T-Shirt Summaries

Gary T. Leavens
Department of Computer Science, 229 Atanasoff Hall
Iowa State University, Ames, IA 50011

September 20, 2006

Abstract

This note discusses a novel approach to showing students my enthusiasm for course material: the use of fabric paint on T-shirts.

1 Context

I originally made T-shirts when teaching the course "Introduction to Computer Programming" in Fall semester 1993. An important objectives for this course was to get students interested and excited about the material. In addition to other ideas, I tried to do this by making fabric-painted T-shirts.

2 T-shirts

I got the idea of making fabric-painted T-shirts idea from an eminent Computer Scientist, Philip Wadler of Edinburgh University. He had used such T-shirts at conferences to illustrate the central idea of each of his talks. When I asked, he was happy to tell me how to make such T-shirts using fabric paint[1], and also mentioned that he used them sometimes in his classes.

2.1 Course Motto

My first T-shirt was based on our course's motto: "never write the same thing twice". I actually made a sweatshirt with a design featuring a large lower-case lambda (λ).[2], which looks something like Figure 1. On the back is an example procedure, which is the central example of the course. (The procedure is `flat-recur`, which is Program 7.23 in the course textbook [2].)

2.2 Course Concepts

After making the course motto sweatshirt, I decided to make a T-shirt for myself and for each teaching assistant (TA) that would capture my hopes for the course, as well as the first concept to be taught in the class. I felt it was important to make a T-shirt for the TAs, so that my enthusiasm would not be filtered out by them, but multiplied by their enthusiasm. The T-shirts I made for myself and the for this idea look something like Figure 2. I gave such a T-shirt to each TA at our first meeting. These T-shirts are aimed at accomplishing the following goals.

- To raise the spirit of the course staff and to start them thinking of the staff as a team. Hence the athletic look of the back of the T-shirts.
- To remind myself, the TAs, and the students that the course should be, as the textbook authors put it, "a creative and literate endeavor" (from the first sentence of the preface of [2]) and not just drill.
- To explain the theory of quotation and symbols, which is the very first topic in the

[1] Such fabric paints are available very cheaply at a fabric store.

[2] This motto says why functional abstraction, done with `lambda` in the programming Scheme used in the course, is important. That is, functional abstraction allows one to avoid writing the same thing twice.

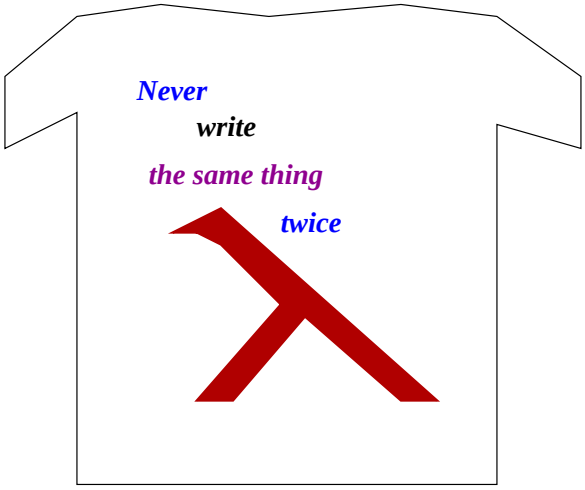

Figure 1: The front of a sweatshirt with the course motto on it. In the original, "Never" and "twice" are in blue, "write" is in black, "the same thing" is in purple, and the giant λ is in bright red. The sweatshirt itself is grey.

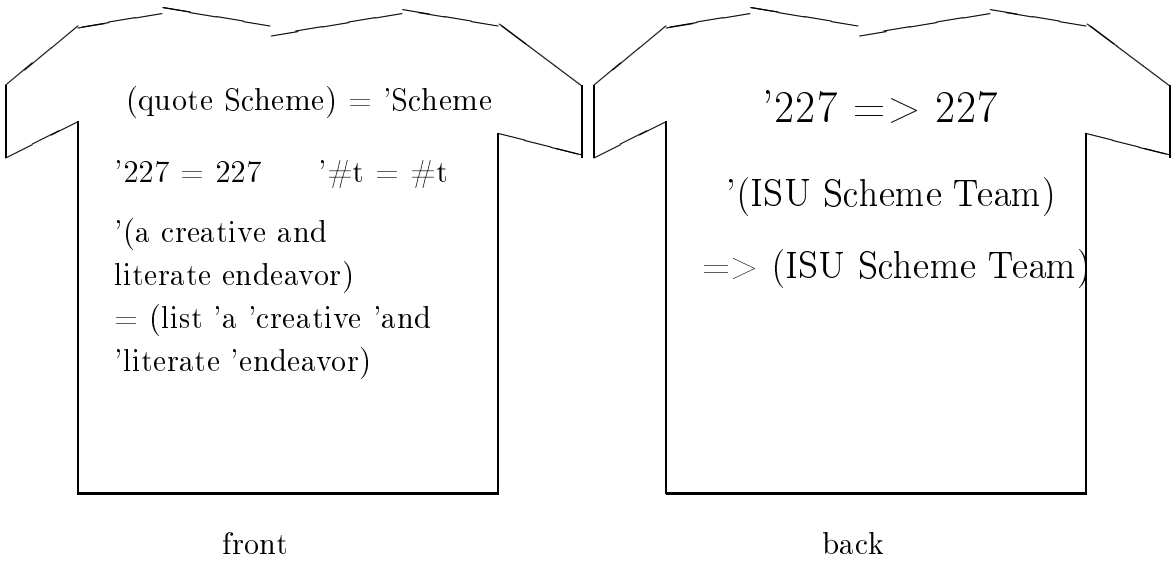

Figure 2: A T-shirt that was given to each TA and worn by me for the first lectures on Scheme. In the original, "quote" and the quotation marks (') are in blue, the equals signs (=) and $\Rightarrow$ are in bright green, the other text is in black, except for the text on the back which follows the $\Rightarrow$s, which is in red. The T-shirts themselves were white.

course text. The front of the T-shirt gives equations for the quotation operator, shows that numbers and booleans (like #t, which means "true") are self-evaluating (do not need to be quoted), and shows that quotation of a list is syntactic sugar for forming a list of the quoted items of the list. The back of the T-shirt gives an operational semantics of quotation; inspired by [1], one thinks of quotation as returning its argument painted red. (This was tied into the lecture, where I used a Scheme interpreter that had that behavior.)

2.3 Use During the Course

In Fall 1993, I made a T-shirt for every major idea in the course. I wore such a T-shirt coordinated with each idea on each day I taught about Scheme. (But I did not wear one during the exams!) The following list gives an idea of what was on such T-shirts.

- Important programming examples, often drawn from the textbook.
- Equational rules for some construct in the language, or some type of data. This is illustrated by the front of the shirt shown in Figure 2. (These were prominent in the first part of the course, where new kinds of data and language features are often introduced.)
- Operational semantics for some construct in the language, or some type of data. A very simple is given by the back of the shirt shown in Figure 2.
- Diagrams illustrating some key idea. For example, one T-shirt about data abstraction contains on its front something like the following diagram.

$$\begin{array}{ccccc} (\frac{1}{2},\frac{3}{4}) & \xrightarrow{r+} & \frac{5}{4} & & \text{values} \\ \uparrow & & \uparrow & & \\ \texttt{((2 1),(4 3))} & \xrightarrow{\texttt{r+}} & \texttt{(8 10)} & & \text{objects} \end{array}$$

  This diagram illustrates how the abstract values (such as 1/2) are represented by the objects in a program (such as the list `(2 1)`).
- Mottos summarizing some way to program or think. For example, the T-shirt for (flat) recursion features the three questions we use to help students learn to think recursively (which are adapted from [3]).

  1. What is at the end?
  2. What is one step?
  3. How can that step help solve it?

All told I made up 15 different T-shirt designs for the class, or roughly one for each week of the class. Each T-shirt (and the one sweatshirt) has something on the both the front and the back.

There are several advantages to making up these T-shirts and wearing them in class.

- Since T-shirts are used in popular culture to display ideas one wishes to promote, students intuitively understood that I believed in and thought highly of the ideas in the course. That is, they were an effective way of communicating my enthusiasm for the course material. Students would often ask me about them, which would give me a chance to explain why I had chosen the particular ideas or examples for the T-shirt. (Others in the department and in my daily life would also ask about them, gaining visibility for Computer Science.)
- An unanticipated but major benefit of making the T-shirts was that they focused my attention on the most important ideas of each part of the course. Not only would I have to select the ideas or examples for each part of the course, but I would spend time thinking about how best to illustrate them. This contributed in no small measure to my explanations of course material, and to the focus of the course as a whole.
- Students had a point of reference for each unit of the course, as they could try to relate the material to my T-shirts. This often provided a puzzle for them, but once they saw the key idea, it gave them a way to remember it. This probably works best with key programming examples; once a student

understands that some example is key, he or she can see how it relates to other examples, and has a way to organize the information.

- The change of T-shirts (not during class, usually!) shows students that the course has moved on to a new unit.

- I often deepened my understanding of some point of the course material by the necessity to fit some key idea onto a single T-shirt, or by my desire to present the idea in the most elegant way possible. For example, the equational theory of lists is usually presented as the following three equations.

$$\begin{aligned} \texttt{(car (cons } x\ l\texttt{))} &= x \\ \texttt{(cdr (cons } x\ l\texttt{))} &= l \\ \texttt{(cons (car } l\texttt{) (cdr } l\texttt{))} &= l \end{aligned}$$

  The only trouble with this is that the last equation does not hold for all lists; it is really a conditional equation, that only holds for non-empty lists. When I thought of writing a side-condition on the T-shirt, it struck me as unsatisfactory. I also realized that I would have to describe what the test for an empty list (`null?`) means in a separate equation. Then I realized that I could use a conditional equation instead of the last equation, which follows.

$$l = \texttt{(if (null? } l\texttt{) '()}\ \texttt{(cons (car } l\texttt{) (cdr } l\texttt{)))}$$

  This is not new, but it is elegant.

3 Evaluation

This technique of making up T-shirts is not without its problems. The main problem is that it takes a great deal of time, energy, and planning to make such T-shirts. While I could usually come up with an idea by reviewing my lecture notes, it takes two days from the time one starts on a T-shirt until it can be worn, because each side has to dry for 24 hours. And of course, one student expressed a dislike for the T-shirts in the course evaluations (perhaps he or she thought that I was demeaning myself?), but this student was a minority of one in a class of 120.

Another problem is that it seems best suited to undergraduate courses. While I have used it somewhat in my graduate courses, the pace of ideas in such a course is too fast to make up T-shirts for all major ideas. Instead I sometimes reuse T-shirts from undergraduate courses if they are appropriate.

In addition I have used such T-shirts for my technical talks at conferences and universities. People are now disappointed if they do not see me with such a T-shirt at a technical talk!

4 Conclusion

If the students did not have interest in the course and its subject matter, they do not learn as much. One way of conveying my interest and excitement to students is by using fabric-painted T-shirts.

References

[1] Iain Ferguson, Edward Martin, and Burt Kaufman. *The SCHEMER'S Guide.* Schemers Inc., Fort Lauderdale, FL, 1990.

[2] George Springer and Daniel P. Friedman. *Scheme and the Art of Programming.* McGraw-Hill, New York, NY, 1989.

[3] David S. Touretzky. *The Mathematics of Inheritance Systems.* Morgan Kaufman Publishers, Inc., Los Altos, Calif., 1986. Revision of the author's doctoral thesis at Carnegie-Mellon University, 1984.