

Continuing the Comparison Between Graphical- and Text-based Programming Instruction

Kathleen A. Harper, Richard J. Freuler, John T. Demel, Stuart H. Brand
Engineering Education Innovation Center
The Ohio State University, Columbus, OH 43210
Email: harper.217@osu.edu

Initial Study

Previous work described a comparison study conducted in 2008 between students who learned introductory programming in either a text-based or graphically-based format.¹ In that study all participants were first-year engineering majors enrolled in the Fundamentals of Engineering for Honors program at The Ohio State University.

Students enrolled in the second quarter of the three-quarter sequence had the option of enrolling in a pilot section of programming fundamentals taught using primarily LabVIEW instead of the traditional C/C++ approach. Fourteen students enrolled in this section, and all gave their consent to be included in the study. From the over 200 students enrolled in parallel sections taught using C/C++, a matched comparison group was selected, based upon previous programming experience, intended major, and, where possible, gender. The courses were structured such that they addressed the same basic programming constructs, including basic arithmetic, repetition selection, and data structures. Both cohorts learned MATLAB for two weeks following the initial eight weeks of instruction in LabVIEW or C/C++. A brief illustration of how a simple problem can be approached in either LabVIEW or C/C++ is included as an appendix to this paper.

To summarize the results of the previous paper, three comparisons of the groups were done. The first focused on the students' ability to apply the language used in their programming instruction to problem solving. To do this, the final exam for each class contained an identically worded problem, solvable in either LabVIEW or C/C++. The C cohort outperformed the LabVIEW cohort by about 10%.

A second area of interest was how well each cohort learned a subsequent language. Since both courses ended with the same MATLAB instruction, the final exams contained two identical MATLAB problems. Both cohorts performed equally well on the MATLAB component of the exam.

Finally, a comparison of student epistemological views was carried out. To develop a suitable measurement instrument, a subset of questions from the Maryland Physics Expectations Survey (MPEX)² were modified to pertain to programming and administered as a pre- and post-survey. The student responses were compared to those of an expert group of programmers, some of whom were familiar with LabVIEW and some of whom were not. In the pre-test administration, the two cohorts were indistinguishable from each other and clearly different from the experts.

By course-end they clearly differed from each other and the LabVIEW cohort bordered on being indistinguishable from the experts.

Subsequent Study

To determine whether these results were robust or not, the study was repeated in 2011 with a larger sample (N=25 in each cohort). The students in this second sample were very similar to those in the 2008 sample, as measured by initial surveys on programming experience and epistemological views. The instructors of record for the courses were the same as in the original experiment, and the instructional materials were largely unchanged. The measurement techniques were identical to those in the first study. This was possible since final exams are never returned to the students in this course.

When the facility with the language of instruction was investigated, there was no significant difference in the performance of the two groups. This is different from the initial study, in which the C cohort outperformed the LabVIEW cohort. In another difference from the first study, the C cohort significantly outperformed the LabVIEW cohort in learning MATLAB ($p < .002$, Wilcoxon signed-rank test). The epistemological survey analysis showed that the two cohorts began and ended the quarter indistinguishable from each other and clearly different from the experts ($p < .001$). In short, none of the results from the previous study were replicated. These details are summarized in Table 1.

Table 1 Summary of Results of LabVIEW vs. C Comparison Studies

Area of Investigation	2008 (N=14)	2011 (N=25)
Facility with original language	C outperformed LabVIEW	No difference
Facility with MATLAB	No difference	C outperformed LabVIEW
Development of expert epistemology	LabVIEW outperformed C	No difference

Discussion

One conclusion to be drawn from these two studies is that it is possible to teach programming fundamentals with LabVIEW. A second conclusion is that while teaching programming via LabVIEW can have a profound positive impact on student programming epistemologies, it is not automatic. While it was posited after the 2008 study that the LabVIEW environment had something to do with this effect, the results from 2011 showed that this is not a sufficient condition. The initial study also showed it is possible for students who learn programming using a graphical interface to learn a second text-based language equally as well as those whose first language was text-based. Again, the second study showed that this effect is not guaranteed.

Since none of the new results were consistent with those in the first study, some further data analysis attempted to determine a possible cause for the difference. A comparison of the initial epistemological survey results showed no significant attitudinal or belief differences between the

populations in the two studies. An inspection of pre-course questionnaires used to match the samples, where students described their programming background, also revealed no noticeable differences between the two populations.

Additionally, there were no significant instructional changes to either course. The materials and course structure were the same in both years, and the professors were the same. The most major difference between the two years [was](#) the teaching assistants, who were in the classroom the entire instructional time. The teaching assistants for the initial study, in addition to being highly regarded instructors, had significant background with both C and LabVIEW. The assistants in the second study were only experienced in LabVIEW. This appears to be the most likely cause of the difference between the two studies' results, but it is not possible at this time to determine if this is the case or not.

Future Work

Initial efforts are already underway to rigorously refine the epistemological survey, as there is a need for this sort of instrument in the programming arena. As of right now, any programming concept inventories that have been developed contain a language dependence that is difficult to eliminate from the questions. An epistemological survey, being naturally independent of programming language, is a good candidate for enabling cross-language comparisons. This will not only aid the research described here, but hopefully that of many others.

It is possible that a third similar, yet not identical, comparison study could be conducted in the autumn of 2013. Due to factors external to this work, the course is currently undergoing a major redesign. Ohio State has switched to semesters, and now the MATLAB portion of the course is conducted prior to the LabVIEW or C content. While this could make it challenging to repeat the study, it should not prevent the research team from collecting further relevant data.

Acknowledgements

This project was supported by the National Instruments Foundation. Any opinions expressed in this paper are solely those of the authors.

Bibliography

1. Richard J. Freuler, John T. Demel, Kathleen A. Harper, Stuart Brand, Craig Morin, & Patrick Wensing, "Comparing the Use of a Graphical Programming Language to a Traditional Text-Based Language to Learn Programming Concepts in a First-Year Course," *Proceedings of the 2009 American Society for Engineering Education Annual Conference*, Austin, TX, June 14-18, 2009 (ASEE, 2009).
2. Redish, Steinberg & Saul, "Student Expectations in Introductory Physics," *American Journal of Physics* 66, 1998, 212-224.

Appendix: LabVIEW vs. C/ C++: An Illustration

Figures A1 and A2 contrast how a problem might be approached in either C/C++ or LabVIEW. For this illustration, the problem is to write a program to request the user to input two numbers, then output the sum and product of those numbers.

```
#include <stdio.h>
int main ( )
{
    int A, B, AplusB, AtimesB;
    printf ("Input 2 numbers, A, then B");
    scanf ("%d%d", &A, &B);
    AplusB = A + B;
    AtimesB = A * B;
    printf ("A + B = %d\n", AplusB);
    printf ("A x B = %d\n", AtimesB);
    return 0;
}
```

Figure A1. Sample Program in

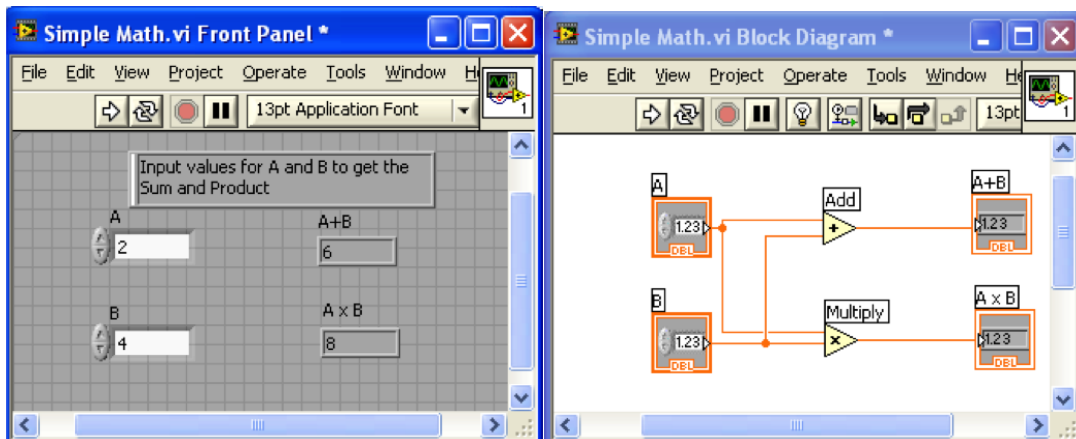


Figure A2. Sample Program in LabVIEW