

日本語プログラミング言語と自然な語順

望月正道

1. 日本語の自然な語順

日本語の語順（連用修飾語の順序）はかなり自由であるといわれる。例えば、次の(1A), (1B)はどちらも自然である。

(1A) 花子を 3年B組のクラス委員に 決める。

(1B) 3年B組のクラス委員に 花子を 決める。

しかし、いつでもそうであるとはかぎらない。上の文の「花子」と「3年B組のクラス委員」を入れ替えた文は、次のようになる。

(2A) 3年B組のクラス委員を 花子に 決める。

(2B) *花子に 3年B組のクラス委員を 決める。

(2A)は、そこで述べている内容は(1A), (1B)とほぼ同じものであり、かつ自然な語順だが、(2B)の語順は落ち着かない（この点で、小泉保・船城道雄・本田 治・仁田義雄・塚本秀樹 編『日本語基本動詞用法辞典』（大修館書店、1989年）の「きめる」の項の(1)《文型d》は、さらに細分する必要があるようだ）。

これは、次の(3B)が落ち着かないのに似ている。

(3A) 3年B組のクラス旗を 赤く 染める。

(3B) *赤く 3年B組のクラス旗を 染める。

このことから、(2A)の「花子に」は、(3A)の「赤く」と同じはたらきをしていると見ていわゆる「補語」と考えるのが適切かもしれない。

では、(1A)の「3年B組のクラス委員に」も同じく「補語」と考えてよいものだろうか。試みに、(1A), (1B), (2A), (2B)の助詞「に」を「と」に変えてみると、

(4A) 花子を 3年B組のクラス委員と 決める。

(4B) *3年B組のクラス委員と 花子を 決める。

(5A) 3年B組のクラス委員を 花子と 決める。

(5B) *花子と 3年B組のクラス委員を 決める。

となって、今度は(1B)に対応する(4B)の不自然さが目立つようになる。語順を変えても不自然ではない(1A), (1B)の「3年B組のクラス委員に」を、(2A)の「花子に」や(3A)の「赤く」、(4A)の「3年B組のクラス委員と」などと同列に扱ってよいものかという疑問も生じる。

ここで連用修飾語の末尾の助詞に注目すると、(2A)では

——を ——に 動詞

という語順である。しかし、これだけが自然な語順ではないことは(1B)を見ても明らかであるし、(2A)と同じ助詞の並びを持つ(1A)の方が、(1B)にくらべてより自然であるとも言えない。

(6A) とてものがんばりやの花子を クラス委員に 決める。

(6B) クラス委員に とてものがんばりやの花子を 決める。

語順の自然さは、その連用修飾語の長さにも影響を受けるようで、(1A)と(1B)、(6A)と(6B)で、どちらが「より自然」かというのは、読み手や場面によって受け取り方が異なるだろう。長さの影響は、次の例の(7B)を(3B)と比較するとよりはっきりする。

(7A) 旗を 赤く 染める。

(7B) 赤く 旗を 染める。

(3B)の「3年B組のクラス旗」はいかにも不自然だが、それを「旗」に置き換えた(7B)では、それほど不自然でなくなっている。

ここでは、これらの語順の自然さ・不自然さが何に由来するかの議論をする用意はない。ただ、それがかなり微妙な問題であることを指摘するにとどめる。

2. 日本語プログラミング言語Mindにおける単語の命名

Mind (マインド) は、パーソナルコンピュータで利用できるプログラミング言語である。Mindの最大の特徴は、自然な日本語にきわめて近い表現でプログラムが記述できることである。

普通の自然な日本語との違いが目立つのは、分かち書きが強制されることだが、これはMindが構文解析や意味解釈といった冒険をしていないからである。単語中のひらがなは読み飛ばされ、例えば

分かち書き

分かち書

分ち書き

分ち書

分書き

分書

は、すべて同じ単語として扱われる。これは、日本語で著しい表記のゆれを吸収するためだが、新聞の斜め読みで漢字だけ拾っていくのと似ていなくもない。

ただし、語末のひらがな(助詞・活用語尾など)の一部は参照していて、

2 から 3 を 引く
と、
2 を 3 から 引く
を混同することはない。

主格の助詞「が」は、条件節くらいにしか現れないが、これは手続き型のプログラミング言語である Mind では、プログラム全体が広い意味での命令表現で書かれるためだと考えられる。

また、FORTH 系のプログラミング言語に共通する特徴として、新しい単語をどんどん定義し増やしていくことが出来る（というより、Mind でプログラムを書くことは新しい単語を定義することそのものである、というほうがより実態に近い）。新しい単語名もちろん日本語で命名する。係助詞「は」が登場するのはほぼこの場面だけで、これも日常の日本語と異なる点といえるが、定義そのものは Mind を知らなくても容易に理解できる。

半分とは 2 で 割ること。

挨拶するとは 「今日は」と 表示すること。

このように、自然な日本語にかなり近い表現でプログラムが書ける（詳細は、片桐明著『日本語プログラミング言語 Mind』（翔泳社、1988年）、加野島英渡+大網呼人『日本語でプログラムを書く方法』（翔泳社、1989年）を参照のこと。ここでは論旨に関わる点だけ述べた）。

だが、ときにはそれが災いすることもある。

3. 「設定する」と「決める」

例えば「文字の色を設定する」という Mind 単語が定義済みだとする。この単語を引用するには、

- (8) 赤で 文字の色を設定する。
- (9) 赤を 文字の色に設定する。

のいずれかを使うことになるが、「設定する」という表現が漢語（字音語）サ変動詞であるせいもあってかどちらもそれほど不自然ではないように思う。

ところが、「設定する」では小中学生には難しいから、代わりに「決める」を使いたいということになると、

- (10) *赤で 文字の色を決める。
- (11) 赤を 文字の色に決める。

(10) のような不自然な表現が生じてしまう（「関節技を決める」の場合の「決める」の語感？）。また、(11) は(1A) と同じ構造だから良いはずなのだが、「赤を」が短過ぎるせいか

落ち着かない（「最も明るい赤を」のような場合は悪くない）。

(12A) 文字の色を 赤に 決める。

(12B) *赤に 文字の色を決める。

これらのうちでいちばん自然な表現は(12A)だろうが、「文字色決」全体で一つの Mind 単語だからここでは使うわけにはいかず、それと語順だけ異なる(12B)は(2B)と同様の非文である。結局、やや落ち着きの悪い(11)しか残らない。

同様に、「位置を設定する」という Mind 単語が難しいから、「位置を決める」と言い換えるということになると、例えば

(13) 画面の中央に 位置を決める。

(14) ?画面の中央を 位置に決める。

(14') *画面の中央の 位置に決める。

のいずれかを記述することになるが、この場合は(13)の方がふさわしいだろう((14')では「画面の中央の位置」という意味の連続を断ち切ってしまう)。

以上二つの例を見ただけでも、「設定する」のような漢語サ変動詞（を含む Mind 単語）を「決める」のような固有語の動詞（を含む Mind 単語）で言い換えるというのは、固有語の動詞にはいわゆる基本動詞が多いだけに、単に機械的に置き換えるのでは弊害が大きいことが予想できる。

むしろプログラミング言語は自然の言語そのものではないから、多少の不自然さはしかたのないところであるが、これはまさに小中学生には難しいからという教育的配慮から出た言い換えなのであるから、自然な語順とふさわしい助詞の選択を大事にする必要があるだろう。(10), (12B), (14)のような不自然な表現は廃すべきだと考える（「設定する」を「決める」に言い換えるという提案については、西之園晴夫著『日本語でプログラムしよう』（学習開発研究会；京都教育大学教育実践研究指導センター内、1989年）を参照）。

4. Mind の逆転介入と自然な語順

Mind においても語順や助詞に注意を払うべきだということには、教育的配慮だけではなく、Mind 内部の事情もある。

既に述べたように Mind では、語末のひらがなの一部が参照され、例えば

(15A) 単価を 合計に 入れる。

(15B) 合計に 単価を 入れる。

のどちらであっても結果は同じになる。けれども

(16A) 単価 1 と 単価 2 を 加えたものを 合計に 入れる。

(16B) 合計に 単価 1 と 単価 2 を 加えたものを 入れる。

においては結果は異なる。(16A)では期待どおりの値が「合計」に入るが、(16B)では「処理

単語 "入" の前に変数が有りません。」というエラーとなる。

これは、Mind にとっての自然な語順は(15A)の方であり、(15B)では助詞を目印に両者を逆転して処理しているのだが、この逆転が働くのは隣接する二つの単語（より正確には、逆転介入の指定がある処理単語の直前の2単語）までであるため、(16B)のように離れてしまうと逆転できなくなることによる。

このタイプの Mind 単語としては「入れる」のほかに「増加する」「減少する」などがあるが、いずれも、その直前に副作用を与える対象（変数）を置く方が Mind での標準形である（なお「増加する」「減少する」は自然な日本語としては自動詞であるので、ぜひ「増やす」「減らす」と他動詞の形に改めたいところである）。

(16B) はけっして不自然な日本語ではない。そうであるだけに Mind でエラーとなるこのような表現は、たいへんやっかいである（もっとも必ずエラーチェックに引っ掛かるのでたいてい二度目にはうまくいくことになるのだが）。

Mind にはこのほかに

(17A) 1 2 を 4 で 割る。

(17B) 4 で 1 2 を 割る。

の(17B)のように、直前に二つのデータが並ぶ場合に働く逆転介入もあるが（既に触れた「引く」もこの類）、この「助詞の優先値」を見るやり方はたいへんよく出来ていて、自然な日本語ともよく一致する（それだけに、(16B)はよけいに残念な気がする）。

5. 「塗る」の場合

このように、自然な日本語の語順と Mind の語順とは必ずしも一致しない。そこで、新しい単語を定義し、命名する際には（それが使い捨ての単語でないならば特に）慎重な配慮が必要になる。

仮に「塗る」という Mind 単語を新しく定義したとしよう。

自然な日本語では、

(18A) 画面を 赤い色で 塗る。

(18B) 赤い色で 画面を 塗る。

のどちらの表現も可能である。単語を定義する際に「割る」や「引く」で働くのと同じように逆転介入を行うように指定しておけば、(18B)の場合に逆転介入が行われ、(18A')の形に整理される。

(18A') 画面 赤色 塗る。

ところが、また

(19A) 赤い色を 画面に 塗る。

(19B) 画面に 赤い色を 塗る。

の表現も（意味はいくぶん異なるが）自然であり、この場合は(19B)で逆転介入が働いて(19A')の形に整理される。

(19A') 赤色 画面 塗。

結局、(18A')と(19A')が残ることになるのだが、「塗る」に渡されるデータは(18A')と(19A')では全く異なっているので、この「塗る」という Mind 単語はしばしば期待を裏切ることになるだろう。そうして悪いことに、(18A) (18B) (19A) (19B)のいずれもが自然な表現なので、間違いの発見はたいへん困難な作業になってしまう。

もちろん、逆転介入を指定せずに、必ず「画面 赤色 塗る。」の語順で単語を並べるという解決法が無いわけではない。しかし、これもまた予期せぬ失敗の原因になりかねない。予想と逆の語順で記述されていても、それを Mind も、プログラマ自身も発見できないのである。

結局、ここで考えた「画面」(対象)と「赤色」(色)の二つを要求するような Mind 単語を定義するのは、あまりうまくいかないことがわかる。どちらか一つだけなら問題はない(BASIC などではパラメータの省略が可能だが、Mind では処理単語の要求するだけのデータを必ず用意しておかなければならないという点からも、むやみに多くのデータを要求する単語を定義するのは考えものである)。「対象」ではなく「位置」ならば、「位置 色 塗る」の語順に固定しても不都合は少ない。

結 び

Mind では、自然な日本語に近い表現でプログラムが記述できるのだが、そのためにかえって自然言語に引かれてしまい、予期したとおりの働きをしないプログラムの出来上がることも起り得る。

特に（ことの当否は別としても）小中学生にプログラム言語として与えようというのであればなおさら、Mind 単語の命名には注意を払うことが必要になるだろう（ただし、反復や分岐のようなプログラム言語特有の制御構文については、むしろ積極的に、ある程度日常語と異なる表現をとるのが望ましいことは言うまでもない）。

(文中敬称略)