

筆算シミュレーターの作成

清水 秀美

(愛知教育大学教育実践総合センター)

Development of a Simulator for Written Calculation

Hidemi SHIMIZU

(Center for Research, Training and Guidance in Educational Practice,
Aichi University of Education)

要約 数学の学力低下が進行している現在, 学習方法を見直す必要がある。算数学習は「理解」と「練習」の二段階からなるが, 特に練習による算数計算の「無意識的操作運用」についての学習方略については学校教育の中で看過されてきたきらいがある。本稿ではこの二段階を切り離すのではなく, 上級課題への挑戦を通して練習を有意義なものとする必要性が時間の短縮と, 動機づけの観点から提案される。細切れの各教科の下位単位を完全に学習させ, 次に新しい学習単に移る方式では学習者に心的飽和をもたらす。これを回避するためには, 不完全でも果敢に上級課題に取り組むなかで, 学習項目間相互の関連を見いだしつつ, 内発的動機づけを伴って, 計算能力を高めるべきであろう。以上の観点から, このような「先取的予習」を支援する学習補助教材の開発が意図された。今回は「人が解く方法で解く」筆算シミュレータについて報告する。

Keywords: 筆算, シミュレータ, 算数学習, 予習, 自習

問 題

ここで作成される筆算シミュレータは, 筆者が以前に発表した小学校計算シミュレータの一部を構成するものとして作成されたものである(清水, 1999)。人の計算方法は計算機と異なり, 色々な法則を用いて人間の限られた短期記憶容量を補いつつ計算を遂行する方略に基づいている。筆算はこの限られた少ない短期記憶容量を補うために紙と鉛筆を用いて, 一定の規則に従い値を求める計算方法である。小学校ではこの規則を修得させることが算数授業の目的の一つである。小数の掛け算と割り算は現行教科書では「小学5年の上巻」で取り扱われていて, 実に筆算に対して長期の学習時間が割り当てられている。

計算の授業ではこの計算方法を理解させることが第一に求められるが, 第二にその方法に習熟することが求められる。すなわち, 計算方法を理解して, 次にそれを無意識のレベルで操作運用できるまで高めることが必要である。この二段階が小中学校での学習の基本形である。そのため多くの練習問題を生徒に課すことになる。従来学校では練習を教師が生徒に宿題として大量に与えることで「無意識的操作運用」という目標に対応しようとしてきた。学力の危機が叫ばれているが(和田, 西村, 戸瀬, 1999; 市川, 和田, 1999), その理由の一つとしてこのような味気のない訓練が, 算数嫌いを助長し, ひいては中学校での大量の数学落ちこぼれを産む結果を招いていると考えられないだろうか。

理解させることに対する教師の技量は色々な研究活動や教育運動(例えば「分かる授業の展開」等)を通して著しい進歩を遂げているが, 残念ながら無意識の

レベルで操作できるまで計算技能を高める方法についてはほとんど手つかずのままである。多くの場合, 練習問題を大量に与え, 時間がなければ宿題として課すことになる。問題は, 宿題として課すことではない。教師がそれによって第二の学習目的, 「無意識的操作運用」の教育方法から目をそらし, 真剣な取り組みをしないことにある。多くの場合そこで採られる方法は, スパルタ的鍛錬か, 放置である。放置はいうにおよばず, スパルタ的訓練は算数のできる生徒にも, できない生徒にも, ともに憂鬱なものとなる。

もちろん, 計算への動機づけを高めるために, ゲームを採り入れる等の工夫が見られた。特に近年, コンピュータの普及につれて, 計算スピードを競うゲーム形式のソフトが流通するようになり, インターネットで容易に入手できるようになってきている(例えば, <http://www.download.com/> の教育関連項目選択により閲覧できる)。その強い副作用も考慮すべきであるが(Heinich, Molenda and Russell, 1989, p328-353), 学習の初期においては, ゲーム方式では計算のスピードを競う点数や勝負などに置き換え, いわば外発的ではあるが, 動機づけの点で有効な手段であると考えられる。ゲーム方式は, 惰性的となりがちな計算練習に変化をもたらす有効な手段である。

しかし, 計算における「無意識的操作運用」段階への到達に要する時間はいずれにしても短縮されない。練習時間を激減させる新たな別の方法を考えることが, 学力低下の現状を打開する方策の一つと考えられる。学校教育では多くの生徒を対象に一斉授業を行う関係で, 一定の指導要領に沿って計算の授業が展開される。

しかし、個別学習では状況は一変する。一定の指導要領に沿って進める必要はない（指導要領を否定するものではない。学習の指針として欠かせないものであると考える。）。例えば、九九の暗記が完璧でなくとも、2桁の掛け算に進み、小数の意味理解から、小数点の掛け算に一気に進むことができる。その過程において足し算の技能も、九九の技能も、練習のプラトー状態に到達するはずである。割り算でも同様である。要するに、計算の「無意識的操作運用」段階への到達は新しい課題への取り組みの中で自然と解消される。一方、理解したことが新たな課題でどの様に適用されるか興味を駆り立てることができる。これは小学校の算数に限定されない。微分積分計算の早期の学習により、それが高校の物理でどの様に利用されるか、その中で微分積分計算とその意味を再学習できるのである。従来の学習観はある一つの事柄の完全学習を目指し過ぎたのではないだろうか。そのためにかえって味気のない単純な事柄の繰り返しになる傾向を生んだのではないだろうか。新しい算数課題への挑戦のなかで算数計算の「無意識的操作運用」段階に至らせることも視野に入れてよいのではなかろうか。

このような学習（教育）を英才教育として批判する傾向があるが、予習の延長と捉らえることができる。またある教師は知識の伝授者としてこのような生徒を好まない。あくまでも自分が伝授の主人公でありたいのである。しかし、上記の学習方法を採用する生徒の側に見れば、自己の学習は粗削りであるから、新しい視点を授業から得る楽しみをもつであろう。また自己の理解を修正、付加する機会として捉らえるであろう。決して授業の軽視にはつながらないのである。逆に、授業がどのような意味をもつのか、より高い視点から見ることができ、一層の学習意欲が沸くものと期待できる。いわば内発的動機づけが与えられるであろう。

このような学習形態で望まれる補助教材の一つとして、シミュレータが考えられる。人間が計算するように解いてみせるソフトである。自学学習では、生徒に範例を提示す補助教材が必要であろう。従来では書物により上級課題の学習を行った。理解を深める上では書物は最重要教材である。しかし、書物は計算問題の疑問に適切に対処してはくれない。従って求められる補助教材として、生徒の疑問に対処できるものが望まれる。

生徒は単に与えられた問題を受動的に解くのではなく、疑問を抱きコンピュータに問い掛け、主体的に学習を進める。例えば“10-5”や“11-5”の計算方法を理解した後、“100-5”や“1000-905”の計算はどうなるか疑問を抱くであろう。生徒はそれについてある一定の予測を立てて、その正誤をシミュレータで確認することになる。ここでは生徒は主体的に問い掛ける存在となる。本稿において、このような学習形態を可能とするシステム、筆算シミュレータについて報告する。

システムの基本

このシステムを構築するに際して技術的問題点は多くはないが、時間と労力を必要とした。ただし「人間の解くように解く」ことをモットーにシステムを構成するわけであるが、人間の解きかたは一律ではない。“ $126 \div 13$ ”を解く時、9をたて、13に9を掛け、右から左に向けて“7”、“1”、“1”と書く。しかし、常に三桁であれば右から左に書くわけではない。“ $100 \div 25$ ”の場合、初めはそうであったとしても、“ $25 \times 4 = 100$ ”を記憶することにより左から一気に“1”、“0”、“0”と書くことになる。この様に算数は理解一辺倒ではなく記憶に依存する場合も多い。もちろん理解に裏打ちされているが、記憶が手順を変更する。別のシミュレートできない例として、小数の足し算では加算途中で小数点を付すか、計算終了後に小数点を付けるのか同一個人でも、場合により異なるであろう。このような点で完全なシミュレータではないが、一定の規則に従い初心者の指針として耐え得るように心掛けた。

また小学生の使用に耐え得るように、使用上の規則をできるだけ少なくするようにし、特に入力については、制限を付けないようにした。全角入力や半角入力、カタカナや平仮名入力、漢字入力、英文入力、記号入力、スペース入力などに対処するようにした。

以下に実行画面を例に試作システムを説明する。

I 足し算について

くり上げの過程を表示できるようにし、9数値（英語入力では3数値まで）の足し算も可能にした。また四則演算全てに共通するが、多様な命令入力に対応できるようにした。例えば「+」、「+」、「たす」、「タス」、「tasu」、「t」、「た」、「タ」、「足す」、等の演算命令に対処できるようにし、かつ、英語の命令も受け付けるようにした。例えば、“Add ninety-nine to(and) one hundred twenty-three”や“One plus two”でも演算を実行するようにした。英語の数詞学習を自己の操作体験と結合させる意図からである。

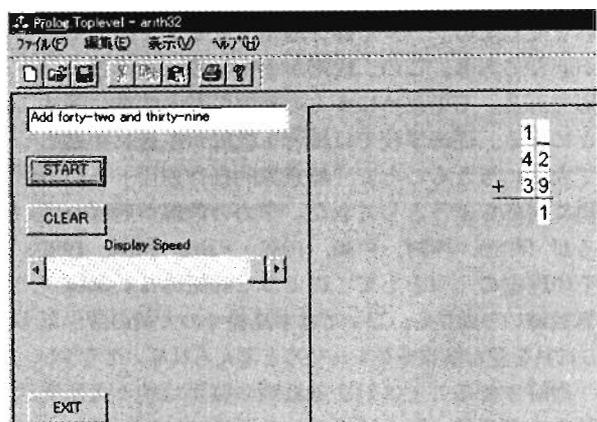


図1-1 英語入力による足し算命令の実行途中経過

図1-1は英語で演算命令を行った場合の、くり上げを表示しつつ演算を行っている途中経過である。演算速度（正しくは表示速度）は“Display Speed”バーで調整される。図1-2は3数値の演算処理途中経過である。図1-3は小数の足し算の処理結果で、小数点の位置揃えが行われている。

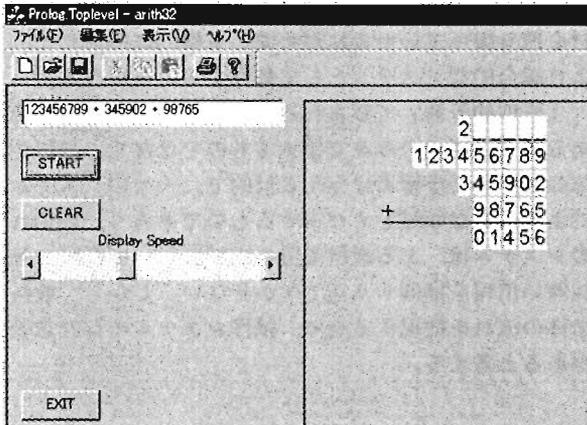


図1-2 3数値の演算処理途中経過

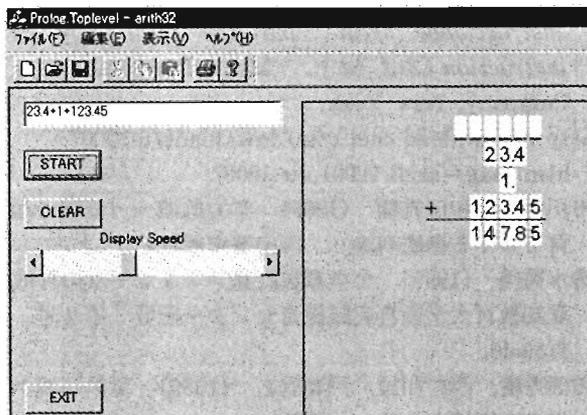


図1-3 小数点の位置揃えが行われている
少数足し算の処理結果

II 引き算について

くり下げの過程を表示できるようにした。引き算でのくり下げの操作を視覚化し、演算位置を明確にするために処理の済んだ部分を消去して提示することにし、終了とともに問題が再提示されるようにした。ここで用いられる命令は足し算と同様に、「-」、「ひく」、「ヒク」、「hiku」、「h」、「ひ」、「ヒ」、「引く」が利用できる。また英語では“Subtract eleven from twenty”と“Twenty minus eleven”の型が利用できる。図2-1は“100100-9009”のくり下げを伴う途中経過を示している。

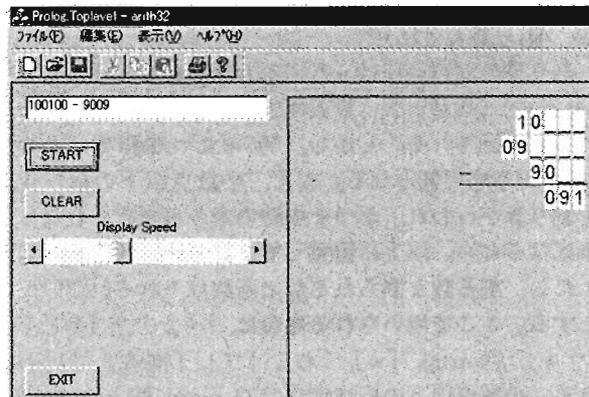


図2-1 “100100-9009”のくり下げを伴う途中経過

III 掛け算について

くり上げの過程を視覚化する。また“12300×450”のような数値の末尾部分が“0”であるような数値の掛け算の処理を可能にした。ただし掛けられる数値の桁と掛ける数値の桁の合計が16以下とする。16より大きい場合は、「数が大きすぎます」というメッセージを表示する。ここで用いる命令は、「×」、「*」、「かけ(る)」、「カケ(ル)」、「kake(ru)」、「k」、「か」、「カ」、「掛け(る)」が利用できる。また英語では“Multiply twenty-one by eleven”, “Multiply twenty-one and eleven together”, “Twenty-one times eleven”, および九九の“Two two”が利用できる。図3-1は“987×65”の途中経過を示している。図3-2は“Multiply twelve point three four by four hundred fifty”の処理結果を示している（小数点以下のゼロ処理が行われていることに注意）。

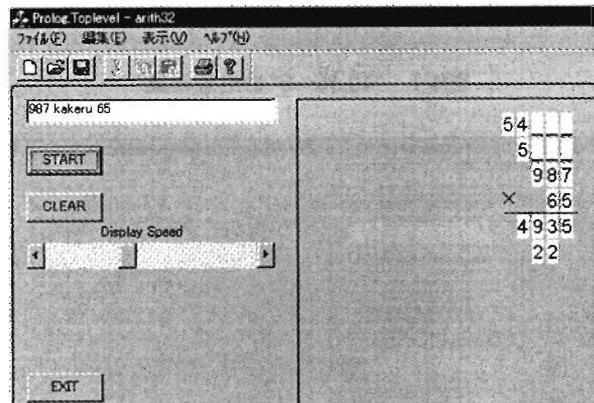


図3-1 “987×65”の途中経過

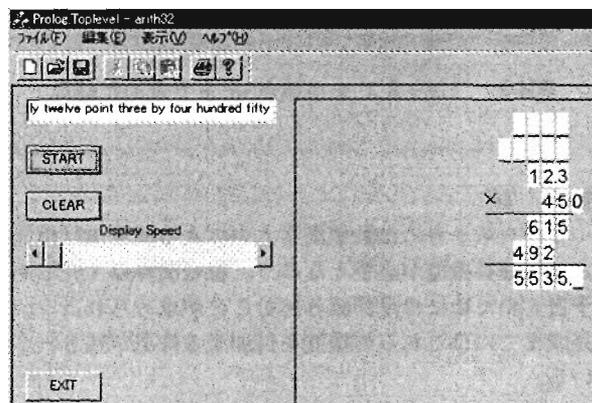


図3-2 “Multiply twelve point three four
by four hundred fifty”の処理結果

IV 割り算について

割り算の演算過程は「たてる」、「かける」、「ひく」、「おろす」(教育出版：算数4上)の4操作の継続操作により遂行される。しかし、「0」のたつ演算では後の3操作が省略可能となる。また、小数点以下の桁指定で割り算が行われ、かつその時の余りが求められなければならない。以上に留意してプログラムが組まれる。ただし、割る数と割られる数の桁数はそれぞれ13以下とする。ここで用いられる命令は、「÷」、「:」、「わる」、「ワル」、「waru」、「w」、「わ」、「ワ」、「割る」で行われる。小数点以下の桁指定は“123 waru 33 s 3”のように行われ、この例では小数点以下3桁までが指定されている。また、利用できる英語表現は、“Divide twenty by five”, “Divide twenty by six and calculate down to the 1st decimal place”の型である。図4-1は“10.15÷5”の処理結果である(啓林館：算数4年下38頁)。図4-2は“125÷8.3”を小数点3桁まで求めたものである。

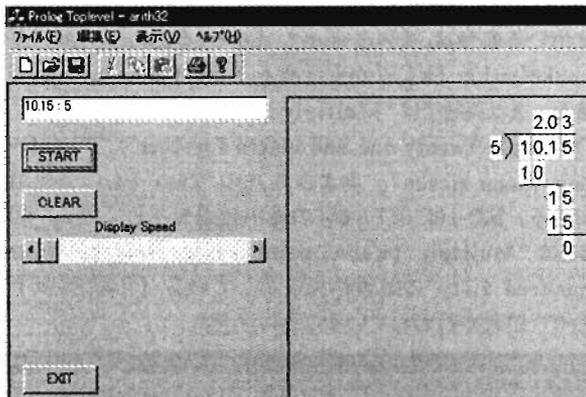


図4-1 “10.15÷5”の処理結果

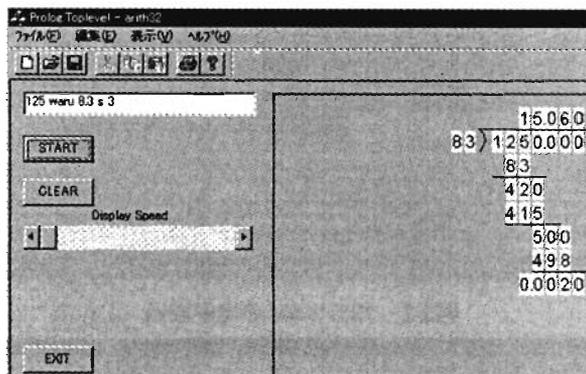


図4-2 “125÷8.3”を小数点3桁まで求めた結果

四則演算の筆算シミュレータの作成は、それぞれが独立したプログラムから構成されると考えられるであろうが、そうではない。掛け算は足し算と、割り算は、掛け算、引き算と密接に関連する。従って本シミュレータは足し算、引き算、掛け算、割り算の順に構成された。前回の算数シミュレータ(清水, 1998)とセットで小学生の算数計算は、完全ではないが、「人が解くように解くシステム」として完成する。従来の教育ソフトの多くが、足し算や掛け算、少数など断片的に教材を取り扱っていたのに対して、有機的に関連した、より総合的なシステムとして動作することになる。

「先取的予習」で望まれるソフトの一つは、教科のある特定の単元のみを支援するものではなく、教科全体の鳥瞰図を学習者に与える目的で、一つのシステムでより大きな領域をカバーするものであろう。細切れのシステムは、ある教科部分の細部について詳細に興味深い情報を提供する点で欠かせない。しかし、教科全体の流れを把握する点で、試作システムの存在意義があると考えている。

参考文献

- Heinich, R., Molenda, M. and Russell, D. R.(1989) *Instructional Media and the technology of instruction (3rd ed.)*. Macmillan Publishing Company, New York.
<http://download.cnet.com/downloads/0-10007.html?tag=st.dl.10001.dir.10007>
 市川伸一, 和田秀樹 (1999) 学力危機——受験と教育をめぐる徹底対論 河出書房新社。
 清水秀美 (1999) 小学算数計算シミュレータの作成 愛知教育大学教育実践総合センター紀要 第2号, 頁39-46。
 和田秀樹, 西村和雄, 戸瀬信之 (1999) 算数軽視が学力を崩壊させる 講談社。

後 記

このシステムには数値漢字入力による演算遂行機能と根号演算機能が必要となろう。根号演算は「先取的予習」のためには是非組み込むことが求められよう。完成までにはこれらの機能を付加する作業が残されている。