Experimental report – Educational Usability Evaluation of Building Algorithm Animations with WinHIPE

Jaime Urquiza-Fuentes

jaime.urquiza@urjc.es

ViDo Research Group, ESCET, Rey Juan Carlos University.

Móstoles, Madrid, Spain.

Table of contents

Introduction

This report describes the evaluation of WinHIPE [3] from the educational usability point of view. The main objective is the use of the viewing and building processes of web-based algorithm animations with WinHIPE [2]. This is not only a pedagogical effectiveness evaluation, in addition, we evaluate efficiency, and student's subjective opinion, thus we evaluate three important aspects of usability: effectiveness, efficency ans user's satisfaction.

The rest of the report is structured as follows. The section two describes the evaluation: participans, variables studied, method and produre. Then results of the evaluation are shown in section three. Finally, in section four, conclusions and future work are described.

Description of the experiment

This evaluation attempts to find if there is any performance difference between viewing algorithm visualizations (viewing engagement level) and constructing algorithm visualizations (constructing engagement level). Improvements will be measured in terms of Bloom's taxonomy [1], with tasks related to the comprehension and application levels.

The context of this evaluation is an Algorithm Design and Analysis course (Fall semester, 2005) at the Rey Juan Carlos University [4], where a group of students had to build algorithm animations using our effortless approach. In the evaluation, the tree breadth traversal algorithm was used.

Subjects

Fifteen different subjects participated in the evaluation, thirteen were male and two female. Participation in the evaluation was voluntary. All of the subjects were students from the Algorithms course.

Participants were randomly divided in two groups: the control group and the experimental group (CG and EG respectively for the rest of the paper). Both groups were asked about their previous knowledge about the algorithm, only one student having previous knowledge. Therefore both groups belong to the same population and further results can be compared.

Variables

The independent variables of the evaluation were: pedagogical effectiveness and efficiency, and students' opinion about both approaches (viewing and building). The dependent variables were: the answers to a number of questions about the algorithm (mapped to the comprehension and application levels of Bloom's taxonomy), the time expended in studying the algorithm, the time used to complete a knowledge test about the algorithm, and the user's subjective opinion.

Learning improvements related to the comprehension level were measured with the following questions (questions 1 to 4 of the knowledge test, appendix D):

- What are the main ideas of this algorithm?
- Given the following tree, write the result of applying the algorithm to it.
- Given the following list, write the tree to which the algorithm was originally applied (note that the tree was balanced).
- Which is the existing relationship among the nodes of the tree, if the result is a strictly ascending ordered list?

Learning improvements related to the application level were measured with the following question: What modifications should be done on the algorithm to change the traverse direction to right-to-left? (question 5 of the knowledge test, appendix D)

Users' opinion was measured with a questionnaire where students were asked:

- if they thought that *building* (or *viewing*) algorithm animations had helped them in understanding the algorithm, and
- if they though that algorithm animations are easy to *build* (or *use*) with WinHIPE.

Method and procedure

The evaluation was divided into two sessions: a training session where the IDE was shown to the students, and the experimental session where knowledge about the algorithm was evaluated. Participation was ten and thirteen students respectively.

The training session was two hours long. The instructor demonstrated the tool, he generated two web-based animations with WinHIPE as an example, and students generated two more animations. The animations used were unrelated to the algorithm that would be used in the experimental session. None of the students appeared to have problems using the tool. At the end of this session a questionnaire about the tool (see appendix A) was completed by the students thus, we got their first impression about the tool.

The experimental session also was two hours long, and two weeks after the training session. First, we explained to the students that we were carrying out the evaluation, and that their participation would be voluntary. Next, we randomly formed the CG (n=7) and EG (n=6), and we checked that all students in the EG had attended the previous training session. Students of both groups were asked about their previous knowledge about the algorithm. Then, we gave the students all the materials they were allowed to use to study the algorithm, which was a textual description of the algorithm for both groups (see appendix B), and:

- a number of web-based algorithm animations to be viewed, built with WinHIPE (see http://vido.escet.urjc.es/winhipe/DAA/daaEng.htm), for the CG, and
- the source code (see appendix C) to build web-based algorithm animations, for the EG.

Students of both groups were asked to study the algorithm, using the materials, until they thought that they had enough knowledge about it. Then, they completed the knowledge test (see appendix D) and another questionnaire to collect their subjective opinion about their viewing/building learning experience (see appendix A for the EG, and appendix E for the CG).

Results of the experiment

Data collected from the questionnaires and the knowledge tests are available in the appendix F. To carry out the significant differences analysis, normality of data has been tested (see appendix G) Details of the statistical analysis related to the learning measurements can be found in the appendix H, and the details related to the time measurements can be found in the appendix I.

Effectiveness analysis

Learning effectiveness was tested by means of two levels of Bloom's taxonomy: comprenhension and application. Both levels were graded in the range [0.0,1.0]. Four questions were asked to test performance related to the *comprehension level*. The first question asked students to identify the main ideas underlying this algorithm; these ideas were: (1) operations with lists, (2) recursion, (3) left-to-right direction in tree traversing, and (4) accumulation of recursive operations with subtrees. Students of both groups performed the same identifying ideas (1) and (3), there was no significant differences for the idea (2) but p is close to the limit ($p=0.063$) moreover, we realized that some students cosidered this idea obvious, so many of them did not wrote it in the answer, therefore we have ignored it; finally idea (4) was identified by none of students in the CG, but by 83% of students (5/6) in the EG. We did not found significant differences in performance in the second, third and fourth questions. Thus, the average grades for the understanding level were 0.88 for the EG and 0.73 for the CG, a 16% of learning improvement.

We found significant differences ($p=0.03$) in the answers to the question related to the *application level*. The CG obtained an average grade of 0.33, while the EG obtainned an average grade of 0.77, a 60% of learning improvement.

Efficiency analysis

We also measured the time expended with the materials and the time used to complete the knowledge test. With respect to the time expended by students using the materials to study the algorithm, students of the EG expended an average time of 49 minutes, while students in the CG expended an average time of 18 minutes. Difference of this time between groups was significant ($p<0.05$). But no differences were found ($p=0.194$) in the time used to complete the knowledge test between the EG and CG.

Student's opinion

The students' first impression (after the training session) about WinHIPE and the building process was very good. None had previously used WinHIPE. All of them (*n=10*) thought that the web-based animations were easy to build, and that building animations would help them in understanding the algorithms.

This opinion was maintainned by students after the experimental session. Answers to the questionnaire about users' satisfaction showed that students in the EG agreed with both ideas: building algorithm animations helped them understanding the algorithm, and web-based animations were easy to build with WinHIPE. All of the students in the CG agreed with: web-based animations helped them in understanding the algorithm, and web-based animations were useful and easy to use. We also asked these students about what approach would be more helpful in learning algorithm concepts: viewing or building. 71% (5/7) thought that both approaches were equally helpful: two of them just said this, but three also said that both approaches should be used together.

References

1. B. Bloom, E. Furst, W. Hill & D.R. Krathwohl *Taxonomy of Educational Objetives: Handbook I, The Cognitive Domain*, Addison-Wesley, 1959.
2. J. Urquiza-Fuentes and J.A. Velazquez-Iturbide. Effortless construction and management of program animations on the web. In Rynson W.H. Lau, Qing Li, Ronnie Cheung, and Wenyin Liu, editors, *Advances in Web-Based Learning - ICWL 2005, 4th International Conference*, volume 3583 of *Lecture Notes in Computer Science*, pages 163-173, Hong Kong, China, 2005a. Springer Verlag.
3. WinHIPE web site: http://vido.escet.urjc.es/winhipe (6/4/2006)
4. Rey Juan Carlos Unversity web site: http://www.urjc.es/ (6/4/2006)

Appendix A. First Impression Questionnaire

Questionnaire about the generation of web-based animations with WinHIPE

1. How often have you generated web-based animations with WinHIPE.

a) Never.

b) Sporadically.

c) Frequently.

2. Building web-based algorithm animations with WinHIPE has helped me to understand the algorithm.

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

3. Building web-based algorithm animations with WinHIPE is an easy task.

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

4. A tool to manage collections of web-based algorithm animations will be useful.

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

Appendix B. Textual description of the algorithm

Problem description

Given a B binary tree, you are asked to build a list with the elements inside the nodes of the tree. Viewing the tree drawn with its root at the top and its leafs at the bottom, the resulting list must have the elements of the tree in a descending-level order. So the first element will be that in the root, then the element in the root of the left-side child subtree, then that of the right-side child subtree, and so on.

Algorithm description

This problem has not a trivial solution. On the one hand, it has to be solved with a repetitive process visiting all the nodes in the tree, on the other hand, the recursive structure of the tree can not be used as the structure for a recursive procedure.

Trying with some examples, it can be seen that, while some elements are added to the result, other elements in a number of subtrees are waiting for. Therefore, it will be convenient to have an auxiliar function to generate the list of elements from a list of trees.

Next is the clasic idea of the algorithm. Initially, the auxiliar function is called with an actual parameter that is a list with one element, the original tree. This function will be recursive because it will perform the repetitive process. The process is finished when the list is empty. If the list is not empty and the first element is an empty tree, this suibtree has been totally processed and the process can continue with the next element.The more general case will occur when the first element of the list is a non-empty tree. In this case, the result is formed with the root of the tree; but as the recursive result for its children correspond to the next level, it will be appended at the end of the list of pending trees.

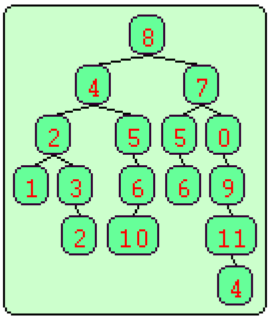
Appendix C. Source code

```
! -----------------------------------------------------------
!     File: TBT.HOP
! Description: tree breadth traversal
! -----------------------------------------------------------

data tree == empty ++ node (tree X num X tree);

dec breathL: list(tree) -> list (num);
--- breathL []                  <= [];
--- breathL (empty::trees)      <= breathL (trees);
--- breathL (node(empty,a,empty) :: ts) <= a :: breathL(ts);
--- breathL (node(empty,a,t2) :: ts) <= a :: breathL(ts<>[t2]);
--- breathL (node(t1,a,empty) :: ts) <= a :: breathL(ts<>[t1]);
--- breathL (node(t1,a,t2) :: ts) <= a :: breathL(ts<>[t1,t2]);

dec breath: tree -> list (num);
--- breath (tree) <= breathL [tree];
```

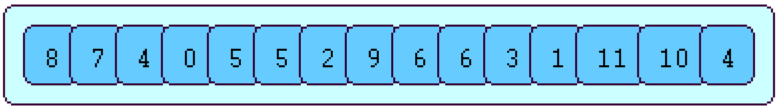
Appendix D. Knowledge Test

1. What are the main ideas of this algortihm?

2.Given this tree, write the result of applying the algorithm to it.



3.Given this list, draw the tree to which the algorithm was originally applied (note that this tree was balanced):



4. Which is the existing relationship among the nodes of the tree, if the result is a strictly ascending ordered list?

5.What modifications should be done on the algorithm to change the traverse direction to right-to-left?

Appendix E. Questionnaire for the CG

Questionnaire about the use of web-based animations generated with WinHIPE

1. How often have you used web-based animations from WinHIPE?

a) Never.

b) Just in the training session.

c) In the training session, and some more, i'm curious.

d) Frequently.

e) Always, when i have had to study an algorithm with web-based animations available.

2. Using web-based algorithm animations with WinHIPE has helped me to understand the algorithm.

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

3. What do you think that is more helpful to understand the algorithm?

a) Building animations is better than viewing them.

b) Viewing animations is better than building them.

c) Both are the same helpful, but i think that both should be used.

d) Both are the same helpful.

e) Without opinion.

4. Web-based animations are easy to use.

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

5. Web-based animations are useful.

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

6. The possibility of using various formats of the web-based animations is useful.

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

7. If you think that some of the componentes of the animation are not needed, mark it.

a) Problem description.

b) Algorithm description.

c) Source code.

d) Animation.

8. ¿Would you add something to the web-based animations?

9. Enumerate negative aspects of the web-based animations.

10. Enumerate positive aspects of the web-based animations.

11. A tool to manage collections of web-based algorithm animations will be useful:

a) Totally agree.

b) Partially agree.

c) Neither agree nor disagree.

d) Partially disagree.

e) Totally disagree.

Appendix F. Data collected

	USAB 1				GRP	TIME		TEST								USAB 2							
	P1	P2	P3	P4		STUD	TEST	P1 P11	P1 P12	P1 P13	P1 P14	P2	P3	P4	P5	P1	P2	P3	P4				
AL1	A	B	B	B	E	39	17	0	0	0	0	1,0	1,0	1,0	1,0	C	A	B	A				
AL2	A	B	A	B	E	50	24	1	1	1	1	1,0	1,0	0,7	1,0	B	B	B	B				
AL3	A	A	B	B	E	50	23	1	1	1	1	1,0	1,0	1,0	0,3	B	B	B	B				
AL4	A	B	A	A	E	51	17	1	0	1	1	1,0	1,0	1,0	0,3	B	C	B	B				
AL5	A	B	B	B	E	52	25	1	1	0	1	1,0	1,0	1,0	1,0	B	B	B	B				
AL6	A	B	A	D	E	52	12	1	1	0	0	1,0	1,0	0,5	1,0	B	B	B	E				
																P1	P2	P3	P4	P5	P6	P7	P11
AL7	A	A	B	A	C	14	8	1	1	1	0	1,0	1,0	1,0	0,3	C	A	C	A	A	A		
AL8	A	B	B	C	C	16	21	1	0	1	0	1,0	0,5	0,0	0,3	C	B	C	B	B	A	B	A
AL9					C	18	13	0	0	0	0	1,0	1,0	1,0	0,3	A	A		A	A	A		
AL10					C	16	21	0	0	1	0	1,0	1,0	0,5	0,3	A	B	D	A	B	B		A
AL11					C	16	20	0	0	0	0	1,0	1,0	0,7	0,3	A	B	B	A	B	B	C	A
AL12					C	16	17	1	0	0	0	1,0	1,0	1,0	0,3	A	A	C	B	A	B	C	
AL13					C	30	8	0	0	1	0	1,0	1,0	1,0	0,5	E	B	D	A	A	B		
AL14	A	B	B	B																			
AL15	A	A	A	A																			

Description of the table. The rows of the table represent the participants in the experiment, note that two students participated only in the training session, and five students of the CG partipated only in the experimental session. The columns have the following meanning:

- USAB1: are the answers to the questionnaire about the first impression. (four questions)
- GRP: identifies the group that a student belongs to.
- TIME: is the time expended in studying the algorithm (STUD) and completing the knowledge test (TEST).
- TEST: are the answers to the knowledge test.
    - P1 represents the first question, and is divided in four columns, each one for the main ideas to be recognized: (P11) operations with lists, (P12) recursion with subtrees (P13) left-to-right direction in tree traversing, (P14) accumulation of recursive operations with subtrees.
    - Rest of columns are the rest of the questions.
- USAB2: are the answers to the questionnaire completed at the end of the experiemntal session.

Appendix G. Normality Data Test

One-Sample Kolmogorov-Smirnov Test

		TPOAPREND	TPOTEST	P11	P12
N		13	13	13	13
Normal Parameters[a,b]	Mean	32,3077	17,3846	,6154	,3846
	Std. Deviation	16,84431	5,70874	,50637	,50637
Most Extreme Differences	Absolute	,264	,165	,392	,392
	Positive	,264	,104	,272	,392
	Negative	-,238	-,165	-,392	-,272
Kolmogorov-Smirnov Z		,951	,597	1,412	1,412
Asymp. Sig. (2-tailed)		,326	,869	,037	,037

One-Sample Kolmogorov-Smirnov Test

		P13	P14	P2	P3
N		13	13	13	13
Normal Parameters[a,b]	Mean	,5385	,3077	16,0000	14,3846
	Std. Deviation	,51887	,48038	,00000[c]	2,21880
Most Extreme Differences	Absolute	,352	,431		,532
	Positive	,312	,431		,391
	Negative	-,352	-,261		-,532
Kolmogorov-Smirnov Z		1,268	1,555		1,919
Asymp. Sig. (2-tailed)		,080	,016		,001

One-Sample Kolmogorov-Smirnov Test

		P4	P5
N		13	13
Normal Parameters[a,b]	Mean	1,2846	,5308
	Std. Deviation	1,74492	,33011
Most Extreme Differences	Absolute	,488	,373
	Positive	,488	,373
	Negative	-,250	-,242
Kolmogorov-Smirnov Z		1,759	1,345
Asymp. Sig. (2-tailed)		,004	,054

a. Test distribution is Normal.

b. Calculated from data.

c. The distribution has no variance for this variable. One-Sample Kolmogorov-Smirnov Test cannot be performed.

TPOAPREND represents the time expended in studing the algorithm. TPOTEST represents the time employed by students in completing the knowledge test. P11-P14 represent the four ideas to be identified by students in the first question of the knledge test. P2-P5 represent the grades of the second, third, fourth and fifth questions of the knowledge test.

Appendix H. Statistical analysis of learning data

Descriptive Statistics

	N	Mean	Std. Deviation	Minimum	Maximum
P11	13	,6154	,50637	,00	1,00
P12	13	,3846	,50637	,00	1,00
P13	13	,5385	,51887	,00	1,00
P14	13	,3077	,48038	,00	1,00
P2	13	16,0000	,00000	16,00	16,00
P3	13	14,3846	2,21880	7,00	15,00
P4	13	1,2846	1,74492	,00	7,00
P5	13	,5308	,33011	,30	1,00
GRUPONUM	13	,46	,519	0	1

Mann-Whitney Test

Ranks

	GRUPONUM	N	Mean Rank	Sum of Ranks
P11	0	7	5,79	40,50
	1	6	8,42	50,50
	Total	13		
P12	0	7	5,43	38,00
	1	6	8,83	53,00
	Total	13		
P13	0	7	7,21	50,50
	1	6	6,75	40,50
	Total	13		
P14	0	7	5,00	35,00
	1	6	9,33	56,00
	Total	13		
P2	0	7	7,00	49,00
	1	6	7,00	42,00
	Total	13		
P3	0	7	6,57	46,00
	1	6	7,50	45,00
	Total	13		
P4	0	7	5,93	41,50
	1	6	8,25	49,50
	Total	13		
P5	0	7	5,14	36,00
	1	6	9,17	55,00
	Total	13		

GRUPONUM is the numerical representation of the groups: 0 = CG, and 1 = EG.

Test Statistics[b]

	P11	P12	P13	P14	P2	P3
Mann-Whitney U	12,500	10,000	19,500	7,000	21,000	18,000
Wilcoxon W	40,500	38,000	40,500	35,000	49,000	46,000
Z	-1,437	-1,859	-,247	-2,494	,000	-,926
Asymp. Sig. (2-tailed)	,151	,063	,805	,013	1,000	,355
Exact Sig. [2*(1-tailed Sig.)]	,234[a]	,138[a]	,836[a]	,051[a]	1,000[a]	,731[a]

Test Statistics[b]

	P4	P5
Mann-Whitney U	13,500	8,000
Wilcoxon W	41,500	36,000
Z	-1,224	-2,156
Asymp. Sig. (2-tailed)	,221	,031
Exact Sig. [2*(1-tailed Sig.)]	,295[a]	,073[a]

a. Not corrected for ties.

b. Grouping Variable: GRUPONUM

The normality test of P13 and P5 has resulted in values close to 0.05, therefore we have performed a significant test supoussing normality for this values. Result of the following analysis (where GRUPO represents the two groups EG=E and GC=C) are the same to the previous one: P13 has not significicant differences, and P5 has them.

	GRUPO	N	Mean	Std. Deviation	Std. Error Mean
P13	E	6	,5000	,54772	,22361
	C	7	,5714	,53452	,20203
P5	E	6	,7667	,36148	,14757
	C	7	,3286	,07559	,02857

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
P13	Equal variances assumed	,106	,751	-,238	11	,817	-,07143	,30074	-,73336	,59050
	Equal variances not assumed			-,237	10,606	,817	-,07143	,30136	-,73773	,59488
P5	Equal variances assumed	27,107	,000	3,150	11	,009	,43810	,13910	,13194	,74425
	Equal variances not assumed			2,915	5,376	,030	,43810	,15031	,05968	,81651

Appendix I. Statistical analysis of time data

T-Test

Group Statistics

	GRUPO	N	Mean	Std. Deviation	Std. Error Mean
TPOAPREND	E	6	49,0000	4,97996	2,03306
	C	7	18,0000	5,41603	2,04707
TPOTEST	E	6	19,6667	5,12510	2,09231
	C	7	15,4286	5,79819	2,19151

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
TPOAPREND	Equal variances assumed	,002	,964	10,670	11	,000	31,00000	2,90544	24,60518	37,39482
	Equal variances not assumed			10,745	10,922	,000	31,00000	2,88510	24,64441	37,35559
TPOTEST	Equal variances assumed	,262	,619	1,384	11	,194	4,23810	3,06129	-2,49975	10,97594
	Equal variances not assumed			1,399	10,978	,190	4,23810	3,02993	-2,43238	10,90857