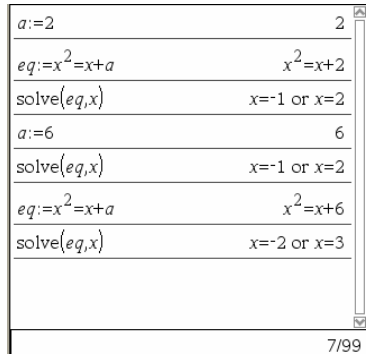
CRITICAL ISSUES OF IMPROVING COMPUTER ALGEBRA SYSTEMS *

Djordje Kadijevich
Megatrend University and Mathematical Institute SANU, Belgrade, Serbia
djkadijevic@megatrend.edu.rs

Abstract. In order to improve Computer Algebra System (CAS), there is a need not only for better built-in features to be implemented, but also for better conditions for creating and using user-defined functions to be provided. By using the TI-Nspire CAS tool and examples from high school mathematics, this paper examines the following critical issues of improving CAS: updatable lines with commands and functions, controllable execution of commands and functions, richer tool documentation, stronger links between different representations, richer opportunities for creating and using user-defined functions, and customizable user interface. These critical issues should be adequately addressed in a future version of the tool.

INTRODUCTION

Gjone [1] examines the possibilities and limitations of the Casio ClassPad tool for critical topics of determining equivalence of algebraic expressions, relating different representations, and making CAS techniques transparent and congruent with their paper-and-pencil versions. He finds that in order to improve CAS, not only do better built-in features need to be implemented, but also better conditions for creating and using user-defined functions need to be provided. Acknowledging the relevance of these requirements, this contribution examines critical issues related to these requirements. The issues deal with updatable lines with commands and functions, controllable execution of commands and functions, richer tool documentation, stronger links between different representations, richer opportunities for creating and using user-defined functions, and customizable user interface. These six issues are exemplified in the next section by using examples from high school mathematics that are mostly worked out on the TI-Nspire CAS tool (www.ti-nspire.com).



Screenshot 1

SIX CRITICAL ISSUES

Updatable lines with commands and functions

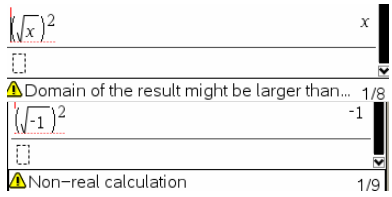
Contrary to potential user expectations, TI-Nspire CAS does not support editing expressions in the calculator history. It is only possible to copy all or part of an expression from that history, paste the material to the entry line and then edit the material, if needed. As the result of such a limited input, the outcome of the new calculation in the entry line does not affect the outcomes of calculations in lines above that line. For example, when parameter *a* is made equal to 6 (Screenshot 1,

* *MathEduc SC*: N84, P54, U74; *MSC 2000*: 00A35, 97-04, 97C80, 97U70. *Keywords and phrases*: computer algebra system, software design, software evaluation, upper secondary.

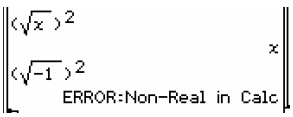
line 4 as counted from the bottom of the screen), the value of equation *eq*, defined as $x^2 = x + a$ and then displayed in line 6 using $a = 2$ from line 7, is not updated to $x^2 = x + 6$, and therefore the zeros in question are not updated in line 3 to -2 and 3. In order to obtain the desired results, two lines (*eq* := $x^2 = x + a$, solve(*eq*, *x*)) are copied and executed again. [Note: Although Casio ClassPad behaves in the expected way (with respect to on-screen editing, and subsequent updating of existing results in lines below the updated line), the content of its geometry links are not updated, however (see [1], Screenshot 4)].

Controllable execution of commands and functions

Knowing that TI-Nspire CAS automatically simplifies expressions when working with complex numbers [2], it is easy to understand why $\left(\sqrt{x}\right)^2$ simplifies to *x* and why the produced value of $\left(\sqrt{-1}\right)^2$ is -1. A good thing is that each of these results—obviously a wrong one in the domain of real numbers—is coupled with an appropriate warning in real mode. These warnings are "Domain of the result might be larger than domain of the input" and "Non-real calculation", respectively (see Screenshot 2). However, TI-Nspire CAS does not save the warnings when it saves a document containing the results.

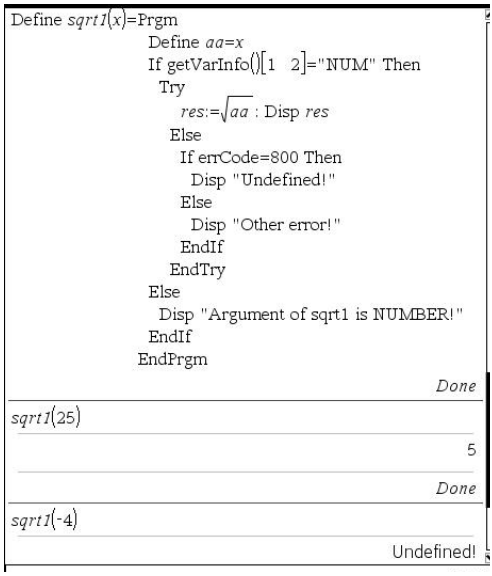


Screenshot 2



Screenshot 3

The user can choose to work with only real numbers (by using option File/Settings/ Document Settings…), but the results are the two reported previously not the following two: "*x* for $x \geq 0$" and "Undefined". (In real mode, TI-Nspire CAS returns "ERROR: Non-real result" for $\sqrt{-1}$, whereas it displays "undef" for log(0).) As regards the automatic simplification, the user cannot turn it on and off to obtain right answers in subtle cases by using skilfully created user-defined functions. Screenshot 3 presents answers to the two questions by Casio ClassPad.



Screenshot 4

Controllable execution of commands includes managing errors. Although TI-Nspire CAS enables managing errors (see Screenshot 4), the

TI-Nspire CAS Reference Guide does not provide codes of the warning messages. Note that ClassPad 330 User's Guide contains tables with error messages and warning messages, but codes for these messages and system variables handling them are not given, and managing errors is thus not possible.

Richer tool documentation

Extending the conversation about managing errors, this third critical issue deals with improperly explained CAS features and unexplained CAS features.

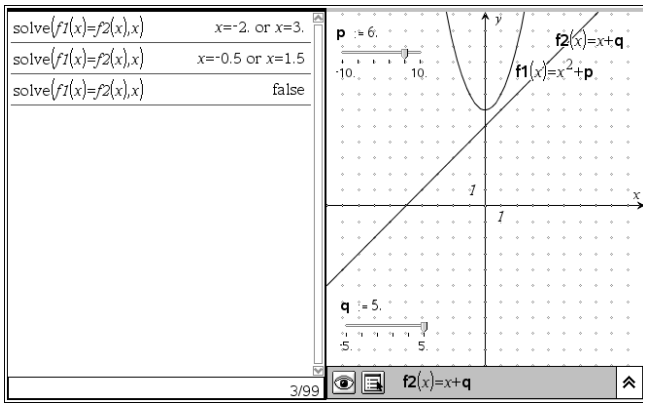
Some commands or functions may give answers that differ from those described in the official documentation. Consider, for example, an implementation of if-then-else-error function: the Casio ClassPad function piecewise wrongly returns 1 for **piecewise**(13, 1, 2, 3), whereas TI-Nspire CAS correctly returns 3 for **when**(13, 1, 2, 3). Furthermore, Casio ClassPad correctly returns "No solution" for both **solve**($x^2 = 4$, a) and **solve**($x^2 = 4x$, a), whereas TI-Nspire CAS wrongly returns $x = 2$ or $x = -2$ for **solve**($x^2 = 4$, a) and even $x = 2\sqrt{x}$ and $x \geq 0$ or $x = -2\sqrt{x}$ and $x \geq 0$ for **solve**($x^2 = 4x$, a).

Official manuals may not describe several useful commands and functions that can be used. Try, for example, **completeSqr**($x^2 - 2x - 8$) on Casio ClassPad. (Such a command or function on TI-Nspire CAS has not been discovered.) Although hiding some features may be a defendable business policy, disclosing some of these features would help us improve our use of the tool.

The previous section examines the question of managing errors from the programmers' point of view. An ability to manage errors depends on detailed information regarding errors, which should appear in CAS official manuals.

Stronger links between different representations

Despite a powerful ClassPad feature called Geometry Link, links between algebraic and geometry representations in two windows (i.e. applications) are not strong because the content of geometry links cannot be updated automatically (see [1]). Links between two applications are also not strong for TI-Nspire CAS because, as Screenshot 5 displays, new values for two parameters p and q (assigned with sliders) require another execution of the same solve command. Better links may simply require an update command that would update the content

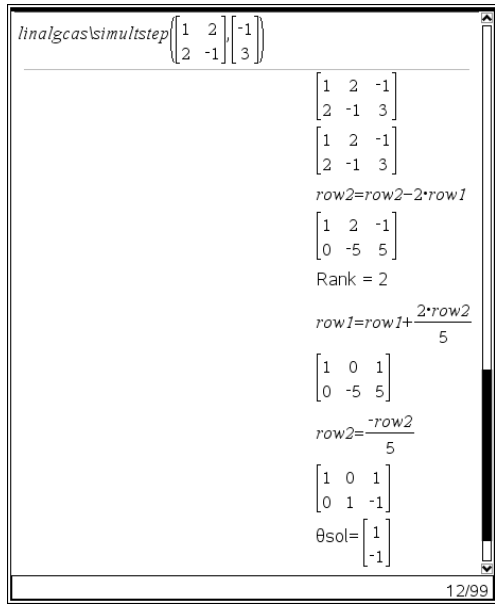


Screenshot 5

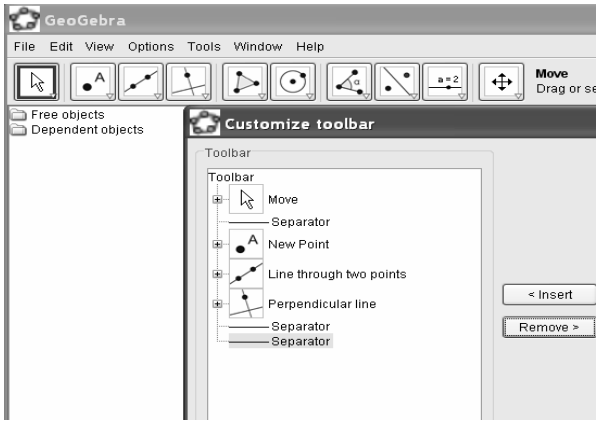
of other applications (here the Calculator application) to reflect the changes in the active application (here the Graphs & Geometry application).

Richer opportunities for creating and using user-defined functions

Opportunities for creating and using user-defined functions should include creating functions in a program-like fashion (with managing errors) and using these functions from libraries of functions. Screenshot 4 presents an example of such a created function. A step-by-step solution of system of two linear equations, $x + 2y = -1$ and $2x - y = 3$, is given in Screenshot 6. This solution is found by the application of function *simultstep* from library *linalgcas* (i.e. document *linalgcas* is stored in a designated library folder such as My Documents\TI-Nspire\MyLib). Both Screenshots 4 and 6 present the opportunities in question offered by TI-Nspire CAS, which should nevertheless include managing warning messages, better control of command and function executions, and stronger links between different representations in different applications. As noted in [1], opportunities for creating user-defined functions are very limited in Casio ClassPad.



Screenshot 6



Screenshot 7

Customizable user interface

An educationally oriented CAS such as Casio ClassPad and TI-Nspire CAS may be used for several years during secondary and tertiary education. In early years of this use, just small subsets of CAS commands and functions would be utilized. As CAS affordances would grow as the mathematics known by its user grows, CAS should have a customizable user interface. Screenshot 7 presents such an interface available in GeoGebra, which is dynamic geome-

try software including, among others things, some CAS features (see www.geogebra.org) This kind of interface, not available in Casio ClassPad and TI-Nspire at present, should deal both with built-in and user-made affordances.

CLOSING REMARKS

In order to have CAS that is a pedagogical tool, CAS should make its techniques transparent and congruent with their paper-and-pencil versions [3,4], distribute computational tasks both to CAS and its user [3], provide step-by-step solutions [5], take care about subtleties such surplus or missing solutions in solving equations [6], update the outcomes reflecting changes in linked entities (cf. [3]), and have a customizable user interface [3]. By taking an integrated perspective of these requirements, this paper examined six critical issues of improving CAS that deal with both built-in and user-made features. Further research may examine these issues in more detail to help CAS manufacturers make appropriate decisions concerning future CAS development. By using these and other relevant critical issues, further research may also study the process of instrumentalization [7] regarding the development of user-made features.

ACKNOWLEDGEMENT

This paper results from the author's work on projects 144032D and 144050A funded by the Serbian Ministry of Science.

REFERENCES

[1] Gjone, G. (2009). The use of CAS in school mathematics: Possibilities and limitations. Contribution to CAME 6, Megatrend University, Serbia. Retrieved 10 May 2009 from www.lkl.ac.uk/research/came/events/CAME6/Index.html

[2] Bosse, M. J., & Adu-Gyamfi, K. (2009). TI-Nspire calculators: Better but still not perfect. *Mathematics Teacher*, *102*(6), 464-467.

[3] Kutzler, B., & Kokol-Voljc, V. (2003). A plead for PECAS (= PEdagogical CAS). In A. Gagatsis & S. Papastavridis (Eds.), *3rd Mediterranean Conference on Mathematical Education: Mathematics in the modern world, mathematics and didactics, mathematics and life, mathematics and society* (pp. 331-334). Athens - Nicosia: Hellenic Mathematical Society & Cyprus Mathematical Society.

[4] Steiner, J. M., & Dana-Picard, T. (2004). Classroom note: Teaching mathematical integration: human computational skills versus computer algebra. *International Journal of Mathematical Education in Science and Technology*, *35*, 249-258.

[5] Tõnisson, E. (1999). Step-by-step solution possibilities in different computer algebra systems. *Proceedings of ACDCA Summer Academy: Recent Research on DERIVE/TI-92-Supported Mathematics Education* (Gösing, Austria). Retrieved 10 May 2009 from www.acdca.ac.at/ kongress/goesing/index.htm

[6] Kadijevich, Dj. (2002). Towards a CAS promoting links between procedural and conceptual mathematical knowledge. *The International Journal of Computer Algebra in Mathematics Education*, *9*(1), 69-74.

[7] Trouche, L. (2005). An instrumental approach to mathematics learning in symbolic calculator environments. In D. Guin, K. Ruthven, & Trouche, L. (Eds.), *The didactical challenge of symbolic calculators: Turning a computational device into a mathematical instrument* (pp. 137-162). New York: Springer.