Conceptual vrs. Procedural Knowledge in Introductory Calculus – Programming Effects

Fredric W. Tufte
University of Wisconsin-Platteville

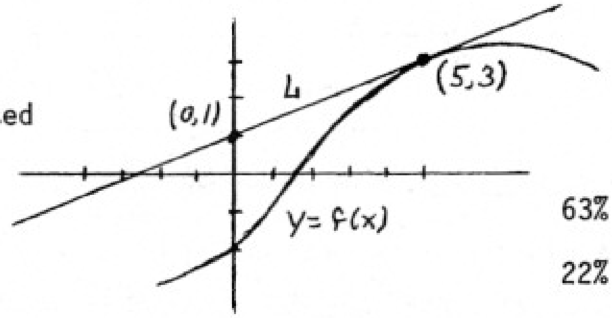
Student Knowledge. Evidence abounds that many calculus students are learning to execute tasks in a routine, algorithmic fashion while lacking very basic conceptual understanding of the topics being studied. For example, Table I gives results of several final exam questions given to approximately 200 students who were completing an introductory calculus course. The examples in this table demonstrate that we are having remarkable success on algorithmic problems. Some would say that with a little more practice we might insure a short life span for the HP-28C.

TABLE I
Points on Selected Drill and Template Problems

Question	Percent of Total
1. Find dy/dx: $e^{x+y} = \sin y$	86%
2. Find dy/dx: $y = x^{\pi} \cos(\pi x)$	92%
3. Integrate: $\int x^2 e^x \, dx$	89%
4. Integrate: $\int \frac{3}{x(x^2+1)} \, dx$	73%
5. Find the first three terms of the Taylor series for $f(x) = \sqrt[4]{x}$ about 1.	90%
6. Find the area of the region bounded by $y = 4x - x^2$ and $y = x^2$.	86%

The same student subjects reported on in Table I were given a Calculus Concept Test during the final week of the semester. Three of the problems on this test are reported in Table II. The results on these problems suggest that many students do not recognize the derivative presented in symbolic form, and that they are unable to formulate or interpret geometric representations of fundamental concepts. Not only did students have conceptual difficulty with fundamental concepts of calculus, but problem 2(a) indicates that almost 40% of these students had conceptual difficulty at an even more basic level. These students were unable to correctly interpret the relationship between a function and its graph. This suggests that many of the intuitive explanations that instructors use in the classroom may be of little benefit for many students. The results on problem 3 indicate that when students had no

TABLE II
Calculus Concept Test

Question	Percent Correct
1. Suppose f, f' and f" are differentiable and that $f(5) = 9$, $f'(5) = 4$ and $f''(5) = -1$. Find $\lim_{h \to 0} \frac{f(5 + h) - f(5)}{h}$.	22%
2. Suppose line L is tangent to the curve $y = f(x)$ at the point (5,3) as indicated at the right. 	
Find (a) f(5)	63%
(b) f'(5)	22%
3. Evaluate $\int_{-1}^{2} [x]\, dx$	7%

"formula", they had nothing to fall back on. Their concept of the Riemann integral did not allow them any alternative approach to the problem.

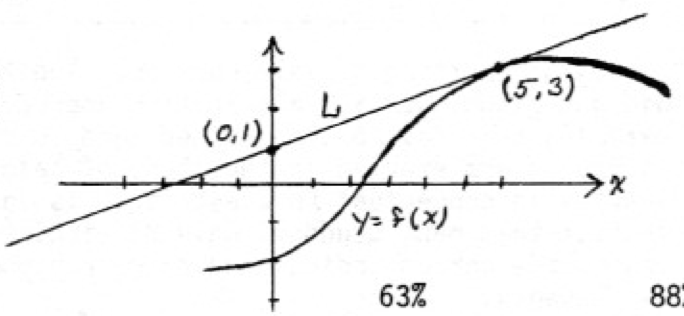
The examples that have been examined tend to confirm the contention that even when students are emersed in the study of calculus, our instructional emphasis results in conceptual knowledge that is indeed minimal. The evidence may also suggest that many students have difficulty in relating geometric explanations to the corresponding algebraic or symbolic representations of fundamental concepts.

A Programming Example. Because many students are now entering colleges and universities with experience in a programming language, we believe that having students write computer programs could be helpful in developing conceptual schemas related to the fundamental ideas of the introductory calculus course. We have instituted a one-credit supplemental course in computational calculus where students write their own programs to find limits, right and left hand derivatives, Riemann integrals, solutions to equations, etc. These programs, used in conjunction with graphics utilities (which students do not write) are used by the students to investigate several questions, among them the existence of derivatives and integrals for various functions at specified points or on specified intervals. Other types of questions are also investigated. It is quite easy, for example, to consider functions with discontinuities and examine the limit of Riemann sums when using left endpoints, right endpoints, midpoints or randomly selected points in the subintervals. By running a few examples, and examining the numerical output, students are easily convinced that the integral is independent of the points selected.

Table III lists several questions that were given on the Calculus Concept Test to those same students referred to earlier in this paper, and also to a group of 24 students who completed the computational calculus course. As can be seen from these results, students who were enrolled in the supplemental course appear to have developed richer schema related to both the derivative and integral concepts.

Justification of programming. Three seemingly logical arguments are often advanced for justifying the use of "canned programs" in calculus instruction. One argument draws a parallel between the discrete nature of computers and students' experiences with discrete situations. Discrete numerical output belongs to the students' world, and is more likely to be properly interpreted than are the symbolic representations of continuous functions. Second, the graphics capability, and in particular the ability to

TABLE III

Concept Test (Percent Correct) Question	Regular Class	Regular + Programming Class
1. Suppose f, f' and f" are differentiable and that $f(5) = 9$, $f'(5) = 4$ and $f''(5) = -1$. Find $\lim_{h \to 0} \frac{f(5+h) - f(5)}{h}$	22%	63%
2. Supose line L is tangent to the curve $y = f(x)$ at the point (5,3) as indicated at the right. Find 		
(a) $f(5)$	63%	88%
(b) $f'(5)$	22%	46%
3. Evaluate $\int_{-1}^{2} \lvert x \rvert \, dx$	7%	50%
4. If $f(x) = (x + 1)^{10}$, find find $\lim_{h \to 0} \frac{f(h) - f(0)}{h}$	8%	42%
5. What is the maximum slope of the curve $f(x) = -x^3 + 3x^2 + 9x - 27$	6%	46%

6.	Suppose P is a partition of $[0,\pi/2]$ into n subintervals, and u_i is an arbitrary point in the ith subinterval $[x_{i-1},x_i]$. Explain why $\lim_{\|P\| \to 0} \sum_{i=1}^{n} (\cos u_i) \Delta x_i = 1$.	10%	67%

produce dynamic representations, provides a new dimension that was not previously available. The human mind appears to be quite adept at processing visual information of this type. And finally, the use of these software packages in an experimental mode creates an environment that is more conducive for learning. Detailed elaborations of these arguments can be found in the literature, and there is beginning to appear a body of supporting research.

But what about the effects of programming itself. We believe the following arguments lend support to the hypothesis that writing computer algorithms can enhance the understanding of fundamental concepts and help in developing mathematical maturity.

1. In constructing their computer programs to find limits, left and right hand derivatives, Riemann integrals, etc., students are forced to deal with the definitions at a more basic level, at a step beyond what is required when merely examining numerical or graphical output from "canned programs". They must pay attention to the language of mathematics, and deal with the association between that language and their own conceptions.

2. The language used in programming can help bridge the gap between natural language and the formal language of mathematics. Understanding a sequence of programming code may enhance understanding of the associated mathematical symbolism. Particularly when working cooperatively in a laboratory situation, students have available another means for communication, another language with which they may feel more comfortable.

3. When writing a computer program, students are put into a situation in which they are doing the teaching. They are teaching the computer what to do. As most teachers will attest, teaching a topic to someone else requires an increased precision in ones knowledge of the topic, in terms of both comprehension and expression.

4. When students use instruments (programs) of their own creation, their mathematical investigations become intrinsically more interesting and exciting. There is a creation of conflict when output does not conform to expectations, whether from naive judgements, analytic or graphical analysis, or by an instructors edict. Might there be added incentive to reconcile this conflict when ones own creation is in error?

5. Creation of a satisfactory computer algorithm requires a degree of planning, and creation of a sequence of logically developed steps that lead to some intelligible results. Consequently its creation is not unlike developing a formal mathematical proof. Since many Calculus instructors have given up on requiring students to produce proofs, the development of computer algorithms may be the next best thing.

It may be that the production and examination of computer algorithms can put a little more excitement and joy into both teaching and learning elementary calculus. And though we have nothing against the use of "canned programs" (they can be very beneficial), we believe that when students can develop their own algorithms, their efforts are more akin to "doing mathematics."