一般のばあいを述べ、ｎ＝１や２の場合を reduce している。
レ点や一二点をもつ最小単位の漢字列から、複雑なものを inductive に構成することも可能なような気がする。しかし、おそらくそれは非常に読みづらい記述になるのではないかと思う。言うなれば、数学と国語の発想の違いがここにある。
また、「レ点に番号をつける」というアイデアは誰でも思いつくことなので、もし評価していただくとすれば、「レ点が前の漢字についているか、うしろの漢字についているか」は、ともかくとして、「閉じた部分列でレ点を embed していること」と「標準順序点列のアイデア」の評価をお願いしたい。
訓読のアルゴリズムを考えたのは、そもそも和算序林をつくっているとき、原文の漢字列を何度も copy and paste していて、「こんなことは機械にやらせればよいのではないか」「なんとか機械的に訓読漢字列を得られないものか」と思ったことに始まっている。
必要は発明の母とは、こういうことを言うのではなかろうか。