

うな叙述の方法を使おうとすれば、頭がおかしくなって手近な溝に転げ落ちるものもありそうなのだ。この形式的な叙述様式は、このような目的には合わない。しかし他に合う形式がある。

教育研究の分野は、そのような形式を発展させるといふ方向には進んでこなかった。しかし、他の研究分野、コンピューター学者の社会は、(それ自身のために)叙述的な言語という問題に取り組まねばならず、教育革新の予期せぬ源泉となった。コンピューターは多くのことに用いられている。そしてコンピューターに何かをさせるには、その基礎をなす過程を、ある水準でコンピューターが遂行するのに十分な正確さでもって叙述することを必要とする。そのため、コンピューター学者は、強力な叙述形式を考え出すことに多くの能力と精力を注ぎ込んだ。コンピューター・サイエンスという名称は不的確だとさえ言える。その大半はコンピューターの科学ではなく、叙述とということ、そして叙述のための言語の科学なのである。コンピューター・サイエンスの産み出した叙述形式のあるものは、まさに、身体的技術の修得の過程をつかむ上で必要なものだ。ここでプログラミングから来る一組の重要な概念を選んでこの点を説明したい。これは、構造的なプログラムという概念であるが、ロゴの環境における小学5年生の学習経験を例にして話そう。

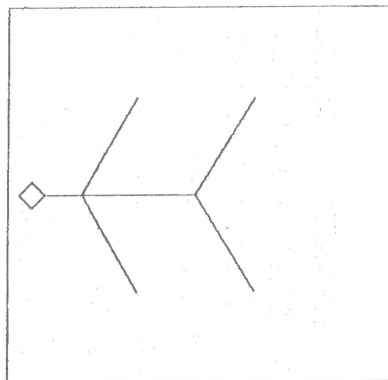


図 10a
目 標

キースは、前頁図のような線画きのヒトの形をコンピューターに画かせようと目標を定めた(図10a参照)。

彼の計画は、左の足から始めて右足へ、そして上へと順番にたどってタートルの筆運びを進めることだった。ここで彼は、コンピューター以前の文化で見慣れたイメージを、かつて点を結んで絵を画いたり、自分の行動を一步

TO MAN
(ヒトの定義)

FORWARD 300
RIGHT 120
FORWARD 300
RIGHT 180
FORWARD 300
LEFT 120
FORWARD 300
LEFT 120
FORWARD 300
RIGHT 180
FORWARD 300
RIGHT 120
FORWARD 300
RIGHT 180
FORWARD 300
LEFT 120
FORWARD 150
LEFT 45
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
END

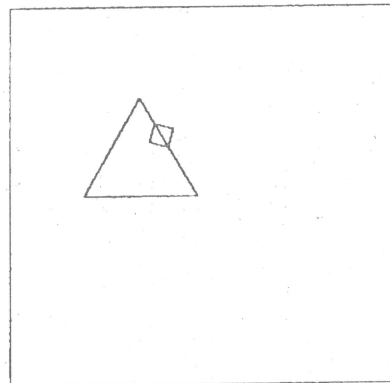


図 10c
バグのあるヒト

一歩説明することを覚えた方法を用いた。だから、ここで彼がこの方法に従ったのはごく自然なことだったのである。目標は少し退屈かも知れないが、ごく簡単に見えた。それがこんなプログラムになった(図10b参照)。

これを使って画面に描いてみると、「バグのあるヒト」と記した図のようになまなく予想しないものになった(図10c)。何がいけなかったのだろう。

キースはこのような不意打ちには驚かなかった。先に述べたように、ロボットの環境のメーンシステムの一つは、バグとデバグに関連した概念の集まりである。初めての試みがあるまま行くとは誰も思わない。「正しい—良い成績」「誤り—悪い成績」という基準で評価するものはいない。むしろ、「どうやったら直せるか」と問い、そして直すためには、何が起ったかをそれ自体の言葉で理解しなければならぬ。その上で初めて、我々の思うように実現させることができるのである。しかし、この状況においてキースは何が起ったのか想像がつかなかった。彼のプログラムの書き方は、間違いを見つけるのを極めて困難なものにしてしまった。プログラムのどこにバグがあるのか? いったいどんな間違いが彼の意図したこととはまるで見当外れなものに変えてしまったのか。

彼の苦境を理解するために、「構造的プログラム方法」として知られている別の手段と、彼のプログラムを比較してみよう。この目的は、プログラムを無理のない部分に分けて、それぞれの部分を別々に修正できるようにすることにある。キースの長い特徴のない命令の羅列ではバグを指摘することは難しいが、小さな部分を扱うのならバグを閉じ込め捉えて理解することは容易だ。この場合の自然な区分は、V型を描くプログラムを使って、それを腕と脚にそれぞれ用い、さらに正方形を使って頭に用いることだ。これらの「部分手順」をいったん書いて試した上で人間の形の全体を画く「全体手順」を書くのは極めて易しい。これは、次のような極めて簡単なプログラムにすることができる。例えば、

TO MAN (ヒトの定義)
VEE (V型型)
FORWARD 50 (前進 50)
VEE (V型型)
FORWARD 25 (前進 25)
HEAD (頭)
END (終り)

この手順は簡単だから全体として把握するのが容易だ。だが勿論、この簡潔性は VEE とか、HEAD とかという命令がコンピューターに理解されるという仮定の上に成り立っている。これも、常に全体として理解できる手順として書くという様式にそって作ることができる。例えば、

TO VEE (V型型の定義)
RIGHT 120 (右 120)
LINE 50 (線 50)
RIGHT 120 (右 120)
LINE 50 (線 50)
RIGHT 120 (右 120)
END (終り)

(このプログラムでは、LINE という命令を定義することを前提としている。これはタートルを前進、後退させる。)

VEE を使うために、次に LINE を定義しよう。

TO LINE : DISTANCE (線の定義 距離)
FORWARD : DISTANCE (前進 距離)
BACK : DISTANCE (後退 距離)
END (終り)

LINE は生来のロボットの命令だけを使ったものであるから、これ以上定義しなくても使えるMANのプログラムを完了するには、もう一つHEADを定義する必要がある。

TO HEAD (頭の定義)

RIGHT 45 (右 45)

SQUARE 20 (正方形 20)

END (終り)

中学1年のロバートは、自分のプログラム様式がこのように変わったことについて次のような感嘆の声を発した。「見てごらん、僕のプログラムは皆、頭に入る大きさ (mind-sized bites) になったよ。」そしてまた、こうつけ加えてこの比喩を強調した。「僕はプログラムにはよく混乱したものだんだけど、今は、いっぺんに噛みこなせる以上は口に入れないんだ。」彼は強力な概念に遭遇したのだ。一度に理解できないほどの段階を踏むことなしに、大きな知的構造を作ることが可能であるという概念だ。そして階層的な構造を用いることは、その組織を全体として、言い換えれば、「上から見た形」で理解することを可能にする。

構造のない MAN のプログラムを書いたキースは、既に部分手順を使うという考えに触れたことはあったが、その時にはこの考えを拒否した。直線的なプログラム様式の方が、彼のなじんだ行動様式に近かった。MAN というプログラムをデバッグできなかった日にいたるまで、構造的なプログラム様式を必要とすることはなかった。ロゴの環境ではこういうことはたびたび起こる。苦境に立って子供がどうしようかと尋ねるとき、大抵は、「どうしたらいいかわかっているでしょう」というだけで足りる。そして子供はたびたび、あるときは勝ち誇ったように、あるときは内気そうに、「部分手順に書き直さなさいといけないんじゃないかしら?」と言う。「正しいやり方」がキースに押しつけられたのではなく、コンピューターから充分な柔軟性と力を与えられたため、彼の探求が真実に彼のものとなったのだ。

仕事を計画し、遂行していく上でのこの二つの進行方法は広くみられるものである。これらは、知的な能力だけでなく、身体的な技術を修得する様式にもみられる。例えば、我々の子供研究室でプログラムミングと身体的技術を共に学んだ2人の小学5年生の場合をみて見よう。

マイケルは、強い、運動好きの、彼自身の目にも「タフな子」だが、ボールは、どちらかというと内向性の勉強家で、ほっそりした身体つきの子である。マイケルは学校の成績が悪く、ボールは成績が良かったのでコンピューターでも、ボールの方が速くすぐに複雑な構造的プログラム手順に進んでいたのを、どちらも不思議には思わなかった。数週間後に、マイケルは依然として、直線的な形のプログラムしか書けなかった。彼がもっとこみ入ったプログラムを書くのに必要な概念をすべて持っていたことに疑いはないのだが、部分手順を使うことに対する古風な強い抵抗感があった、それに引き止められていたのだ。

この頃に2人は竹馬乗りを練習し始めた。マイケルの方法は、竹馬乗りのモデルを直線的な形で頭に浮かべることだった。「足を横木にのせ、身体を持ち上げ、もう一方の足をもう一方の横木にのせ、最初の足を前進……」実際にやってみると、間をおかずして墜落したが、勇敢に何度も立ち上がり、いずれは出来るようになるという自信はゆらぎもなかった。また事実そうだった。しかし、2人が驚いたことには、ボールの方が先に出来るようになったのだ。

ボールのやり方は違っていた。同じやり方で始めたのだが、少しも進歩しないのをみて問題のある部分「バグ」を取り出してそれを修正しようとした。1歩前に踏み出すとき、竹を後ろへ置き忘れる傾向がある。このバグは、いったんそれと見極めたら根こそぎにするのは難しくない。その方法の一つは、足ではなく竹を前進させようという風に考え、足は竹が運んでくれるのに任せることである。これには、足と反対に腕で竹を持ち上げればよい。この考え方がプログラムミングに類似していることはボールの目には明らかだった。従ってこれをプログラムミングの勉強が身体的技術の修得に応用された実例と考えてもよからう。

実際、プログラムミングと竹馬とどちらの場合も、ボールが長年持っていた一般的な認識様式にある特色からくるものだと言えないこともない。だが、