Abstract

KAHLER, SUSAN ELIZABETH. A Comparison of Knowledge Acquisition Methods for the Elicitation of Procedural Mental Models. (Under the direction of Sharolyn A. Converse-Lane.)

This study compared the effectiveness of two different knowledge acquisition methods in terms of their ability to measure the structure of a student's procedural mental model of computer programming as it evolved in a classroom setting over the course of a semester. Ratings and backward thinking tasks were used to estimate the structure of students' procedural mental models at several points throughout an introductory programming course. For both methods, comparisons between the student's procedural mental model and a prototype model (i.e., the instructor's conceptual model of the knowledge provided in the course) were used to identify how the procedural tasks and the relationships among tasks are represented in the student's mental model. The values from the ratings task were input into the Pathfinder scaling algorithm, which compares the structural similarity between a prototype model and the student's mental model, to generate an index of similarity. Additionally, the Pathfinder scaling algorithm measured the consistency of the student's rating process via the coherence index. The steps recorded during the backward thinking task were compared to the prototype model to identify the number of steps out of sequence. The first hypothesis was that there would be a positive correlation between the index of similarity and the project score. However, this hypothesis was only partially supported, as the correlation between the index of similarity and project score was significant only at test time three. The second hypothesis was that as the student completed each successive project, the index of similarity would increase over the

course of the semester. This hypothesis was not supported because the index of similarity never increased beyond the value obtained at test time one. The third hypothesis was that as the student completed each successive project, the coherence index would increase. This hypothesis was not supported, as no main effect of test time on the coherence index was found. The fourth hypothesis was that there would be a negative correlation between the project score and the number of steps out of sequence. The number of steps out of sequence was excluded from further analysis due to low inter-rater reliability. Therefore this hypothesis cannot be evaluated. Lastly, it was hypothesized that the index of similarity would correlate more positively with student's performance than the number of steps out of sequence. Again, without the number of steps out of sequence, this hypothesis could not be fully evaluated. However, for project three, the index of similarity was significantly positively correlated with the project score.

A COMPARISON OF KNOWLEDGE ACQUISITION METHODS FOR THE ELICITATION OF PROCEDURAL MENTAL MODELS

By

SUSAN ELIZABETH KAHLER

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in fulfillment of the requirements for the Doctorate of Philosophy

PSYCHOLOGY DEPARTMENT

Raleigh

2002

APPROVED BY:

____________________________ ____________________________

____________________________ ____________________________

Chair of Advisory Committee

BIOGRAPHY

Susan Kahler lives in Raleigh, North Carolina with her husband Kevin. Their family currently consists of two dogs, a cat, and a love bird. Prior to entering North Carolina State University in 1989, Susan received her B.A. and M.S. in Engineering Psychology from the University of Illinois at Champaign-Urbana. In 1992, she received her MIE in Industrial Engineering from North Carolina State University, with a concentration in Ergonomics and Human Factors. Susan has been employed by IBM in Research Triangle Park, NC as a Human Factors Engineer since 1992. She currently works in IBM's Global Services division.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

Learning: The Development of Mental models

A mental model is the student's cognitive representation of a task or system, and is developed through experience (Wilson and Rutherford, 1989). A mental model can allow humans to describe a system's purpose and form, explain system function and states, and predict future system states (Rouse and Morris, 1986). The development of a student's mental model is influenced by the structure of the knowledge (i.e., conceptual model, Young, 1983) that is presented to the student during interaction with the learning environment.

Mental models contain multiple types of knowledge. In the literature, these types of knowledge have been referred to as declarative or conceptual (Anderson, 1983), procedural or task, and strategic (Converse and Kahler, 1992; Kraiger, Ford, and Salas, 1991; Shadbolt and Wielinga, 1990). Declarative knowledge contains information about concepts and their defining features, and the relationships among those features and concepts (Shadbolt and Wielinga, 1990). From a systems perspective, examples include information about a system's components, functions, behaviors, states and layout (Kieras, 1988). Declarative knowledge has been characterized as being (1) a network of interconnected concepts (Anderson, 1983; Norman and Rumelhart, 1975), (2) static (Gordon, 1992), and (3) easy to verbalize (Gordon, 1992).

Procedural knowledge contains information about how goals and tasks should be performed (Shadbolt and Wielinga, 1990), including the order in which the actions should be performed (Modrick, 1991). Procedural knowledge is often represented in the form of IF-THEN production rules, with the IF side of the rule representing conditions of a situation and the THEN side of the rule representing the corresponding actions to be performed (Anderson, 1983).

Strategic knowledge is the combination of procedural and declarative knowledge that has become compiled, tacit, and automatic due to the continued rehearsal of routine activities (Converse and Kahler, 1992). Gordon (1992) summarizes the relationship between declarative and procedural knowledge as the direct association between the system conditions and system responses, which also encompasses the steps required to obtain a system response. Strategic knowledge is also the knowledge that enables experts to monitor and control problem solving (Shadbolt and Wielinga, 1990). In situations where information is incomplete or inappropriate, strategic knowledge guides the identification of alternative solutions to the problem (Shadbolt and Wielinga, 1990). However, strategic knowledge may be so well learned that it is automatic and not available for verbal report (Bainbridge, 1979; Anderson, 1985).

Knowledge Acquisition

Knowledge acquisition is the process of eliciting knowledge from a human and transferring this knowledge to either an individual or a computer system (Gordon, 1992). Although knowledge acquisition is often performed using experts, knowledge acquisition can also be performed using novices or students when in the context of a learning environment. Knowledge acquisition methods can be incorporated into studies to elicit, structure, and represent a student's mental model (Gammock and Young, 1985).

There are several knowledge acquisition methods from which to choose. According to the differential access hypothesis, each method has a degree of suitability for the type of knowledge to be acquired (Cooke, 1994; Converse and Kahler, 1992; Hoffman, 1992). In her knowledge acquisition taxonomy, Cooke (1994) incorporated a matrix of knowledge acquisition methods mapped to the type of knowledge to be elicited. For example, card sorting and scaling techniques were hypothesized to be methods appropriate for eliciting the conceptual structure of a domain (i.e., declarative knowledge) while thinking aloud during task performance (i.e., verbal protocol) was hypothesized to be an appropriate method for eliciting procedural knowledge.

Cooke (1994) emphasized the need to conduct empirical studies that compare knowledge acquisition methods in terms of their ability to elicit a particular type of knowledge. As an example of such a study, Wilt (1998) examined the validity of verbal protocols and laddered grids (a form of structured interviewing which focuses on the relationship between concepts) in terms of their ability to capture declarative and procedural knowledge across several domains. Results were that laddered grids were more effective at eliciting declarative knowledge than verbal protocols. However, there was no difference between the two methods in terms of their ability to elicit procedural knowledge. Regretfully, this comparison study is a rarity in the domain.

<u>Structural knowledge assessment</u>. One commonly used method of knowledge

acquisition is structural knowledge assessment (Goldsmith and Kraiger, 1997). During structural assessment, the participant is asked either to rate concepts according to their similarity or to sort the concepts into related groups. Scaling algorithms are used to calculate the spatial coordinates of concepts. The spatial coordinates are represented in graphical form to show the strength of the relationships between the concepts, with the degree of spatial proximity indicating the degree of positive or negative relationship between two concepts or features. Positive relationships are depicted with close spatial proximity. Scaling methods can be applied to group data as well as individual data, thus allowing comparisons at the group level (Cooke and McDonald, 1986). The advantage of using a scaling algorithm is that it has a high reliability; multiple researchers who use the same rating or distance measures and the same algorithm will produce nearly identical representations (Cooke and McDonald, 1986).

Structural knowledge assessment is commonly used to acquire declarative mental models (Converse and Kahler, 1992; Kahler, Converse, Lester, Cheatham, and Stelling, 2000) but has typically not been used to acquire procedural mental models. However, there is no empirical evidence to suggest that structural knowledge assessment cannot be used to acquire procedural mental models. In fact, researchers (Goldsmith and Kraiger, 1997; Acton, 1991) have identified the need to evaluate structural knowledge assessment in terms of its ability to represent procedural knowledge.

Pathfinder scaling algorithm. One example of a scaling algorithm is the Pathfinder Associative Network. The output of the Pathfinder algorithm is a network of concepts and their links. When the links are labeled, the Pathfinder networks can then be referred to as semantic nets (Cooke and McDonald, 1986). An example of a Pathfinder network is shown in Figure 1. Concepts that are closely related are located in close proximity to each other, either directly linked or superimposed together. For example, cottonwood, plant, and leaves are closely related because they overlap one another. Additionally, flower, green, and tree are also closely related because they are linked directly to plant. However, green is more closely related to plant than to either flower or tree because it has the shortest link of the three concepts connected to plant.

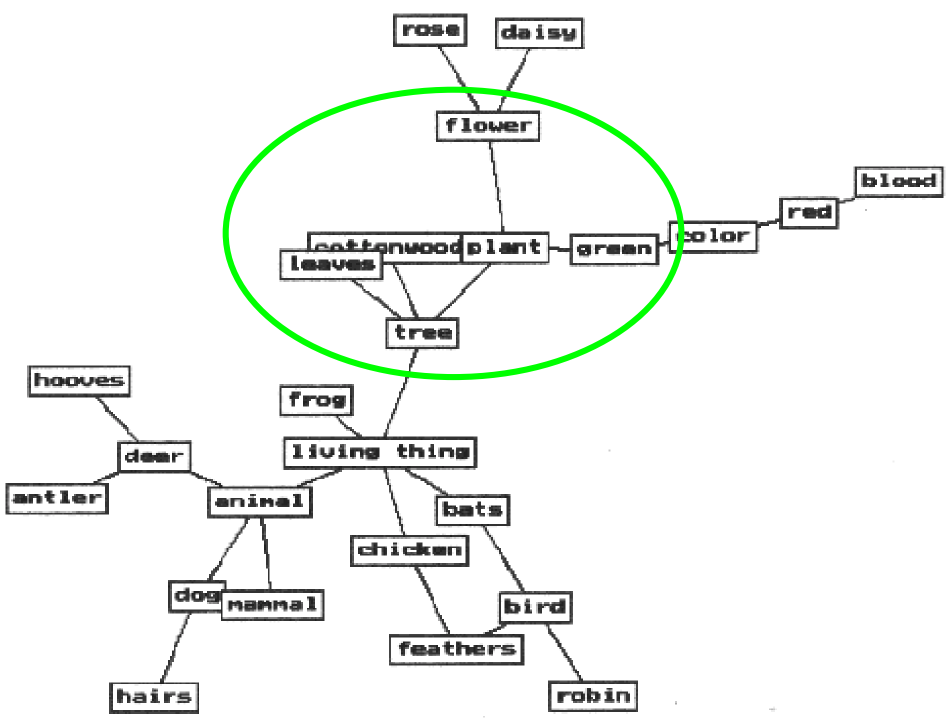

Figure 1. Example of a Pathfinder Network

Pathfinder networks have been used to identify the differences in mental models between expert and novice computer programmers (Cooke and Schvaneveldt, 1986) and between expert and novice users of commands for specifying the formatting of a document (Kellogg and Breen, 1990). Cooke and Schvaneveldt (1986) found that the networks revealed critical associations between concepts understood by the expert programmers, and incorrect understandings of associations of concepts among naive programmers.

The Kellogg and Breen (1990) study used Pathfinder to evaluate the performance of participants who either had experience or did not have experience with document formatting commands. Participants were asked to sort the concept cards into piles of related concept cards. Each group's network was compared to a prototype network, which represented the ideal conceptual model of the commands for specifying the layout of a document. To perform this comparison, Pathfinder generated an index of similarity. The index of similarity is the degree of agreement or correlation between group A's classification of the relationships between relevant concepts and group B's classification of those same concepts, or between and expert's and novice's classification of the concepts. By comparing the novice and the expert groups' index of similarity to the ideal index of similarity, which was created by the system designer, Kellogg and Breen (1990) found that the expert users' network more closely matched the prototype network than did the novices' network.

While the index of similarity provides the metric to compare two networks, the

Coherence Index measures the consistency of a set of similarity ratings (Goldsmith and Kraiger, 1997). Coherence provides information on the consistency of a student's mental model by reflecting how consistent the student was when assigning similarity ratings to pairs of concepts. The underlying assumption of coherence is based on the triangle inequality law. For example, if a student rates concepts A and B as highly related, and B and C as highly related, one would expect that the student would rate concepts A and C as highly related. This same logic can be applied to an entire set of concepts. Pathfinder measures coherence by correlating the student ratings to a set of expected ratings. The expected ratings for each pair of concepts (e.g., concepts A and B) are derived from the comparison of the linked concepts surrounding A and B. For example, Pathfinder would compare the linked concepts surrounding A and the linked concepts surrounding B to calculate an expected rating for concepts A and B. The higher the correlation between the student and expected ratings, the higher the coherence of the student's network. Therefore, the coherence index is the degree of agreement or correlation between the student’s ratings and the expected ratings calculated by Pathfinder (automated by PCKNOT 4.3).

The coherence index has been used to investigate how students’ mental models developed over a period of two semesters in courses in introductory methods and statistics, and computer programming (Acton, 1991). Acton (1991) found that as students gained more experience in the domain, their coherence indices increased and that the coherence indices were significantly related to grades. These findings indicate that the coherence index is a valid measure of mental models in a procedural domain such as computer programming.

Acquisition of Procedural Mental Models

There are three general categories of knowledge acquisition methods that have been used to acquire procedural mental models: structured interviews and questionnaires, verbal protocols, and observation (Converse and Kahler, 1992). During interviews and questionnaires, the participant is asked to respond to a series of questions about the steps required to perform a task. Verbal protocols are transcripts of the participant's verbalization of his or her thoughts either during task performance (concurrent) or shortly after task performance (retrospective). During observation, the participant performs the task (without talking about the thought process either during or after task performance) in the presence of observers who record the participant's task steps. These non-statistical methods require subjective analysis of the data by the researcher and inter-rater reliability must be established (Converse and Kahler, 1992).

One type of structured interviewing that is hypothesized to acquire procedural knowledge is backward thinking (Converse and Kahler, 1992). During a backward thinking interview, the student is asked to work backwards from the goal of the problem to identify the sequence of steps required to arrive at that goal. The student starts by specifying the step prior to reaching the goal and continues to decompose the current step into the steps preceding it. The output of backward thinking is a series of if-then rules. Backward thinking narrows the possible paths that can be pursued because the goal state has been specified. Each step in the path is present only because it was derived from a higher level or more recent step. Andrus (1988) suggests that backward thinking makes for a more efficient method than forward thinking (e.g., verbal protocols), as the opportunities for branching onto new paths are limited due to the constraints imposed by existing steps in the path.

Evaluations of Procedural Knowledge Acquisition Methods

Table 1 provides a summary of empirical studies that have investigated the predictive validity of procedural knowledge acquisition methods. Only one study used backward thinking (Andrus, 1988). The remaining studies had participants perform a ratings task, which in some cases, the ratings served as the input into the Pathfinder algorithm. The studies in Table 1 are discussed in further detail.

In Barker and van Schaik (1999), an unknown number of college students participated in KA tasks designed to elicit their mental models of a word processing software package. Each student completed all of the KA tasks in the order listed: rating, sorting, laddering, concept mapping, and teachback interviewing. Task performance was evaluated by having students complete basic word processing tasks, which were scored by a panel of judges. However, correlations were not calculated between all KA and performance task measures. Therefore, no specific conclusions or recommendations can be drawn from this study regarding the predictive validity of KA methods.

In the first study reported in Cooke, Neville, and Rowe (1996), four participants interacted with a NASA mission control simulator to identify and diagnose faults by selecting their choices via menu driven software. These menu selections were translated into action sequences, which were then input into the Pathfinder algorithm to generate the participant's Pathfinder network. Over the course of the study, participants' indices of similarity increased, when compared against a gold standard. However, in this study, the number of tasks and their difficulty were not reported.

The second study reported in Cooke, Neville, and Rowe (1996) was an analysis of the verbal protocols produced by thirty two Air Force technicians who were engaged in avionics troubleshooting. As in study one, troubleshooting steps were translated into action sequences which were then input into the Pathfinder algorithm to generate the participant's Pathfinder network, which was then compared to a gold standard network. The main finding of the second study was that the correlation between the index of similarity and the score assigned to task performance was significant for one of the problems. However, the archival data analysis was incomplete, as participants completed a total of ten problems, but only two of those problems were examined during the analysis. As with study one, task difficulty was not reported.

In Rowe, Cooke, Hall, and Halgren (1996), nineteen Air Force technicians performed a variety of KA tasks in the troubleshooting domain: laddering interview, concept relatedness ratings, diagramming, and verbal protocols. Participants performed the KA tasks in succession. A panel of judges scored the verbal protocol to determine a participant's task performance. From the results, the researchers concluded that ratings and laddering were predictive of task performance while verbal protocols were not.

Vortac, Edwards, and Manning (1994) had ten air traffic controllers simulate air traffic control situations using a high and a low complexity scenario, in that complexity was defined by the number of departures, arrivals, etc. Action sequences, as recorded by observers, were input into the Pathfinder algorithm to generate Pathfinder networks which were visually inspected to identify common links. The researchers concluded that the Pathfinder networks were similar across task difficulty. The Pathfinder index of similarity was not used to evaluate the similarity of the networks generated by participants for the low and high complexity tasks.

In the last study, Andrus (1988), six media selection experts worked through four media selection problems using one KA method per problem: backward thinking, constrained processing (the participant is given a finite time in which to solve a task problem), verbal protocols, and behavioral event interviews (similar to critical incidents). The transcripts for each of the KA methods were analyzed to identify the number of media selection rules generated. It was reported that backward thinking generated two additional rules that were not identified by the other KA methods.

As discussed above, several studies have attempted to elicit procedural knowledge from participants using a variety of knowledge acquisition methods (Cooke, Neville, and Rowe, 1996; Rowe, Cooke, Hall, and Halgren, 1996; Vortac, Edwards, and Manning, 1994). What is problematic about these studies is that no true experiments were conducted. Instead, these studies employed quasi-experimental designs in which no comparison groups were utilized. However, this problem is not unique to these studies. It is the general state of the domain, where experimental evaluations of knowledge acquisition methods are sparse (Shadbolt, O'Hara, and Crow, 1999). Additionally, the number of participants ($N<20$) used in four of the six studies was quite small. Shadbolt, O'Hara, and Crow (1999) suggest that a sample size of larger than twenty participants is needed to achieve statistical significance. One study (Barker and van Schaik, 1999) does not even report the number of participants; this does not allow the reader to determine the study's statistical power or to evaluate the generalizability of the study results.

In some studies using the Pathfinder scaling algorithm (Cooke, Neville, and Rowe, 1996; Rowe, Cooke, Hall, and Halgren, 1996; Vortac, Edwards, and Manning, 1994), the data input into the algorithm were behavioral rather than cognitive indices such as similarity or distance values. Instead, action sequences generated during task performance were input into the scaling algorithm. Action sequence, when collected during task performance, is not a representative knowledge acquisition measure but rather is an indicator of performance. The problem with using performance measures is that they do not reflect the impact that instruction has on a student's knowledge base because they do not necessarily assess the content and structure of a student's mental model. It may be that recently obtained knowledge doesn't immediately improve task performance. A better method is to compare the structure of the student's mental model with the structure of the conceptual model, knowledge that the student did or did not learn can be identified, and that can assist in developing follow-on training sessions. By evaluating mental model structure as well as performance, initial learning as well as principles that were not learned may be detected more rapidly, and the relationship between model structure and performance can be more fully explored. Therefore, more experimental studies need to be conducted that include both knowledge acquisition and performance measures.

In studies whereby KA methods were used instead of action sequences (Barker and van Schaik (1999); Rowe, Cooke, Hall, and Halgren (1996)) and whereby a separate task performance measure was generated, the KA tasks were completed prior to completing the performance task. Johnson (1995) speculated that perhaps performing the KA task prior to the performance task provides some type of advantage in structuring the problem. This type of speculation points to the need to incorporate task order as an independent variable when conducting KA studies, in that comparisons should be made between participants who perform the KA task before and after the performance task.

To date, the majority of studies that have evaluated procedural knowledge acquisition methods and their relationship to performance measures have used a cross-sectional methodology whereby the participants are involved in a single data collection session over a relatively short period of time. Alternately, studies that employ a longitudinal methodology span a long duration and utilize repeated data collection sessions. A longitudinal study should be most sensitive to changes in the structure of a mental model that indicates that learning has occurred and the relationship between changes in mental model structure and performance characteristics. This evaluation will be more likely to establish the reliability of knowledge acquisition techniques than will cross-sectional studies. Unfortunately, longitudinal studies are almost non-existent in the knowledge acquisition domain. Of the studies cited in Table 1, only one study was conducted over multiple data collection sessions (Cooke, Neville, and Rowe, 1996) but the duration of that study was only three weeks. Although Shadbolt, O'Hara, and Crow, (1999) suggest methods for improving the experimental design of studies that evaluate knowledge acquisition methods, they omit any reference to the need for longitudinal studies. However, there is a need in the knowledge acquisition domain to conduct longitudinal studies to examine the structural changes in mental models over time so that the relative effectiveness of conducting such a study can be assessed and communicated to other researchers.

Overview of Study

This study compared the effectiveness of two different knowledge acquisition methods in terms of their ability to measure the structure of a student's procedural mental model of the task of computer programming and to compare the changes in model structure to potential changes in performance on a programming task. The measures of mental model structure were repeated over the course of a semester in which students were enrolled in a class that teaches this skill. Ratings and backward thinking tasks were used to measure students' procedural mental models throughout the introductory programming course. For both methods, comparisons between the student's procedural mental model and a prototype model were used to identify how the procedural tasks and the relationships among tasks were represented in the student's mental model, and how the knowledge of these relationships grew and changed over the course of a semester, and how closely the student's mental model matched the prototype model at various points in the semester. Additionally, this study investigated the relationship between performance and knowledge acquisition measures at various points in the semester to establish the validity of the two knowledge acquisition methods employed in this study for identifying and capturing procedural knowledge.

Hypotheses

H1: When a student's project score is high, the index of similarity (generated from the ratings task) will be high as well, thus implying more similarity with the instructor's conceptual model of the course. This hypothesis will be supported if the correlation between the project score and the index of similarity is significantly higher for students who performed well on the project than for students who performed poorly on the project. This hypothesis was adopted because researchers have found that the similarity between students' and experts' mental models can serve as a predictor of course performance (Goldsmith and Kraiger, 1997, Acton, 1991). Additionally, this appears to be the case for procedural domains, such as computer programming, as well as declarative domains (Acton, 1991).

H2: As the student completes each successive project, the similarity index (generated from the ratings task) will increase, thus implying that the student's mental model is evolving to more closely match the instructor's conceptual model of the course. This hypothesis will be supported if the similarity index for Project 3 is significantly higher than the similarity index for Project 2, which is significantly higher than the similarity index for Project 1. Due to the cross-sectional nature of studies that have evaluated procedural knowledge acquisition methods, this hypothesis has not been investigated. A longitudinal study provides the opportunity to do so. Action (1991) found that a related metric, the index of coherence, did increase over the course of his longitudinal study.

H3: As the student completes each successive project, the coherence index (generated from the ratings task) will increase, thus implying a more internally consistent mental model. This hypothesis will be supported if the coherence index for project 3 is significantly higher than the coherence index for project 2, which is significantly higher than the coherence index for project 1. Acton (1991) found that students' index of coherence did increase over the period of two semesters. Again, this finding held for procedural as well as declarative domains.

H4: When a student's project score is high, the number of steps out of sequence (generated from the backward thinking task) will be low, thus implying more congruence with the instructor's conceptual model of the course. This hypothesis will be supported if the negative correlation between the project score and the number of steps out of sequence is significantly higher for students who performed well on the project than for students who performed poorly on the project. There are no studies on backward thinking to support or refute this hypothesis. However, if the backward thinking measure has predictive validity, this relationship should be found.

H5: Although no research has been found to support this hypothesis, it is reasonable to assume that the index of similarity (generated from the ratings task) will correlate more positively with a student's project score than the number of steps out of sequence (generated from the backward thinking task). The index of similarity is a metric that should be better at indicating the accuracy of the structure of a student's mental model than the number of steps out of sequence. The predictive validity of the index of similarity generated by Pathfinder has been empirically established as a valid knowledge acquisition measure for representing the structure of a declarative mental model but not for a procedural mental model. However, no research has established the predictive validity of the number of steps out of sequence for representing the structure of any type of knowledge. This hypothesis will be supported if the correlation between the project score and the index of similarity is significantly higher than the correlation between the project score and the number of steps out of sequence.

Table 1

cedural Knowledge Acquisition Methods: A Review of Empirical Studies

Authors	KA Methods Evaluated	Methodology	Results and Conclusions	Critique
Barker and van Schaik (1999)	--rating --sorting --laddering --concept mapping --teachback interviews	--Number of S's not reported --S's completed tasks related to knowledge acquisition, then S's completed a performance task --DV's included: 1. ratings of task performance 2. number of concepts used during task performance	--Correlation between task performance and the number of concepts used during task performance was significant	--Not all of the results for the different knowledge acquisition methods were reported --No baseline measures of S's pre-existing knowledge --No gold standard for comparison --Having S's serve as within subjects may have affected the problem-solving process

Table 1 Cont.

Authors	**KA Methods Evaluated**	**Methodology**	**Results and Conclusions**	**Critique**
Study 1: Cooke, Neville, and Rowe (1996)	--action sequences	--4 S's over 3 weekly 4 hour sessions -- S's interacted with a NASA mission control simulation to identify and diagnose faults -- S's chose the faults and their corrective actions from a pre-defined menu of choices -- DV's included: 1. the Pathfinder Index of Similarity 2. % correct fault identification 3. % correct fault diagnosis 4. % correct actions taken	-- Over sessions, S's index of similarity increased and more closely approximated the ideal network (gold standard) --The more closely S's procedural networks approximated the ideal network, the better the performance as the index of similarity correlated significantly with the fault identification, diagnosis and corrective action DV's	--Small sample --Number of problems and their difficulty not reported

Table 1 Cont.

Authors	KA Methods Evaluated	Methodology	Results and Conclusions	Critique
Study 2: Cooke, Neville, and Rowe (1996)	--action sequences	--archival data of verbal protocols for pre/post test design --32 Air Force technicians in the Avionics troubleshooting domain --1 pretest and 1 post-test problem out of 10 problems available for analysis --DV's included: 1. the Pathfinder Index of Similarity 2. score assigned to task performance	--Networks of high and low performers have distinct characteristics --Correlation between the index of similarity and the worksheet score was significant for pretest problem	--Problems used for data analysis were not described in terms of task difficulty

Table 1 Cont.

Authors	KA Methods Evaluated	Methodology	Results and Conclusions	Critique
Rowe, Cooke, Hall, and Halgren (1996)	--laddering interview --concept relatedness ratings --diagramming --think aloud and verbal troubleshooting	--19 Air Force technicians in F-15 avionics troubleshooting --Technicians went through all of the KA methods --DV's included: 1. For laddering, the ratio of shared to different items was calculated 2. Ratings were submitted to Pathfinder and the index of similarity was calculated 3.For diagramming, the proportion of shared connections between two models was calculated 4. For think aloud, the transition probabilities on the action sequences were calculated 5. Experts scored task performance on 1 problem	--think aloud not predictive of performance (dissociation between task performance and protocol) --ratings and laddering were predictive of performance	--Having S's serve as within subjects may have affected the problem-solving process --Need to incorporate problems with varying levels of problem difficulty

Table 1 Cont.

Authors	KA Methods Evaluated	Methodology	Results and Conclusions	Critique
Vortac, Edwards, and Manning (1994)	--action sequences	--10 air traffic controllers in an air traffic control environment using problems with 2 levels of task difficulty --DV's included: 1.Pathfinder networks aggregated across S's	--Pathfinder networks were similar across task difficulty	--Small sample --the Pathfinder Index of Similarity was not used to compare the 2 networks representing the 2 levels of task difficulty --Task difficulty did not vary over a continuum
Andrus (1988)	--backward thinking --constrained processing --protocol analysis --behavioral event interview	--6 media selection experts --DV's included: 1. number of rules generated	--Backward thinking generated an additional two rules	--Small sample

METHOD

Participants

Sixty-nine students who were enrolled in an introductory computer programming class at North Carolina State University initially participated in the study. Several students dropped out of the study because they dropped out of the course. Other students simply did not show up and therefore there was no communication as to why they dropped out of the study. By the end of the semester, only forty-four students (twenty-seven males, seventeen females) had completed all of the knowledge acquisition tasks; forty-two students had completed all of the experimental tasks including the post-test and the ease of use questionnaire.

For their participation in the study, students received two points extra credit which was applied to their course average for the semester. For example, if a student had an average of 89 in the class, the additional two points increased the average to 91. As an option, students could have completed an additional project for the same two points of extra credit if they chose not to participate in the study. Two students completed all but the post-test; they received one point extra credit.

The attrition rate for gender is shown in Table 2. The attrition rate over the course of the study, collapsed across all conditions, is shown in Table 3. Table 4 shows the attrition rate over the course of the study for each combination of knowledge acquisition task and task order.

Table 2

Attrition Rate by Gender

	Backward Thinking Task		Ratings Task	
	Before Project	**After Project**	**Before Project**	**After Project**
Male (Start of Study)	10	10	12	11
Male (End of Study)	7	7	6	6
Attrition Rate	3 (30.0 %)	3 (30.0 %)	6 (50.0 %)	5 (45.5 %)
Female (Start of Study)	7	7	5	7
Female (End of Study)	4	2	4	6
Attrition Rate	3 (42.9 %)	5 (71.4 %)	1 (20.0%)	1 (14.3 %)

Table 3

Attrition Rate Across Conditions Over the Course of the Study

Tasks in the Study	Number of Participants	Cumulative Attrition Rate
Complete Pre-test	69	
Complete KA session 1	51	18 (26.1 %)
Complete KA session 2	46	23 (33.3 %)
Complete KA session 3	44	25 (36.2 %)
Complete Post-test	42	27 (39.1 %)

Experimental Design

The experimental design was a 2x2x2x3 mixed model MANOVA. Knowledge acquisition task (ratings or backward thinking), task order (before or after the project due date) and gender (male or female) were the between-subjects variables. In terms of the task order variable, the student performed the knowledge acquisition task after having learned the project-related concepts in the course lectures. However, there was a time delay between the lecture and the project due date. Therefore, students were asked to perform the knowledge acquisition task either before or after the project due date. Subjects were randomly assigned to combinations of knowledge acquisition task and task order. The within-subjects variable was testing time (project one, project two, project three).

Table 4

Attrition Rate for Each Combination of KA Task and Task Order Over the Course of the Study

Knowledge Acquisition Task						
			Backward Thinking		**Rating**	
			Number of Participants	Cumulative Attrition Rate	Number of Participants	Cumulative Attrition Rate
Task Order	**Before**	Complete Pre-test	17		17	
		Complete KA session 1	12	5 (29.4%)	13	4 (23.5%)
		Complete KA session 2	11	6 (35.3%)	12	5 (29.4%)
		Complete KA session 3	11	6 (35.3%)	11	6 (35.3%)
		Complete Post-test	**11**	6 (35.3%)	**10**	7 (41.2%)
	After	Complete Pre-test	17		18	
		Complete KA session 1	12	5 (29.4%)	14	4 (22.2%)
		Complete KA session 2	09	8 (47.1%)	14	4 (22.2%)
		Complete KA session 3	09	8 (47.1%)	13	5 (27.8%)
		Complete Post-test	**09**	8 (47.1%)	**12**	6 (33.3%)

The dependent variables, regardless of whether the participant performed the ratings task or the backward thinking task, were pre-test score, project score, post-test score, final course score, and the subjective estimates of ease of use. The dependent variables unique to the ratings task were the Pathfinder Index of Similarity (henceforth referred to as the index of similarity) and the Pathfinder Coherence Index (henceforth referred to as the coherence index). The dependent variables unique to the backward thinking task were the number of steps out of sequence, the number of steps missing, and the backward thinking score.

Apparatus

Data collection room for tasks requiring a computer. The ratings and backward thinking tasks were performed on a computer workstation, which had Windows 98 installed as the operating system. Three individual workstations were set-up in a small room (approximately 10' x 12'), as shown in Figure 2. Each workstation contained a computer, with processor speeds of 100, 200, or 300 Mhz. The processor speed did not make a visible difference as to how either the operating system or any of the software applications performed. The monitors were 17" on the diagonal and were set to a 1028 X 768 screen resolution. A two-button mouse was attached to the computer workstation to serve as the input device. The participant worked individually at a workstation that contained a chair and table for the computer workstation.

Data collection room for tasks not requiring a computer. The pre-test, post-test, and subjective ease of use measures were collected using paper and pencil instruments, and did not require access to computers. Students completed the paper and pencil instruments in a conference room capable of seating 12 people. Students completed the instruments in groups, ranging in size from three to ten students.

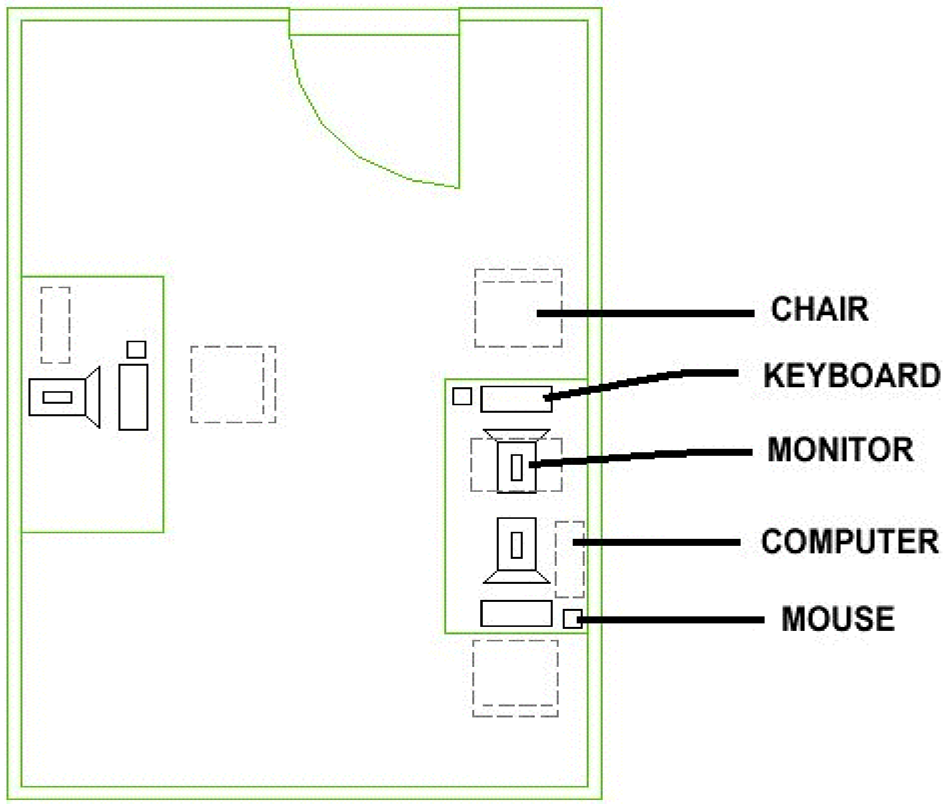

Figure 2. Data Collection Room for Tasks Requiring a Computer

Materials

Participants' computer programming experience was assessed in a paper and pencil demographics questionnaire. The demographics questionnaire is shown in Appendix A. In addition, participants were asked to provide their gender, college, department, and college level on the demographics questionnaire.

The pre-test was designed to measure students' C ++ computer programming competency before exposure to the introductory course in C ++ computer programming.

This test was also administered via paper and pencil. The exams from a previous year's introductory C ++ programming class were used to develop the pre-test, which was done in conjunction with the course instructor. The exams were designed to be taken in a three-hour test period. However, for this study, a pre-test, which could be taken in an hour or less, was created. Therefore, the questions on the pre-test were a randomly selected subset of the exam questions. The pre-test is shown in Appendix B. The pre-test and the post-test were identical in content. The order of the pre-test questions was randomized during the creation of the post-test.

Projects. There were three computer programming projects assigned by the course instructor during the semester. The first project was the video store project, which asked students to create a program that would calculate the cost of video rentals based on a discount scheme. The second project was the stock traders project, which asked students to create a program to plot stock price highs and lows. The third project was the poker project, which asked students to create a program for playing blackjack.

Ratings task. Materials for the ratings task were created by asking the course instructor to identify the programming procedures and the relationships amongst those procedures that he expected students to understand in order for a student to successfully complete a project. The programming procedures are shown in Table 5. For project one, students rated the relatedness of concepts one through nine. For project two, students rated the relatedness of concepts one-fifteen. For project three, students rated the relatedness of all twenty concepts.

During the ratings task, students were asked “How related are these two terms?”. The response scale was a five-point Likert scale using anchors of slightly related and highly related. A separate button was provided for the student to indicate that two concepts were not related. The pairwise comparisons were performed using an online ratings tool (Single Relatedness Software, Version 1.0) developed by Dr. Nancy Cooke in conjunction with Sandia Research Corporation.

A training instrument was created using four procedures (taking a quiz, taking an exam, preparing a report, and making a presentation) to familiarize the student with the online ratings tool. The software on which the ratings task was performed in shown in Figure 3.

An optimal or prototype Pathfinder network was created using the instructor's ratings to represent the optimal rating of relationships between the procedures. This prototype network was used in the data analysis only and was not presented to participants. Upon inspection of the coherence indices derived from the prototype network for each of the three projects, the values of the indices were too low for projects one and three to be considered stable, as shown in Table 6. Goldsmith (2001), who participated in the development of the coherence index, sets a .400 criterion level for the coherence index when the prototype model is generated from experts. Therefore, the course instructor was asked to re-rate the concepts for projects one and three to improve the consistency of the prototype model. The values of the coherence indices from the first and second rating sessions are shown in Table 6. The values input into the data analysis for projects one and three were the values generated during the second rating session, while the value used in the project two data analysis was from the first rating session.

Backward thinking task. The backward thinking task was performed using a simple text editor. During the backward thinking task, students were asked, "What are the steps you would go through to complete the project, starting with the most recent step first and working backwards to the initial step?". Two training backward thinking tasks were created to familiarize the student with the backward thinking task. The first task asked the student to describe a path to the data collection room, and the second task was to describe the procedure of picking up a package from the post office.

Table 5

Programming Procedures for Ratings Task

Number	Procedure	Project
1	program design	one
2	functional decomposition	one
3	calling functions	one
4	defining variables	one
5	getting input	one
6	generating input	one
7	aligning output	one
8	performing calculations	one
9	testing and debugging	one
10	working with bool expressions	two
11	using if and switch statements	two
12	using loops	two
13	using class libraries	two
14	designing geometric solutions	two
15	reading from a file	two
16	using arrays	three
17	sorting	three
18	searching	three
19	using structs	three
20	using strings	three

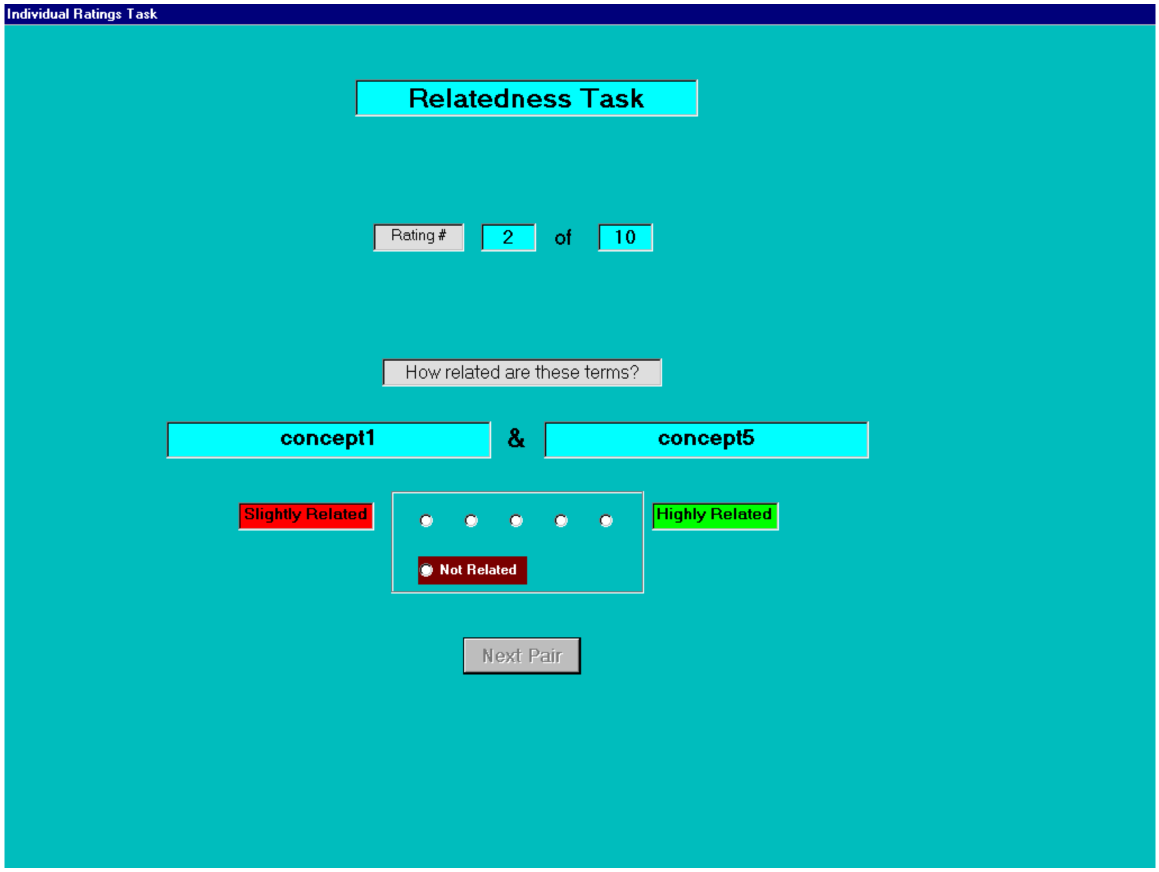

Figure 3. Online Ratings Task

Table 6

Coherence Indices for the Prototype Model

Project	First Rating Session	Second Rating Session
1	.145	.558
2	.478	N/A
3	.348	.506

Materials for an optimal or prototype model for the backward thinking task were created by asking the course instructor to identify the sequence of steps (from latest to earliest) necessary to complete each of the three course projects. These prototype models were used in the data analysis only and were not presented to participants.

Subjective measures. Participants' opinions on the ease of use of the knowledge acquisition task were assessed in a paper and pencil evaluation questionnaire. The ease of use questionnaire is shown in Appendix C. The ease of use questionnaire has five questions, with each response using a five-point Likert rating scale. The questionnaire covers five aspects of the knowledge acquisition task: clarity of the task instructions, ease of task performance, time spent during task performance, contribution of the task to understanding the course material, and integration of the task with the course.

Procedure

During the first week of the course (two class periods), the researcher introduced the study. Participants completed a written consent form prior to participating in the study. This consent form can be found in Appendix D. The consent form and demographics questionnaire were attached to the course syllabus; students could return both at the end of the first class period or they had the option of returning them at the beginning of the next class period. The consent form assured them of their anonymity and confidentiality in this study.

Participants completed the pre-test during the first two weeks of the course. Participants were asked to complete the pre-test in a conference room during scheduled times. Participants were given one hour to take the pre-test.

Participants were tested in groups of three to complete the KA tasks. Each participant worked at an individual workstation. Each testing session lasted approximately 30-45 minutes. The instructions read by the researchers are shown in Appendix E. The instructions guided the participants through the sequence of tasks they completed during all of the sessions.

For each of the three projects, participants completed either the ratings or backwards thinking task. At each of the three testing times throughout the semester, measures were collected for the project score for all participants. In addition, the index of similarity and the coherence index were calculated at each of the three testing times for those participants who performed the ratings task, while the number of steps out of sequence, the number of steps missing, and a backward thinking score were calculated at each of the three testing times for those participants who performed the backward thinking task. At each of the three testing sessions, students were given training to practice the knowledge acquisition task prior to completing the knowledge acquisition task for the programming project. Half completed the knowledge acquisition task before the programming project; the other half completed the knowledge acquisition task after the programming project had been completed, but prior to receiving the graded project results. Students were given a five-minute break for every half hour spent performing the knowledge acquisition task. The length of time between the testing sessions was dependent on the project due dates (approximately one month apart), with projects due in

September, October, and November.

Prior to the week of final exams, participants completed the post-test. Participants were asked to complete the post-test in the conference room during scheduled hours. Participants were given one hour to take the post-test. Upon completion of the post-test, participants completed the ease of use questionnaire to provide their opinions on the ease of use of the knowledge acquisition task. At the conclusion of the session, participants were given the opportunity to ask any questions they may have had about the study and were added to the list of students who were eligible for receiving extra credit for their participation in the study.

Preparation for Data Analysis

Pre- and post- tests. Participants' responses from the pre and post- tests were graded by the two course graders, who used a grading key provided by the course instructor. The researcher is not an subject matter expert in C++ computer programming, hence two course graders were requested to perform the grading tasks. Each grader scored every test.

Number of steps missing. The number of steps missing was converted to a percentage, as the number of steps for projects one and three was ten, while the number of steps for project two was eight.

Knowledge acquisition measures for the ratings task. The PCKNOT 4.3 software program was used to create the Pathfinder networks, and from those networks, calculate the indices of similarity and the coherence indices. This is a commercially available PC based program that can be purchased from Interlink, Inc.

The coherence index provides an indication of whether or not the student is evaluating the pairs of concepts using a consistent criteria for the comparison. The Pathfinder Algorithm calculates the coherence index by comparing indirect and direct measures for each pair of concepts. The direct measure for each pair of concepts is the actual rating entered in by the student. The indirect measure for each pair of concepts is the rating that the Pathfinder Algorithm predicts or expects for the pair of concepts based on how those concepts were rated against other concepts. To generate the coherence index, the indirect and direct measures are correlated with each other. The coherence index can range from zero to one. The higher the correlation, the more consistent the direct measure is with the indirect (or expected) measure, which implies that the student is consistent with the evaluation of the pairs of concepts.

The Pathfinder Algorithm can generate an index of the similarity between two networks. In this study, the two networks used for comparison were the participant's network and the prototype network, which was derived from the instructor's ratings of the concepts. Similarity of the networks is determined by the number of links that are common between the two networks, divided by the number of total links in the two networks minus the number of common links. For example, if network A has 25 links, network B has 26 links, and there are 14 links in common, the network similarity is calculated as 14/ (25+26-14) = .378. This is the proportion of the links in either network that are in both networks. If the networks are identical, similarity will equal one, if the networks have no links in common, then similarity will equal zero.

A high rating between two concepts implies that the concepts are more related while a low rating implies that the concepts are less related. The similarity ratings are automatically converted into a distance matrix by PC-KNOT. Distance is the relationship between the rating and its minimum and maximum values. It is calculated as X' = min + max –X, where X is the rating. Ratings that are slightly related are not close while ratings that are more closely related are closer. The distances of the concepts are symmetrical, as the distance between concepts i and j is no different than the distance between j and i, for all pairs of (i,j). Hence the links in the network have the same interpretation regardless of the direction in which they are read (i.e., reading from i to j is the same as reading from j to i).

Networks are created from the spatial coordinates, with nodes representing the concepts and the links among the nodes representing the type of relationships that exist between concepts. Links are assigned weights based on the strength of the relationship, as indicated by the entries in the distance matrix.

The complexity of the Pathfinder network can be varied according to the values of two parameters, r and q. The r is the Minkowski exponent, which specifies how the path length is computed and is a real number between one and infinity, inclusive. The q specifies the density of the network or the number of links in the path examined and is an integer between 2 and n-1 (n = number of concepts). As both values increase, the number of links in the diagram increases. According to the documentation accompanying the PCKNOT software, it is recommended that these parameters be set to r = infinity and q = n - 1, which will generate a network with the minimum number of links. A network created with specific values of parameters r and q is referred to as a Pfnet (r,q).

Knowledge acquisition measures for the backward thinking task. For each participant, a backward thinking score was calculated using a standardized protocol scoring key developed in conjunction with the course instructor. The backward thinking protocol scoring keys and guidelines are shown in Appendix F. The researcher and course instructor piloted the backward thinking protocol scoring key by using it themselves and making comparisons to each other's findings. The output of this comparison was the revision of the backward thinking protocol scoring sheets with the addition of guidelines prior to their use by the course graders. The additional guidelines are shown in the project scoring keys that are shown in Appendix F. The backward thinking protocol scoring key lists the items the protocol must contain and the order in which those items should be mentioned, with each item having a point value. The evaluation of the pre-test, post-test, and backward thinking protocols used two course graders, who were recommended by the course instructor because of their exceptional prior grading experience in the introductory C++ computer programming class.

For each project, the researcher trained the two course graders to score the backward thinking protocol using a sample of five randomly selected backward thinking protocols. These five protocols were the same ones graded by the researcher and course instructor. The graders were given an opportunity to ask questions and compare their grading strategies. Their inter-rater reliability was calculated for each project and is shown in Table 7. The inter-rater reliability is very high, with the exception of "number of steps out of sequence project 2" and "number of steps out of sequence project 3". For those two projects, graders varied mainly in their determination of when the participant performed the step of testing the program in the process. When to test a program was a critical step in the project, and performing the testing in the incorrect sequence resulted in the loss of seven to eight points, depending on the project. Due to the low inter-rater reliability, the number of steps out of sequence was excluded from further data analysis.

Table 7

Inter-rater Reliability

	pre-test	bwt score project1	steps out of sequence project 1	steps missing in project 1	bwt score project 2	steps out of sequence project 2	steps missing in project 2	bwt score project 3	steps out of sequence project 3	steps missing in project 3	post-test
Corr coef	.96	.99	.77	.98	.97	.41	.96	.99	.53	.99	.97
# in cell	44	20	20	20	20	20	20	20	20	20	41

Data Analysis

Backward thinking data. For the backward thinking data, the means for the backward thinking score and number of steps missing were first calculated for each combination of gender, testing time, and task order, and were then submitted to a mixed-model MANCOVA using pre-test score as the covariate.

Ratings data. The means for the index of similarity and the coherence index were first calculated for each combination of gender, testing time, and task order, and were then submitted to a mixed-model MANCOVA using pre-test score as the covariate.

Project scores. For all students, the means for the project score were calculated for each combination of gender, KA task, task order, and testing time, then submitted to a mixed-model ANCOVA using pre-test score as the covariate.

Competency test scores and subjective measures. For all students, the means for the pre and post programming competency test scores and the responses to the ease of use questions first were calculated for each combination of gender, KA task, and task order, then submitted to a mixed-model MANCOVA using pre-test score as the covariate.

RESULTS

Unless otherwise stated, all ANOVA results are reported with an alpha level of .05. Post hoc comparisons of significant results were conducted using Tukey's Studentized Range Test.

<u>Pathfinder Index of Similarity (Ratings Task)</u>

A significant main effect was found for test time for the index of similarity, as shown in Table 8. The means for test times one through three for the index of similarity were .287, .191, and .254 respectively. The index of similarity was significantly lower at test time two than at test times one or three. There were no significant differences between the indices of similarity when comparing test time one and test time three. However, it is interesting to see the trend that the index of similarity is at its highest at test time one.

A significant interaction effect was found for Gender x Task Order for the indices of similarity, as shown in Table 8. The means for Gender x Task Order for the indices of similarity are shown in Table 9. The post hoc test confirmed that females who completed the KA task prior to completing the project had a significantly higher index of similarity than females who completed the KA task after completing the project. For males, the post hoc test did not find a significant difference between those who had completed the KA task prior to completing the project and those who had completed the KA task after completing the project.

Table 8

ANOVA Results for Pathfinder Index of Similarity (Ratings Task)

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	.0009	.15	.6979
Task Order	1	.0016	.27	.6068
Test Time	2	.1077	8.96	.0004*
Task Order x Test Time	2	.0073	.61	.5454
Gender x Task Order	1	.0372	6.19	.0157*
Gender x Test Time	2	.0088	.73	.4850
Gender x Task Order x Test Time	2	.0279	2.33	.1063

Table 9

Means for Indices of Similarity: Gender x Task Order

	Before Project	**After Project**
Male	.222	.259
Female	.276	.219

Pathfinder Coherence Index (Ratings Task)

No significant main effects or interaction effects were found for the coherence index, as shown in Table 10.

Number of Steps Missing (Backward Thinking Task)

A significant main effect was found for gender for the number of steps missing, as shown in Table 11. The means for males and females for the number of steps missing were 62.512 and 45.999 respectively. Females had a significantly lower percentage of steps missing than did males. No other significant main effects or interaction effects were found for the number of steps missing.

Table 10

ANOVA Results for Pathfinder Coherence Index (Ratings Task)

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	.0389	.68	.4122
Task Order	1	.0952	1.67	.2018
Test Time	2	.0150	.13	.8766
Task Order x Test Time	2	.0003	.00	.9969
Gender x Task Order	1	.0003	.01	.9413
Gender x Test Time	2	.0190	.17	.8472
Gender x Task Order x Test Time	2	.2152	1.88	.1612

Backward thinking score (Backward Thinking Task)

A significant main effect was found for gender for the backward thinking score, as shown in Table 12. The means for males and females for the backward thinking score were 44.484 and 61.375 respectively. Females had a significantly higher backward thinking score than did males. No other significant main effects or interaction effects were found.

Table 11

ANOVA Results for Number of Steps Missing (Backward Thinking Task)

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	2856.1091	4.06	.0497*
Task Order	1	29.1871	.04	.8395
Test Time	2	1292.4312	.92	.4062
Task Order x Test Time	2	606.2243	.43	.6525
Gender x Task Order	1	632.7405	.90	.3478
Gender x Test Time	2	1672.6036	1.19	.3136
Gender x Task Order x Test Time	2	560.5346	.40	.6737

Project Score

Significant main effects were found for test time for project score, as shown in Table 13. The means for test times one through three for the project score were 94.117, 86.114, and 82.729 respectively. For test time, the project score was significantly higher at test time one than at test times two or three. There were no significant differences between the project scores when comparing test time one and three, or when comparing test time two and three. It is interesting to note the trend that the project score decreased over test time.

A significant interaction effect was found for Gender x Task Order for project score as shown in Table 13. The means for Gender x Task Order for project score are shown in Table 14. The post hoc test confirmed that males who completed the KA task after the project had significantly higher project scores than females who completed the KA task after the project.

Table 12

ANOVA Results for Backward Thinking Score (Backward Thinking Task)

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	2988.2385	5.46	.0238*
Task Order	1	146.8406	.27	.6070
Test Time	2	1357.6203	1.24	.2987
Task Order x Test Time	2	260.0917	.24	.7895
Gender x Task Order	1	849.9194	1.55	.2189
Gender x Test Time	2	2097.0203	1.92	.1586
Gender x Task Order x Test Time	2	462.5447	.42	.6579

Post-test Score

Significant main effects were found for gender for post-test score, as shown in

Table 15. The means for males and females for the project score were 24.341 and 26. 059, respectively. Females had a significantly higher post-test score than did males. No other significant main effects or interaction effects were found for post-test score.

Table 13

ANOVA Results for Project Score

Source	DF	ANOVA SS	F Value	P
Gender	1	531.9677	2.00	.1607
KA Task	1	49.4018	.19	.6677
Task Order	1	32.4627	.12	.7278
Test Time	2	2518.6703	4.72	.0108*
Gender x KA Task	1	14.4028	.05	.8167
Gender x Task Order	1	1124.2266	4.22	.0425*
KA Task x Task Order	1	3.1099	.01	.9142
Gender x KA Task x Task Order	1	104.9420	.39	.5318
Gender x Test Time	2	494.0708	.93	.3991
KA Task x Test Time	2	175.3702	.33	.7205
Task Order x Test Time	2	71.7081	.13	.8743
KA Task x Task Order x Test Time	2	774.1080	1.45	.2388
Gender x Task Order x Test Time	2	621.9123	1.17	.3155
Gender x KA Task x Test Time	2	207.7766	.39	.6783
Gender x KA Task x Task Order x Test Time	2	664.3892	1.25	.2918

Table 14

Means for Project Score: Gender x Task Order

	Before Project	After Project
Male	87.161	92.574
Female	89.248	81.629

Table 15

ANOVA Results for Post-test Score

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	77.1417	3.97	.0488*
KA Task	1	15.6302	.80	.3719
Task Order	1	36.7366	1.89	.1720
Gender x KA Task	1	3.8793	.20	.6560
Gender x Task Order	1	45.1041	2.32	.1306
KA Task x Task Order	1	34.7013	1.78	.1843
Gender x KA Task x Task Order	1	.1814	.01	.9232

Final Score

No significant main effects or interaction effects were found for final score as shown in Table 16.

Table 16

ANOVA Results for Final Score

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	61.2650	.82	.3677
KA Task	1	4.7796	.06	.8010
Task Order	1	212.3180	2.83	.0950
Gender x KA Task	1	18.9442	.25	.6160
Gender x Task Order	1	34.4698	.46	.4989
KA Task x Task Order	1	177.9814	2.38	.1260
Gender x KA Task x Task Order	1	40.4239	.54	.4641

Correlational Analysis

For each project, a correlational analysis was performed to examine the correlations between the following variables: project score, index of similarity, coherence index, the number of steps missing, and backward thinking score. The correlational results are shown in Tables 17-19. For project one, only the correlation between the backward thinking score and the number of steps missing was significant. For project two, only the correlation between the backward thinking score and the number of steps missing was significant. For project three, two correlations were significant: the correlation between the backward thinking score and the number of steps missing; and the correlation between the project three score and the index of similarity (a dependent variable from the ratings task). All significant correlations are positive and above .500.

Table 17

<u>Pearson Correlation Coefficients for Project One</u>

	Project 1 Score	Index of Similarity (Ratings Task)	Coherence Index (Ratings Task)	Backward thinking score (Backward Thinking Task)	Steps Missing (Backward Thinking Task)
Project 1 Score	1.000				
Index of Similarity (Ratings Task)	0.396 0.055	1.000			
Coherence Index (Ratings Task)	-0.042 0.844	0.076 0.725	1.000		
Backward thinking score (Backward Thinking Task)	-.058 0.807	Not applicable	Not applicable	1.000	
Steps Missing (Backward Thinking Task)	-0.137 0.565	Not applicable	Not applicable	-0.756 0.0001*	1.000

Note. Within each table cell, the correlation coefficient is reported as the top number and the P value is reported as the bottom number. A cell entry of “Not Applicable” means that a correlation between the two dependent measures was not possible as participants performed either the ratings or the backward thinking task, but not both.

Table 18

Pearson Correlation Coefficients for Project Two

	Project 2 Score	Index of Similarity (Ratings Task)	Coherence Index (Ratings Task)	Backward thinking score (Backward Thinking Task)	Steps Missing (Backward Thinking Task)
Project 2 Score	1.000				
Index of Similarity (Ratings Task)	-0.131 0.541	1.000			
Coherence Index (Ratings Task)	0.239 0.261	-0.206 0.335	1.000		
Backward thinking score (Backward Thinking Task)	-0.152 0.949	Not applicable	Not applicable	1.000	
Steps Missing (Backward Thinking Task)	0.136 0.566	Not applicable	Not applicable	-0.915 <0.0001*	1.000

Note. Within each table cell, the correlation coefficient is reported as the top number and the P value is reported as the bottom number. A cell entry of “Not Applicable” means that a correlation between the two dependent measures was not possible as participants performed either the ratings or the backward thinking task, but not both.

Table 19

Pearson Correlation Coefficients for Project Three

	Project 3 Score	Index of Similarity (Ratings Task)	Coherence Index (Ratings Task)	Backward thinking score (Backward Thinking Task)	Steps Missing (Backward Thinking Task)
Project 3 Score	1.000				
Index of Similarity (Ratings Task)	0.501 0.013*	1.000			
Coherence Index (Ratings Task)	-0.344 0.100	-0.057 0.792	1.000		
Backward thinking score (Backward Thinking Task)	0.201 0.396	Not applicable	Not applicable	1.000	
Steps Missing (Backward Thinking Task)	-0.159 0.503	Not applicable	Not applicable	-0.863 <.0001*	1.000

Note. Within each table cell, the correlation coefficient is reported as the top number and the P value is reported as the bottom number. A cell entry of “Not Applicable” means that a correlation between the two dependent measures was not possible as participants performed either the ratings or the backward thinking task, but not both.

Subjective Ratings

Clarity of KA task instructions. Significant main effects were found for KA task for how participants perceived the clarity of KA task instructions (ease of use question one), as shown in Table 20. A higher mean value means that the participant rated the instructions as being less clearly stated. The means for clarity of KA task instructions for the backward thinking and ratings group were 1.019 and 1.345 respectively. Participants who performed the ratings task rated the instructions as significantly less clearly stated than participants who performed the backward thinking task. No other significant main effects or interaction effects were found for clarity of KA task instructions.

Difficulty of the KA task. A significant main effect was found for KA task for how participants perceived the difficulty of the KA task (ease of use question two), as shown in Table 21. A higher mean value means that the participant rated the task as being more difficult to perform. The means for difficulty of the KA task for the backward thinking and ratings group were 1.777 and 2.684, respectively. Participants who performed the ratings task rated the KA task as significantly more difficult to perform than participants who performed the backward thinking task.

Significant interaction effects were found for Gender x KA Task x Task Order for difficulty of the KA task as shown in Table 21. The means for Gender x KA Task x Task Order are shown in Table 22. The post-hoc test confirmed that females who performed either the ratings task or the backward thinking task prior to completing the project rated the KA task as more difficult to perform than their male counterparts. However, males who performed the ratings task after completing the project rated the KA task as more difficult to perform than females who performed the ratings task after completing the project. There were no significant differences when comparing males and females who completed the backward thinking task after the project. The post-hoc test also confirmed that the ratings task was significantly more difficult to perform than the backward thinking task, with the only exception being the comparison of females who performed either the ratings task or the backward thinking task after the project; no significant differences were found between those two groups.

Table 20

ANOVA Results for Clarity of KA Task Instructions

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	.4631	2.91	.0906
KA Task	1	2.7085	17.03	<.0001*
Task Order	1	.0096	.06	.8064
Gender x KA Task	1	.3063	1.93	.1678
Gender x Task Order	1	.1317	.83	.3648
KA Task x Task Order	1	.0133	.08	.7725
Gender x KA Task x Task Order	1	.1295	.81	.3687

Lastly, the post-hoc test confirmed that males who performed either the ratings task or the backward thinking task after completing the project rated it as being significantly more difficult to perform than males who performed the ratings task or backward thinking task prior to completing the project. However, females who performed the ratings task prior to completing the project rated it as being significantly more difficult to perform than females who performed the ratings task after the project. No significant differences were found when comparing females who performed the backward thinking task before or after the project.

Table 21

ANOVA Results for Difficulty of the KA Task

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	.2021	.29	.5900
KA Task	1	20.9454	30.26	<.0001*
Task Order	1	.0099	.01	.9052
Gender x KA Task	1	.7287	1.05	.3070
Gender x Task Order	1	20.9073	30.21	<.0001*
KA Task x Task Order	1	.0243	.04	.8517
Gender x KA Task x Task Order	1	2.7890	4.03	.0471*

Satisfaction with time spent performing KA task. A significant main effect was found for KA task for participant's satisfaction with time spent performing KA task (ease of use question three), as shown in Table 23. A higher mean value means that the participant was less satisfied with the time spent performing the KA task. The means for time spent for performing the KA task for the backward thinking and ratings group were 1.457 and 1.720, respectively. Participants who performed the ratings task were significantly more dissatisfied with the amount of time spent performing the task than participants who performed the backward thinking task.

Table 22

Means for Difficulty of the KA Task: Gender x KA Task x Task Order

	Backward Thinking		**Ratings**	
	Before	**After**	**Before**	**After**
Male	1.359	1.935	2.082	3.369
Female	2.205	1.607	3.236	2.049

A significant interaction effect was found for Gender x KA Task x Task Order for satisfaction with time spent performing the KA task as shown in Table 23.

Table 23

ANOVA Results for Satisfaction with Time Spent Performing the KA Task

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	.7376	1.68	.1970
KA Task	1	1.7527	4.00	.0478*
Task Order	1	.5167	1.18	.2796
Gender x KA Task	1	.0813	.19	.6673
Gender x Task Order	1	1.4503	3.31	.0714
KA Task x Task Order	1	.0236	.05	.8168
Gender x KA Task x Task Order	1	4.4931	10.26	.0018*

The means for Gender x KA Task x Task Order for ease of use question three are shown in Table 24.

Table 24

Means for Satisfaction with Time Spent Performing the KA Task: Gender x KA Task x Task Order

	Backward Thinking		**Ratings**	
	Before	**After**	**Before**	**After**
Male	1.544	1.259	1.366	1.848
Female	1.481	1.543	2.244	1.420

The post-hoc test confirmed that females who performed the ratings task prior to completing the project and males who performed the ratings task after completing the project were significantly less satisfied with the time spent performing the KA task than participants in all other conditions. There was no significant difference between the females who performed the ratings task prior to completing the project and the males who completed the ratings task after the project. The post-hoc test also confirmed that males who performed the ratings task after the project were significantly less satisfied with the time spent performing the KA task than males who performed the backward thinking task after completing the project. Yet, females who performed the ratings task prior to completing the project were significantly less satisfied with the time spent performing the KA task than females who performed the backward thinking task prior to completing the project. Lastly, the post-hoc test confirmed that males who performed the ratings task after the project were significantly less satisfied with the time spent performing the KA task than males who performed the ratings task prior to completing the project. However, females who performed the ratings task prior to the project were significantly less satisfied with the time spent performing the KA task than females who performed the ratings task after the project. There were no significant differences with the satisfaction of the time spent performing the KA task, when comparing males and females who performed the backward thinking task, either before or after the project.

Helpfulness of the KA task in understanding course material. A significant main effect was found for gender for helpfulness of the KA task in understanding the course material (ease of use question four), as shown in Table 25. A higher mean value means that the participant rated the helpfulness of the KA task in understanding the course material as less helpful. The means for males and females for helpfulness of the KA task in understanding the course material were 2.655 and 2.398 respectively. Males rated the helpfulness of the knowledge acquisition task as significantly less helpful in learning the course materials than did females.

A significant interaction effect for Gender x Task Order was found for helpfulness of the KA task in understanding course material as shown in Table 25. The means for Gender x Task Order for ease of use question four are shown in Table 26. The post hoc test confirmed that males who completed the KA task prior to completing the project rated the helpfulness of the KA task in understanding the course material as significantly less helpful than males who performed the KA task after completing the project or females who completed the KA task prior to completing the project.

Table 25

ANOVA Results for Helpfulness of the KA Task in Understanding the Course Material

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	1.7262	4.94	.0283*
KA Task	1	.0269	.08	.7821
Task Order	1	.0264	.08	.7841
Gender x KA Task	1	.0086	.02	.8758
Gender x Task Order	1	2.6328	7.53	.0071*
KA Task x Task Order	1	1.1071	3.17	.0779
Gender x KA Task x Task Order	1	.6154	1.76	.1873

Table 26

Means for Helpfulness of the KA Task in Understanding the Course Material: Gender x Task Order

	Before Project	**After Project**
Male	2.833	2.477
Female	2.252	2.544

Incorporation of the KA task into the course. A significant main effect for task order was found for how well the KA task was incorporated into the course (ease of use question five), as shown in Table 27. The means for the before group and the after group for incorporation of the KA task into the course were 2.301 and 2.765 respectively (A higher mean value means that the participant thought that the KA task was more poorly incorporated into the task.). Participants who performed the KA task after completing the project thought that the KA task was significantly more poorly incorporated into the course than did participants who performed the KA task prior to completing the project. No other significant main effects or interaction effects were found.

Table 27

ANOVA Results for Incorporation of the KA Task into the Course

Source	**DF**	**ANOVA SS**	**F Value**	**P**
Gender	1	.0771	.06	.8037
KA Task	1	3.7802	3.04	.0838
Task Order	1	5.5544	4.47	.0367*
Gender x KA Task	1	1.4447	1.16	.2832
Gender x Task Order	1	.2382	.19	.6623
KA Task x Task Order	1	.1548	.12	.7248
Gender x KA Task x Task Order	1	1.0399	.84	.3622

DISCUSSION

Evaluation of Hypotheses

Comparison of Pathfinder index of similarity and performance. It was hypothesized that there would be a positive correlation between the index of similarity and the project score. This hypothesis was partially supported, as the correlation between the index of similarity and project score was significant only at test time three. Based on comments from the course instructor, the underlying assumption of the study was that the complexity of the projects increased over the course of the semester. The number of concepts rated increased over the course of the semester with the participants having rated the pairwise comparisons for nine concepts for project one, fifteen concepts for project two, and twenty concepts for project three. At test time three, participants had to incorporate the highest number of concepts into their conceptual model of the project as well as apply their knowledge of those concepts to the programming project. It may be that both the KA and the performance measures had to represent a high enough degree of task difficulty in order for this relationship to become significant. It is also possible that the number of concepts rated that had to reach a large enough number in order for the index of similarity to be sensitive to differences.

Effect of test time on the Pathfinder index of similarity. It was hypothesized that as the student completed each successive project that the index of similarity would increase over the course of the semester. Instead of having the highest index of similarity at project time three, followed by project time two, then project time one as predicted, the index of similarity was at its highest at project time one, its next highest at project time three, and was significantly lowest at project time two. Therefore, this hypothesis was not supported because the index of similarity never increased beyond the value obtained at project time one. This finding indicates that it may not be the number of concepts rated that influences the difficulty of the KA task but rather how conceptually difficult the concepts themselves are. It may be that the concepts underlying project two were conceptually more difficult than the concepts underlying either project one or three, and that the student's mental model for the second project was less detailed and accurate than their mental model of the other projects. However, the conceptual difficulty of the set of concepts underlying each project as a whole was not experimentally assessed. Future research studies should incorporate the assessment of the conceptual difficulty into the experimental methodology (Lester, Converse, Stone, Kahler, and Barlow, 1997).

There exists a similar pattern for project score. The project score successively decreased over the three test times; with the project score significantly higher at test time one than at test time three. It is likely that the findings for project score support the underlying assumption of the study, in that the difficulty of the projects increased over the course of the semester. It could also be concluded from the pattern of the indices of similarity and project scores that the students did not learn over the course of the semester.

It may be that the instructor did not verbalize the relationships between concepts very thoroughly, as his index of coherence follows the same pattern as the index of similarity, with the coherence index being its highest at project time one, its next highest at project time three, and its lowest at project time two. Also, the instructor had to repeat the rating tasks for projects one and three because his coherence score did not reach a high enough criterion level. Therefore, it appears that his mental model influenced the accuracy of the conceptual model that he conveyed to his students, which in turn is reflected in the pattern of the values for the indices of similarity for the students as previously discussed. This finding conveys that in addition to carefully measuring a mental model, it is equally as important to have a source from which a highly accurate mental model can be elicited and used as a gold standard.

<u>Effect of test time on the Pathfinder coherence index.</u> It was hypothesized that as the student completed each successive project, the coherence index would increase. However, this hypothesis was not supported, as no main effect of test time on the coherence index was found. The means for the coherence index for test times one through three were .281, .246, and .270, from which no significant differences were obtained. It may be that the participants employed a consistent rating strategy over the duration of the study as it appears that their rating strategy was unaffected by the introduction of additional concepts over the course of the study. It is possible that participants need longer than one semester to improve their consistency in evaluating the relationships between concepts, as Acton (1991) found that students' index of coherence did increase, but it was over the period of two semesters. Future research studies are needed to better

understand how the coherence index develops over longer periods of time.

<u>Comparison of ratings task and backward thinking task with project score</u>. It was hypothesized that the index of similarity (generated from the ratings task) would correlate more positively with a student's project score than would the number of steps out of sequence (generated from the backward thinking task). This hypothesis could not be fully evaluated because the number of steps out of sequence was omitted from further data analysis, due to the low inter-rater reliability. However, for project three, the index of similarity was significantly positively correlated with the project score. This finding is particularly interesting because it establishes that the index of similarity, which is derived from a KA task typically used to structure declarative mental models, can be applied in a procedural domain such as computer programming.

Although other studies have validated the use of the index of similarity to represent procedural knowledge (Cooke, Neville and Rowe, 1996; Rowe, Cooke, Hall, and Halgren, 1996; Vortac, Edwards, and Manning, 1994), there are several differences between this study and those studies. The first difference is that in this study, participants completed both a KA task and a performance task (e.g., programming project). In Cooke, Neville and Rowe (1996), participants did not complete a KA task. Instead, their performance on a simulated mission control task was used as input into the Pathfinder algorithm. Additionally, participants received feedback on each task, which could have impacted their performance on the next task. In this study, no feedback on a project was provided prior to completing the KA task. The second difference is that this study employed a performance measure in which the participants were evaluated based on the outcome of their task performance. In Rowe, Cooke, Hall, and Halgren (1996) and Vortac, Edwards, and Manning, (1994), the participants were graded by observers while describing the task steps, instead of actually performing the task steps. This procedure appears to be more commonly used to capture knowledge organization rather than reflect task performance.

Evaluation of Backward Thinking

Correlation between backward thinking score and number of steps missing. During the correlational analysis, the backward thinking score was significantly negatively correlated with the number of steps missing for all three projects. As the backward thinking score increased, the percentage of the number of steps missing decreased. The total value assigned to the backward thinking score was comprised of the number of steps missing and the number of steps out of sequence. Therefore, it is not surprising that a component of the backward thinking score should correlate with the total value of the backward thinking score.

Comparison of backward thinking and performance. None of the correlations between the number of steps missing and project score were significant. It is likely that the project score was not a sensitive enough measure to correlate with the number of steps missing, as the project score was a composite score rather than just a representation of the number of steps missing for the project. Had the actual number of steps missing for the project score been available, as identified from the actual programming code submitted by students, a comparison between that and the KA measure of the number of steps missing would have been possible. Future research studies should include the examination of the actual programming code to allow for the comparison of the number of steps missing as identified during the KA task and the number of steps missing as identified from the programming task.

There was a trend towards a negative correlation between backward thinking score and project score for projects one and two. It appears then that participants, while able to communicate the steps to complete the programming task in the backward thinking task, they were unable to apply their knowledge to the task of implementing the programming procedures necessary to complete the project. Perhaps participants were not able to translate that knowledge into the syntax of the programming language but were able to translate it into the format of the backward thinking task. The backward thinking protocols do not ask the participant to supply actual code, where as the project does. Hence, neither the backward thinking score nor the number of steps missing were good predictors of task performance.

<u>Effect of Task Order</u>

No significant main effect of task order was found for any of the ratings task measures (the index of similarity, coherence index) or backward thinking task measures (backward thinking score, number of steps missing). Johnson (1995) speculated that performing the knowledge acquisition task might provide some task learning. Although no hypotheses were generated for task order, it was important to employ task order as an independent variable, due to the potential impact of task learning. For this study, it appears then that it did not matter whether the ratings task or the backward thinking task was performed before or after the project was completed. This indicates that performing the KA task before the project did not inflate the project scores. It should be stated that for those participants who completed the KA task prior to the project, there was a maximum of two weeks between completing the ratings task and the project. It may be that task order becomes a critical factor when the lapse between the KA task and performance task is longer, as it is anticipated that the participants' retention of the relationship between concepts and the order of the task steps would decrease over time.

Gender Effects

Effect of gender on backward thinking score. Females had a significantly higher backward thinking score than did males. No hypotheses were made regarding the effect of gender on the backward thinking score. This finding is unexpected, as no significant main effects of gender on the other KA measures were found. The graders were unaware of the gender of the participants, as each participant's backward thinking protocol was identified only by a number on a computer-generated print out. It may be that the females were more thorough in their creation of the backward thinking protocols than were the males, due to the verbal element of describing the detailed task steps during the backward thinking task. Meta-analyses of studies comparing gender differences in verbal abilities have detected a difference between males and females, with females having a slight advantage (Hyde and McKinley, 1997). However, the backward thinking task could not be described as purely verbal or spatial, as participants had to supply a description of the task step (most likely a verbal effort) while placing the task steps in the correct order (most likely a spatial effort). Therefore, it appears that performing the backward thinking task may have tapped into a blend of verbal and spatial abilities in which females have an advantage. Future research efforts need to elicit more thorough descriptions of the task steps from the course instructor to make it easier for the graders to identify whether or not the protocols contain a particular step (e.g., number of steps missing).

Effect of gender on number of steps missing. Females had a significantly lower percentage of steps missing than did males. No hypotheses were made regarding the effect of gender on the number of steps missing. This finding is expected, as the total value assigned to the backward thinking score was comprised of the number of steps missing and the number of steps out of sequence. Therefore, for the females to have a higher backward thinking score, they also had to have a lower percentage of steps missing.

Effect of gender on post-test. Females had a significantly higher post-test score than males. This finding is surprising as there were no hypotheses made regarding gender. The other experimental data do not provide an explanation as to why this finding appeared. Future research efforts may want to include post-test interviews to identify potential reasons as to why females would perform better on a post-test than males.

Effect of gender on project score. Males who completed the KA task after the project had significantly higher project scores than females who completed the KA task after the project. This result cannot be attributed to learning due to the KA task, as the project was turned in prior to performing the KA task. This is an interesting finding, because the females did better on the post-test than the males, but did not do as well on the projects as did the males. However, females also achieved a higher backward thinking score than did males. The underlying theme here is that females appeared to perform better in the experimental setting, as both the backward thinking task and the post-test were performed in that setting, while males performed better in the classroom setting where the projects were completed. It is possible that the females were more comfortable in the experimental testing environment and were able to perform better. However, no measures were collected to assess participants' comfort level with the experimental or classroom setting. Future research efforts may want to include post-test interviews to identify how comfortable participants were with the experimental setting.

Effect of gender on the index of similarity. There was a significant Gender x Task Order interaction for the index of similarity. The post hoc test confirmed that females who completed the ratings task prior to completing the project had a significantly higher index of similarity than females who completed the ratings task after completing the project. There are two possible explanations for this finding. The first is that the females who completed the ratings task after the project may not have put much effort into the ratings task because the concepts were being rated after they had already turned in their projects, at which time they may not have perceived any value in performing the ratings task. Their subjective assessments of the ratings task may assist in the further interpretation of this finding, as females who completed the ratings task after the project did not rate the ratings task as being difficult to perform. Neither were they dissatisfied with the amount of time spent performing the ratings task. This is opposite of the ratings assigned by females who completed the ratings task prior to the project. Those females rated the ratings task as being significantly more difficult to perform and they were significantly less satisfied with the time spent performing the KA task. The second explanation is that females who completed the ratings task after the project did not have the conceptual model necessary for them to successfully rate the concepts, thus explaining the lower index of similarity. Of course, it is possible that both explanations could go hand in hand; they did not have a well developed conceptual model of the concepts which did not require them to spend either the time or the effort to complete the ratings task, both of which could have contributed to a lower index of similarity.

Subjective Ratings

Clarity of KA task instructions. Participants who performed the ratings task rated the instructions as significantly less clearly stated than participants who performed the backward thinking task. Prior to completing the KA task using project related materials, participants were given a training task on their respective KA task (ratings or backward thinking) to help them become familiar with what was required of them to complete the task. No participants verbally expressed any difficulty in using the online ratings tool during the testing sessions. Future research efforts may need to include a basic test of the participant's ability to perform the KA task, prior to data collection.

Difficulty of the KA task. Participants who performed the ratings task rated the task as significantly more difficult to perform than participants who performed the backward thinking task. The most likely explanation for this finding is that participants may have been more comfortable with describing task steps than rating concepts, due to the procedural nature of a programming task. It has been hypothesized that backward thinking is a more appropriate KA task than ratings to obtain procedural knowledge (Converse and Kahler, 1992). Another possible explanation for this finding is that participants who performed the ratings task did not understand the instructions to perform the task, as has previously been discussed. If the task instructions were unclear, then it is plausible for the participants to express difficulty in performing the task.

Males who performed the KA task after completing the project rated it as being significantly more difficult to perform than males who performed the KA task prior to completing the project. It may be that the males who had already completed the project were already focused towards learning the new concepts that were applicable to the next project and therefore found it difficult to concentrate on only the relevant concepts for a previous project. However, females who performed the ratings task prior to completing the project rated it as being significantly more difficult to perform than females who performed the ratings task after the project. Perhaps the females who performed the ratings task after the project gained some experience with the concepts by applying them to the project, which would reduce the difficulty associated with rating the concepts.

Satisfaction with time spent performing KA task. Participants who performed the ratings task were significantly more dissatisfied with the amount of time spent performing the task than participants who performed the backward thinking task. Although no objective measure of task completion time was collected, it was observed that participants generally did take longer to perform the ratings task than the backward thinking task, as the number of pairwise comparisons ranged from 36 to 153, across the three projects. In the case of the backward thinking task, the number of instructor generated steps in the backward thinking protocols ranged from eight to ten. It appears then that because the ratings task took so long to complete that the participants disliked it as expressed in the higher dissatisfaction ratings.

Females who performed the ratings task prior to completing the project and males who performed the ratings task after completing the project were significantly less satisfied with the time spent performing the KA task than the remaining participants. It may be that there is a relationship between the KA task difficulty and satisfaction with the time spent performing the KA task, in that a KA task that requires more time to complete may be perceived as being more difficult to complete.

<u>Helpfulness of the KA task in understanding course material.</u> Females rated the helpfulness of the KA task as significantly more helpful in learning the course materials than did the males. Furthermore, males who completed the KA task prior to completing the project rated the helpfulness of the KA task in understanding the course material as significantly less helpful than either females who completed the KA task prior to completing the project or males who performed the KA task after completing the project. Although females thought that the KA task was more helpful in learning the course

materials, they did not perform better on the projects than the males. This is an example of a dissociation between participants' subjective ratings and their task performance. Dissociation effects were found in Lester, Converse, Kahler, Barlow, Stone, and Bhogal (1997), in that students expressed a strong preference for a particular type of pedagogical agent, but instruction from that agent did not result in better task performance than if the student had received instruction from other types of agents. Future research efforts may want to further examine gender differences by incorporating some open-ended questions to help interpret the subjective ratings assigned by participants.

Incorporation of the KA task into the course. Participants who performed the KA task after completing the project thought that the KA task was significantly more poorly incorporated into the course than did participants who performed the KA task prior to completing the project. It may be that participants who performed the KA task after the project thought it unusual to focus on project concepts after the project had been completed. Instead, they may have expected to focus on the relevant concepts for the next project.

Summary

The only KA measure to have a significant correlation with project score was the index of similarity as generated from the ratings task. This finding establishes the predictive validity of incorporating the indices of similarity into an experimental study, in addition to task performance measures. The caveat is that task difficulty appears to be an important factor and should also be assessed during the course of the study, as the correlation between the index of similarity and the project score was only significant at project three, which was assumed to be the most difficult project of the three assigned to the students.

Participants subjectively rated the ratings task as having instructions that were less clearly stated, as being more difficult to perform, and they were less satisfied with the amount of time spent performing the ratings task. Most likely there was a flaw in the clarity of the instructions for the ratings task because the instructions for the ratings task were not as easily understood by participants as the instructions for the backward thinking task. It also appears that there is a dissociation between the predictive validity and the subjective assessment of the ratings task. As mentioned previously, research by Lester, Converse, Kahler, Barlow, Stone, and Bhogal (1997) has also found dissociation effects. Although the ratings task is a more valid measure to use to predict a student's project score, the students were not particularly pleased with performing the ratings task. However, this does not mean that experimental studies should only incorporate KA tasks that the participants find agreeable.

Recommendations for Conducting Knowledge Acquisition Research

Several recommendations for conducting knowledge acquisition research can be derived from this study. First, it appears that it does not matter whether the KA task is performed either before or after the performance task that is being evaluated. This provides evidence that participating in the KA task did not affect performance or enhance learning on the project. This was true for two very different types of KA tasks, ratings and backward thinking. In this study, participants performed the KA task within a few weeks of completing the project. It may be that task order would play a more significant role if the time between KA task and the performance task was longer, and the decrease in retention could be observed. However, with that being said, participants who performed the KA task after the programming task viewed the KA task as being poorly incorporated into the course. Since there was no experimental effect present due to task order, then it is recommended that study participants complete the KA task prior to the performance task.

Second, the use of subjective assessments to evaluate a student's conceptual model should be made using two or more judges and the inter-rater reliability should be established before data collection. It is still possible to establish a high inter-rater reliability early on, as was the case with the number of steps out of sequence for project one, and then have it decrease over time. However, a high inter-rater reliability is critical. Without a high inter-rater reliability, the results cannot be considered valid.

Third, performance measures and KA measures need to be at the same level of analysis. This recommendation is derived from the fact that there were no comparisons made between the number of steps missing from the actual programming code submitted by students and the KA measure of the number of steps missing from the backward thinking task. For example, if the performance measure is an overall measure of the task (e.g., project score), then the KA measure should also be an overall measure of the conceptual model (e.g., index of similarity, backward thinking score). However, if the KA measure represents a partial conceptual model, then the performance measure should represent that same subset of the task. It will be difficult to interpret findings that are not at the same level of analysis.

Fourth, the difficulty of a KA task may not be determined solely by either the number of concepts or the number of task steps that are expected to be incorporated in the conceptual model. Instead, the difficulty of a KA task is based on how conceptually difficult the concepts are to learn and how difficult the task steps are to perform. Future research efforts may want to assess the KA task's difficulty by asking a panel of domain experts as well as the participants in the study determine the conceptual difficulty of the various projects or exercises used in the research study.

Lastly, there were a number of gender effects found, both main and interaction, in this study. Kahler, Converse, Lester, Cheatham, and Stelling (2000), also found gender effects when evaluating an animated pedagogical agent. It appears that gender should not be ignored in research studies involving computer related environments as it plays an important role.

Implications for Teaching Introductory Computer Programming

Two main implications for teaching introductory computer programming can be derived from this study. The first implication is that while a student may understand the steps needed to complete a programming task, they do not necessarily understand how the syntax of the computer programming language must be implemented to successfully complete the programming task. It appears then that computer programming has two different procedural components, the overall order of the task steps, and the syntax within a particular step, both of which require emphasis during course instruction. The second implication is that it would be reasonable to incorporate ratings tasks into the course. The ratings task has predictive validity and its analysis could be extended to assist in diagnosing where students have problems understanding the relationships between course concepts. Even though backward thinking protocols did not demonstrate predictive validity in this study, they could be applied in a diagnostic capacity to identify the exact task steps that the student omits or applies out of order, rather than predict performance. However, the incorporation of a KA task into the classroom setting should occur prior to any programming activities, as the students in this study viewed the completion of a KA task after the project as not being very tightly integrated into the course materials.

REFERENCES

Acton, W.H. (1991). Comparison of criterion referenced and criterion free measures of cognitive structure. Doctoral dissertation, University of New Mexico, Albuquerque.

Anderson, J.R. (1983). The architecture of cognition. Cambridge, MA: Harvard University Press.

Anderson, J.R. (1985). Cognitive psychology and its implications. New York: Freeman.

Andrus, G.R. (1988). Methods of acquiring expert knowledge on media selection for interactive video. Doctoral dissertation, Peabody College for Teachers of Vanderbilt University, Nashville, TN.

Bainbridge, L. (1979). Verbal reports as evidence of the process operator's knowledge. International Journal of Man-Machine Studies, 11, 411-436.

Barker, P.G. and van Schaik, P., (1999). Mental models and their implications for the design of computer-based learning resources. CBLIS '99, Fourth International Conference on Computer Based Learning in Science, 2nd July - 6th July 1999, Twente University, Enschede, the Netherlands.

Chao, C. and Salvendy, G. (1994). Percentage of procedural knowledge acquired as a function of the number of experts from whom knowledge is acquired for diagnosis, debugging, and interpretation tasks. International Journal of Human-Computer Interaction, 6(3), 221-233.

Converse, S.A., and Kahler, S.E. (1992). Knowledge acquisition and the measurement of shared mental models, (Contract No. DAAL03-86-D-0001). Orlando, FL: Naval Air Warfare Center Training Systems Division.

Cooke, N.J. (1994). Varieties of knowledge elicitation methods. International Journal of Human-Computer Studies, 41, 801-849.

Cooke, N.J., and McDonald, J.E. (1986). The application of psychological scaling methods to knowledge elicitation for knowledge-based systems. International Journal of Man-Machine Studies, 26, 533-550.

Cooke, N.J., Neville, K.J., and Rowe, A.L. (1996). Procedural network representations of sequential data. Human-Computer Interaction, 11, 29-68.

Cooke, N.J. and Schvaneveldt R.W. (1988). Effects of computer programming experience on network representations of abstract programming concepts. International Journal of Man-Machine Studies, 29, 407-427.

Gammock, J.G., and Young, R.M. (1985). Psychological methods for eliciting expert knowledge. In M.A. Bramer (Ed.), Research and development in expert systems (pp.105-112). Cambridge, MA: Cambridge University Press.

Goldsmith, T.E. (Personal Communication, 2001). Criterion values for the Pathfinder Coherence Index.

Goldsmith, T.E., and Davenport, D.M. (1990). Assessing structural similarity of graphs. In R. Schvaneveldt (Ed.), Pathfinder Associative Networks Studies in Knowledge Organization. (pp.75-87). Norwood, NJ: ABLEX Publishing Corporation.

Gordon, S.E. (1992). Implications of cognitive theory for knowledge acquisition. In (R.Hoffman, Ed.), The psychology of expertise: Cognitive research and empirical AI (pp. 99-120). Springer-Verlag: New York.

Gordon, S.E., and Gill, R. (1992). Knowledge acquisition with question probes and conceptual graph structures. In T. Lauer, E. Peacock, and A. Graesser (Eds.), Questions and Information Systems (pp.29-46). Hillsdale, NJ: Erlbaum.

Hoffman, R. (1992). Doing psychology in an AI context: A personal perspective and introduction to this volume. In R. Hoffman (Ed.), The psychology of expertise: Cognitive research and empirical AI (pp.3-11). New York: Springer-Verlag.

Hoffman, R. (1989). A brief survey of methods for extracting the knowledge of experts, SIGART Newsletter, Special Issue on Knowledge Acquisition, 108 (April) 19-27.

Hyde, J.S., & McKinley, N.M. (1997). Gender differences in cognition: Results from meta-analyses. In P.J. Caplan, M. Crawford, J.S. Hyde, and J.T. Richardson (Eds.), Gender differences in cognition (pp.30-51). New York, NY: Oxford University Press.

Jackson, D.M. (1970). A sequential system for personality scale development. In C. Spielberger, Current topics in clinical and community psychology (pp.61-96), Vol.s, Academic Press.

Johnson, S.D. (1995). Invited reaction: Measuring mental models-Thoughts on a research agenda. Human Resource Development Quarterly, 6(3), 257-262.

Kahler, S., Converse, S., Lester, J., Cheatham, D., and Stelling, G. (2000). Effect of intelligent instruction on students' mental models and performance. Proceedings of the

IEA 2000/HFES 2000 Congress, 1, 101-104.

Kellogg, W. A., and Breen, T.J. (1990). Using Pathfinder to evaluate user and system models. In R. Schvaneveldt (Ed.), Pathfinder Associative Networks Studies in Knowledge Organization. (pp.179-195). Norwood, NJ: ABLEX Publishing Corporation.

Kieras, D. (1988). What mental model should be taught: Choosing instructional content for complex engineered systems. In J. Psotka, L. Massey, and S. Mutter (Eds.) Intelligent Tutoring Systems (pp. 85-111). New York: Erlbaum.

Lester, J.C., Converse, S.A., Kahler, S.E., Barlow, S.T., Stone, B.A., and Bhogal, R. (1997). The persona effect: Affective impact of animated pedagogical agents. Proceedings of CHI'97, 359-366.

Lester, J.C., Converse, S.A., Stone, B.A., Kahler, S.E., and Barlow, S.T. (1997). Animated Pedagogical Agents and Problem-Solving Effectiveness: A Large-Scale Empirical Evaluation. Proceedings of the Eighth World Conference on Artificial Intelligence in Education, Kobe, Japan.

McDonald, J.E. and Schvaneveldt, R.W. (1988). The application of user knowledge to interface design. In R. Guindon (Ed.), Cognitive science and its applications for human-computer interaction (pp.289-338). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Modrick, J.A. (1991). A conceptual framework for classifying and measuring cognitive skills. Proceedings of the Human Factors Society 35th Annual Meeting, 1288-1292.

Norman, D.A., and Rumelhart, D.E. (1975). Explorations in cognition. New York, NY: W.H. Freeman.

PCKNOT 4.3, developed by Interlink, Inc., P.O. Box 4086 UPB, Las Cruces, NM 88003

Roske-Hofstrand, R.J., and Paap, K.R. (1990). Discriminating between degrees of low or high similarity: Implications for scaling methods using semantic judgments. In R. Schvaneveldt (Ed.), Pathfinder Associative Networks Studies in Knowledge Organization. (pp.61-73). Norwood, NJ: ABLEX Publishing Corporation.

Rouse, W.R., and Morris, N.M. (1986). On looking into the black box: Prospects and limits in the search for mental models. Psychological Bulletin, 100, 349-363.

Rowe, A.L., Cooke, N.J., Hall, E.P., Halgren, T.L. (1996). Toward an on-line knowledge assessment methodology: building on the relationship between knowing and doing. Journal of Experimental Psychology: Applied, 2, 31-47.

Schvaneveldt, R.W. (Ed). (1990). Pathfinder associative networks: Studies in knowledge organization. Norwood, NJ: ABLEX Publishing Corporation.

Schvaneveldt, R., Durso, F., Goldsmith, T., Breen, T., and Cooke, N. (1985). Measuring the structure of expertise. International Journal of Man-Machine Studies, 23, 699-728.

Shachter, R.D. and Ross, D.E. (1987). Thinking backward for knowledge acquisition. AI-Magazine, 8(3), 55-61.

Shadbolt, N., O'Hara, K., and Crow, L. (1999). The experimental evaluation of knowledge acquisition methods and methods: history, problems, and new directions. International Journal of Human-Computer Studies, 51, 729-755.

Vortac, O.U., Edwards, M.B., and Manning, C.A. (1994). Sequences of actions for individual and teams of air traffic controllers. Human-Computer Interaction, 9, 319-343.

Welbank, M. (1990). An overview of knowledge acquisition methods. Interacting with Computers 2 (1), 83-91.

Wilt, G.A. (1998). A comparison of knowledge elicitation techniques for describing conceptual knowledge in declarative and procedural domains. Unpublished doctoral dissertation, George Mason University, Fairfax, VA.

Wilson, J.R., and Rutherford, A. (1989). Mental models: Theory and application in human factors. Human Factors, 31, 617-634.

Young, R. M. (1983). Surrogates and mappings: Two kinds of conceptual models for interactive devices. In D. Gentner and A.L. Stevens (Eds.), *Mental Models* (pp.35-52), Hillsdale, NJ: Erlbaum.

APPENDIX A: DEMOGRAPHICS QUESTIONNAIRE

Demographics Questionnaire for CSC 114 Students

Part 1: Please tell me about yourself.

1. What is your gender? Check only one answer.

    - ☐ Male
    - ☐ Female

2. In which college are you enrolled? Check only one answer.

Agriculture & Life Sciences	The Graduate School
Design	Humanities & Social Sciences
Education & Psychology	Management
Engineering	Physical & Mathematical Sciences
First Year College	Textiles
Forest Resources	Veterinary Medicine

3. In which academic department are you enrolled? Do not abbreviate your response.

    ____

4. What is your classification? Check only one answer.

Undergraduate student	Lifelong education undergraduate student
Graduate student	Lifelong education graduate student

5. Throughout the semester, I will need to contact you to inform you of the dates of the study. What is your e-mail address?

    ____

Part 2: Please tell me about your computer programming experience.

6. Is this the first time that you have enrolled in CSC 114? Check only one answer.

    - ☐ Yes, this is the first time that I have enrolled in CSC 114
    - ☐ No, I was previously enrolled in CSC 114

7. Have you taken any other college-level computer programming classes, either at NCSU or another college or technical institute? Check only one answer.

 Yes, I have taken other college-level computer programming classes
If yes, how many classes have you taken? _________________

 No, this is my first college-level computer programming class

8. Are you currently employed as either a computer programmer or a software engineer? Check only one answer.

 Yes, I am currently employed as either a computer programmer or software engineer
 No, I am not currently employed as either a computer programmer or software engineer

APPENDIX B: PRE-TEST

Pre-test for CSC 114 Students

Part 1: True or False, Answer T or F

-------- 1. Given an array defined as: **int myArray[50];**
The highest element is **myArray[50];**

-------- 2. It is legal to have two functions with the same name as long as their argument lists are different.

-------- 3. The following code will cause 6 to be displayed:

```
int value;
value = strlen("hello");
cout << value;
```

-------- 4. In C++ the beginning subscript of an array can be set to any value desired.

-------- 5. Arguments passed by value can affect the actual arguments in the calling function.

-------- 6. A program with the following include directive can define an output stream

```
#include <fstream.h>
```

-------- 7. **double (11/2)** evaluates to **5.5**

-------- 8. **strlen(" ");**

-------- 9. Suppose you have defined an instance of a class and have named it **myClass**, this class has a function with the following prototype:

```
void myFunct (int x);
```

Then the following is a valid call to this function:

```
int num=0;
myClass.myFunct(num);
```

-------- 10. The following statement will open a file correctly on disk:

```
istream input.create("input.data");
```

Part 2: Short Answer/Code Tracing

1. Assume that the following declarations have been made:

```
int size=8;       //# of items currently stored in the list
int list [size];  //A list of numbers
int i;            //Array element variable
```

Assume that the items given below are stored in **list [0]** through **list [7]**

If the list stored in the array list is:

3 6 8 6 1 5 2 1

What is the list after the following code is executed?

```
for (i=0;  i < size – 2;  i++)
        list [i] = list [i + 2];
```

____

2. What are the values of **myAge**, **a** and **b** after this code is executed?

```
int myAge,  a,  b;
myAge = 39;
a = myAge++;
b = ++myAge;
```

myAge________ **a**________ **b**________

3a. Define a **struct** called **myStruct**, that has 2 member data objects: an **int** called **age**, and a C style string called **name** that can contain up to 20 printable characters in the name.

3b. Using the **myStruct** type you just created in 3a, write a statement to create a variable of type **myStruct** called me

3c. Write a statement or statements to change the contents of **me** so that the **age** is equal to 90 and the **name** is equal to **George Washington**

APPENDIX C: EASE OF USE QUESTIONNAIRE

Ease of Use Questionnaire for CSC 114 Students
Similarity Ratings Task

1. How clearly stated were the instructions on how to do the similarity ratings task? Circle your answer.

a. Extremely clearly
b. Somewhat clearly
c. Neither clearly nor unclearly
d. Somewhat unclearly
e. Extremely unclearly

2. How easy was it to rate the similarity of the computer programming procedures? Circle your answer.

a. Extremely easy
b. Somewhat easy
c. Neither easy nor difficult
d. Somewhat difficult
e. Extremely difficult

3. How satisfied are you with the amount of time spent performing the similarity ratings task? Circle your answer.

a. Extremely satisfied
b. Somewhat satisfied
c. Neither satisfied nor dissatisfied
d. Somewhat dissatisfied
e. Extremely dissatisfied

4. How helpful was the similarity ratings task in terms of helping you understand the course material? Circle your answer.

a. Extremely helpful
b. Somewhat helpful
c. Neither helpful nor hindering
d. Somewhat hindering
e. Extremely hindering

5. How well was the similarity ratings task incorporated into the course? Circle your answer.

a. Extremely well
b. Somewhat well
c. Neither well nor poorly
d. Somewhat poorly
e. Extremely poorly

Ease of Use Questionnaire for CSC 114 Students
Backward Thinking Task

1. How clearly stated were the instructions on how to do the backward thinking task? Circle your answer.

a. Extremely clearly
b. Somewhat clearly
c. Neither clearly nor unclearly
d. Somewhat unclearly
e. Extremely unclearly

2. How easy was it to perform the backward thinking task? Circle your answer.

a. Extremely easy
b. Somewhat easy
c. Neither easy nor difficult
d. Somewhat difficult
e. Extremely difficult

3. How satisfied are you with the amount of time spent performing the backward thinking task? Circle your answer.

a. Extremely satisfied
b. Somewhat satisfied
c. Neither satisfied nor dissatisfied
d. Somewhat dissatisfied
e. Extremely dissatisfied

4. How helpful was the backward thinking task in terms of helping you understand the course material? Circle your answer.

a. Extremely helpful
b. Somewhat helpful
c. Neither helpful nor hindering
d. Somewhat hindering
e. Extremely hindering

5. How well was the backward thinking task incorporated into the course? Circle your answer.

a. Extremely well
b. Somewhat well
c. Neither well nor poorly
d. Somewhat poorly
e. Extremely poorly

APPENDIX D: CONSENT FORM

North Carolina State University
INFORMED CONSENT FORM

How Students Learn Programming

Susan Kahler Dr. Sharolyn Converse

You are invited to participate in a research study. The purpose of this study is to investigate how students learn programming over the course of a semester using two different techniques to measure learning.

INFORMATION

You will be asked to sign a consent form to participate in the study. You will complete a demographics questionnaire, which asks you about your computer programming experience. You will complete a test to assess your knowledge of computer programming. Three times during the semester, you will be asked to either rate the relationships between programming procedures or you will be asked to write down the programming steps that you went through to complete your projects. At the end of the semester, you will be asked to complete the ease of use questionnaire to provide your opinions on the how easy or difficult the ratings or written task was to perform. At the conclusion of the semester, you will be told the purpose of the study and given an opportunity to ask any questions you may have had about the study. The amount of time required to participate is 4 hours over the course of the semester. The study will run from 8/21/00 to 12/15/00.

RISKS

There are no risks associated with participating in this study. Your identity will remain anonymous. **No individual data will be shared with either the course instructor, grader, or teaching assistants.**

BENEFITS

Your feedback will help us to develop methods to better understand how students learn programming procedures over the course of a semester.

CONFIDENTIALITY

The information in the study records will be kept strictly confidential. Data will be stored securely and will be made available only to persons conducting the study unless you specifically give permission in writing to do otherwise. No reference will be made in oral or written reports that could link you to the study.

COMPENSATION

You will receive 2 points extra credit to be applied to your course average for the semester. For example, if you have an average of 89 (B+) in the class, then you will receive 2 points and your average will change to 91 (A-).

CONTACT

If you have questions at any time about the study or the procedures, you may contact the researcher, Susan Kahler**, at** Poe Hall, Box 7801, NCSU Campus**, or** [254-0316]**. If you feel you have not been treated according to the descriptions in this form, or your rights as a participant in research have been violated during the course of this project, you may contact Dr. Gary A. Mirka, Chair of the NCSU IRB for the Use of Human Subjects in Research Committee, Box 7906, NCSU Campus.**

PARTICIPATION

Your participation in this study is voluntary; you may decline to participate without penalty. If you

decide to participate, you may withdraw from the study at any time without penalty and without loss of benefits to which you are otherwise entitled. If you withdraw from the study before data collection is completed your data will be returned to you or destroyed.

CONSENT
I have read and understand the above information. I have received a copy of this form. I agree to participate in this study. I also agree to release my final exam grade to the researcher with the understanding that my identity will be kept confidential.

Subject's signature_______________________________________ Date _________________

(If you are under the age of 18, you will need to obtain your parent' s signature to participate in this research study.)

Parent's signature _______________________________________ Date _________________

Investigator's signature_________________________________ Date _________________

APPENDIX E: PARTICIPANT INSTRUCTIONS

HOW STUDENTS LEARN PROGRAMMING

RESEARCHER'S HANDBOOK

September 2000

Susan Kahler

TABLE OF CONTENTS

IMPORTANT PHONE NUMBERS

Name	Role	Phone Number
	Primary researcher	H: 233-6629, W:254-0316 CEL: 612-7225
Shari Converse	Faculty sponsor	515-1722
Michael Chevalier	Assistant researcher	H: 676-1832
Ginny Jones	Assistant researcher	H: 858-5140, CEL: 961-2627

Participant Task	Data Collection Dates
	August 28 to September 1
Knowledge Acquisition task for Project 1 (week before)	September 18-22
Project 1 is due	September 24
Knowledge Acquisition task for Project 1 (week after)	September 25-29
Knowledge Acquisition task for Project 2 (week before)	October 16-20
Project 2 is due	October 22
Knowledge Acquisition task for Project 2 (week after)	October 23-27
Knowledge Acquisition task for Project 3 (week before)	November 13-17
Project 3 is due	November 19
Knowledge Acquisition task for Project 3 (week after)	November 20, 21,27,28,29
Post-test and EOU	December 4-8

INSTRUCTIONS FOR PRE-TEST DATA COLLECTION SESSION

[Read aloud to student]

The purpose of the pre-test is to get a baseline measure of your C++ computer programming knowledge. The pre-test contains ten “True” or “False” questions and three short answer questions. The test is designed to be completed in one hour or less.

Your pre-test score will not be shared with either the course instructor, grader, or teaching assistants. Your score on the pre-test is for the purposes of the research study only and does not affect your course grade.

Do you have any questions?

[Hand student the pre-test and check name off list in spreadsheet printout]

[When student has completed the pre-test, return his or her consent form]

PROJECT 1

DENTS IN BACKWARD THINKING TASK, CONDITION 1

Subject Number ____________ **Date** ____________ **Time** ____________

Preparation for Condition 1

Arrive at the classroom at least 15 minutes before the data collection session starts. The **spreadsheet** shows the student's subject number, and the condition that he or she is in. When the student walks in the room, ask for his or her name. Check the appropriate activity by the name to indicate that he or she has shown up for the activity. We have to track this so students get their extra credit for participation.

Pull the instructions and diskettes for each participant attending your session. Place them in the folder by the workstation.

Introduction to Condition 1

Hi (*student's name* . My name is __________. Thanks a lot for coming to help us out.

For the purposes of the research study, I need to know if you have started project one. This information will not be shared with the course instructor. Have you started project one?

[Circle answer].

- Yes, What percentage of the project have you completed? _________ % completed
- No

[Turn Page]

Example of a Backward Thinking Task

1. Click on the shortcut to “Backward Thinking for Project One”.
2. Click on the file for that subject’s number ___________

The task that you are going to perform today is backward thinking. In backward thinking, you describe your most recent step first and work backwards to the initial step that you did.

Let’s look at the example on the screen.

For example, if you were asked to tell me how you got to Poe 639 today from the parking lot, you might tell me something like:

[Read all of the steps to the subject]

Most recent step: I walked through the doorway of Poe 639
I got off the elevator
I got onto the elevator
I walked into Poe
I crossed the street
I got out of the car
Initial step: I parked the car in the parking lot

Do you have any questions?

[Turn Page]

Sample Backward Thinking Task

Now let's move on to a sample backward thinking task. What are the steps that you would go through to pick up a package from the post office? Start with thanking the clerk and work backwards to the initial step of parking in the post office parking lot.

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could thank the clerk?

Do you have any questions?

Go ahead and type your steps into the file.

Let me know when you are finished.

[Turn Page]

Backward Thinking Task for Project One

Now you're ready to begin the backwards thinking task for project one. The project description is provided for your review.

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

What are the steps you would go (or did go) through to complete project 1, starting with the most recent step first and working backwards to the initial step? Start with dropping off the printout in Leazar and work backwards to the initial step of reading and understanding the project 1 specifications.

You do not need to include details which are specific to the software that you are using such as "I clicked on the mouse" or I selected "Save from the File menu".

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could drop-off the printout?

Do you have any questions?

Go ahead and type your steps into the file. Let me know when you are finished.

[Turn Page]

When the student is finished

Thank you for your time today. You will be asked to come back in October to do a similar backward thinking task. There is a sign-up sheet for you to select your time for October. We will send you out a reminder.

Data Backup Procedures

1. Save the student's file to the hard drive first by selecting Save from the File Menu in Word.
2. Save the student's file to the diskette by selecting Save as... from the File Menu in Word, then select the A drive where it says "Save in", then select the project 1 folder.
3. Put the diskette back into the diskette case.

PROJECT 1

INSTRUCTIONS FOR STUDENTS IN RATINGS TASK FOR PROJECT ONE, CONDITION 2

Subject Number ____________ **Date** ____________ **Time** ____________

Preparation for Condition 2

Arrive at the classroom at least 15 minutes before the data collection session starts. The
shows the student's subject number, and the condition that he or she is in. When the student walks in the room, ask for his or her name. Check the appropriate activity by the name to indicate that he or she has shown up for the activity. We have to track this so students get their extra credit for participation.

Pull the instructions and diskettes for each participant attending your session. Place them in the folder by the workstation.

Introduction to Condition 2

Hi (_____________). My name is __________. Thanks a lot for coming to help us out.

For the purposes of the research study, I need to know if you have started project one. This information will not be shared with the course instructor. Have you started project one? [Circle answer].

- Yes, What percentage of the project have you completed? _________ % completed
- No

[Turn Page]

Sample Ratings Task

To start the sample ratings program, click on the desktop icon “Shortcut to SampleRatings”. This will start the ratings program.

1. Go ahead and “Push to Start”

2. When prompted to “Enter your team number”, **type-in 99**

procedures, you will be asked “How related are these two terms? “ Let’s start out with a sample ratings task. In this task, you will be asked to rate the relatedness of four coursework procedures. We would like you to rate each pair according to the degree of overall relatedness between the two items in a pair. To rate the pairs, a scale of “Slightly Related” to “Highly Related” is provided. If you think that a pair of items is not at all related, then select “Not Related”. After you made a choice, select the “Next Pair” button. You may use your own criteria to judge relatedness.

Do you have any questions?

Go ahead and start. Let me know when you are finished.

[Turn Page]

Project Ratings Task

To start the project one ratings program, click on the desktop icon “Shortcut to ProjectOneRatings”. This will start the ratings program.

1. Go ahead and “Push to Start”

2. When prompted to “Enter your team number”, **type-in the subject number**

In this ratings task, there are nine programming procedures. The programming procedures are:

- program design
- functional decomposition
- calling functions
- defining variables
- getting input
- generating output
- aligning output
- performing calculations
- testing and debugging

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

When the student is finished

Thank you for your time today. You will be asked to come back in October to do a similar ratings task. There is a sign-up sheet for you to select your time for October. We will send you out a reminder.

<u>Data Backup Procedures</u>

1. The rating files are located in the shortcut "Student Data for Ratings Task".

\project1

PROJECT 1

NG TASK, CONDITION 3

Subject Number ____________ **Date** ____________ **Time** ____________

Preparation for Condition 3

spreadsheet

Introduction to Condition 3

____

-
-

Example of a Backward Thinking Task

1. Click on the shortcut to "Backward Thinking for Project One".
2. Click on the file for that **subject's number ___________**

The task that you are going to perform today is backward thinking. In backward thinking, you describe your most recent step first and work backwards to the initial step that you did.

Let's look at the example on the screen.

For example, if you were asked to tell me how you got to Poe 639 today from the parking lot, you might tell me something like:

[Read all of the steps to the subject]

Most recent step: I walked through the doorway of Poe 639
I got off the elevator
I got onto the elevator
I walked into Poe
I crossed the street
I got out of the car
Initial step: I parked the car in the parking lot

Do you have any questions?

Sample Backward Thinking Task

Now let's move on to a sample backward thinking task. What are the steps that you would go through to pick up a package from the post office? Start with thanking the clerk and work backwards to the initial step of parking in the post office parking lot.

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could thank the clerk?

Do you have any questions?

Go ahead and type your steps into the file.

Let me know when you are finished.

[Turn Page]

Backward Thinking Task for Project One

Now you're ready to begin the backwards thinking task for project one. The project description is provided for your review.

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

What are the steps that you went through to complete project 1, starting with the most recent step first and working backwards to the initial step? Start with dropping off the printout in Leazar and work backwards to the initial step of reading and understanding the project 1 specifications.

You do not need to include details which are specific to the software that you are using such as "I clicked on the mouse" or I selected "Save from the File menu".

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could drop-off the printout?

Do you have any questions?

Go ahead and type your steps into the file. Let me know when you are finished.

When the student is finished

Thank you for your time today. You will be asked to come back in October to do a similar backward thinking task. There is a sign-up sheet for you to select your time for October. We will send you out a reminder.

Data Backup Procedures

1. Save the student's file to the hard drive first by selecting Save from the File Menu in Word.
2. Save the student's file to the diskette by selecting Save as... from the File Menu in Word, then select the A drive where it says "Save in", then select the project 1 folder.
3. Put the diskette back into the diskette case.

PROJECT 1

INSTRUCTIONS FOR STUDENTS IN RATINGS TASK FOR PROJECT ONE, CONDITION 4

Subject Number ____________ Date ____________ Time ____________

Preparation for Condition 4

spreadsheet

Introduction to Condition 4

Sample Ratings Task

type-in 99

Project Ratings Task

type-in the subject number

In this ratings task, there are nine programming procedures. The programming procedures are:

- program design
- functional decomposition
- calling functions
- defining variables
- getting input
- generating output
- aligning output
- performing calculations
- testing and debugging

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

When the student is finished

Thank you for your time today. You will be asked to come back in October to do a similar ratings task. There is a sign-up sheet for you to select your time for October. We will send you out a reminder.

<u>Data Backup Procedures</u>

1. The rating files are located in the shortcut "Student Data for Ratings Task".

PROJECT 2

INSTRUCTIONS FOR STUDENTS IN BACKWARD THINKING TASK, CONDITION 1

Subject Number ____________ **Date** ____________ **Time** ____________

Preparation for Condition 1

spreadsheet

Introduction to Condition 1

____

-
-

Example of a Backward Thinking Task

1. Click on the shortcut to "Backward Thinking for Project Two".
2. Click on the file for that subject's number ___________

The task that you are going to perform today is backward thinking. You did this task last month. In backward thinking, you describe your most recent step first and work backwards to the initial step that you did.

As a refresher, let's look at the example on the screen.

For example, if you were asked to tell me how you got to Poe 639 today from the parking lot, you might tell me something like:

[Read all of the steps to the subject]

Most recent step: I walked through the doorway of Poe 639
I got off the elevator
I got onto the elevator
I walked into Poe
I crossed the street
I got out of the car
Initial step: I parked the car in the parking lot

Do you have any questions?

<u>Sample Backward Thinking Task</u>

Now let's move on to a sample backward thinking task. You've worked with this example before. What are the steps that you would go through to pick up a package from the post office? Start with thanking the clerk and work backwards to the initial step of parking in the post office parking lot.

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could thank the clerk?

Do you have any questions?

Go ahead and type your steps into the file.

Let me know when you are finished.

[Turn Page]

Backward Thinking Task for Project Two

Now you're ready to begin the backwards thinking task for project two. The project description is provided for your review.

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

What are the steps you would go (or did go) through to complete project 2, starting with the most recent step first and working backwards to the initial step? Start with dropping off the printout in Leazar and work backwards to the initial step of reading and understanding the project 2 specifications.

You do not need to include details which are specific to the software that you are using such as "I clicked on the mouse" or I selected "Save from the File menu".

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could drop-off the printout?

Do you have any questions?

Go ahead and type your steps into the file. Let me know when you are finished.

When the student is finished

Thank you for your time today. You will be asked to come back in November to do a similar backward thinking task. There is a sign-up sheet for you to select your time for November. We will send you out a reminder.

Data Backup Procedures

1. Save the student's file to the hard drive first by selecting Save from the File Menu in Word.
2. Save the student's file to the diskette by selecting Save as... from the File Menu in Word, then select the A drive where it says "Save in", then select the project 2 folder.
3. Put the diskette back into the diskette case.

PROJECT 2

INSTRUCTIONS FOR STUDENTS IN RATINGS TASK, CONDITION 2

Subject Number ____________ **Date** ____________ **Time** ____________

Preparation for Condition 2

spreadsheet

Introduction to Condition 2

____

-
-

Sample Ratings Task

type-in 99

Project Ratings Task

type-in the subject number

In this ratings task, there are fifteen programming procedures. The programming procedures are:

- program design
- functional decomposition
- calling functions
- defining variables
- getting input
- generating output
- aligning output
- performing calculations
- testing and debugging
- working with bool expressions
- using if/switch statements
- using loops
- using class libraries
- designing geometric solutions
- reading from a file

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

When the student is finished

Thank you for your time today. You will be asked to come back in November to do a similar ratings task. There is a sign-up sheet for you to select your time for November. We will send you out a reminder.

Data Backup Procedures

1. The rating files are located in the shortcut “Student Data for Ratings Task”.

PROJECT 2

INSTRUCTIONS FOR STUDENTS IN BACKWARD THINKING TASK, CONDITION 3

Subject Number ____________ Date ____________ Time ____________

Preparation for Condition 3

spreadsheet

Introduction to Condition 3

-
-

Example of a Backward Thinking Task

1. Click on the shortcut to “Backward Thinking for Project Two”.
2. Click on the file for that **subject’s number ___________**

The task that you are going to perform today is backward thinking. You did this task last month. In backward thinking, you describe your most recent step first and work backwards to the initial step that you did.

As a refresher, let’s look at the example on the screen.

For example, if you were asked to tell me how you got to Poe 639 today from the parking lot, you might tell me something like:

[Read all of the steps to the subject]

Most recent step: I walked through the doorway of Poe 639 I got off the elevator I got onto the elevator I walked into Poe I crossed the street I got out of the car **Initial step**: I parked the car in the parking lot

Do you have any questions?

Sample Backward Thinking Task

Now let's move on to a sample backward thinking task. You've worked with this example before. What are the steps that you would go through to pick up a package from the post office? Start with thanking the clerk and work backwards to the initial step of parking in the post office parking lot.

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could thank the clerk?

Do you have any questions?

Go ahead and type your steps into the file.

Let me know when you are finished.

[Turn Page]

Backward Thinking Task for Project 2

Now you're ready to begin the backwards thinking task for project 2. The project description is provided for your review.

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

What are the steps that you went through to complete project 2, starting with the most recent step first and working backwards to the initial step? Start with dropping off the printout in Leazar and work backwards to the initial step of reading and understanding the project 2 specifications.

You do not need to include details which are specific to the software that you are using such as "I clicked on the mouse" or I selected "Save from the File menu".

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could drop-off the printout?

Do you have any questions?

Go ahead and type your steps into the file. Let me know when you are finished.

When the student is finished

Thank you for your time today. You will be asked to come back in November to do a similar backward thinking task. There is a sign-up sheet for you to select your time for November. We will send you out a reminder.

Data Backup Procedures

1. Save the student's file to the hard drive first by selecting Save from the File Menu in Word.
2. Save the student's file to the diskette by selecting Save as... from the File Menu in Word, then select the A drive where it says "Save in", then select the project 2 folder.
3. Put the diskette back into the diskette case.

PROJECT 2

INSTRUCTIONS FOR STUDENTS IN RATINGS TASK, CONDITION 4

Subject Number ____________ **Date** ____________ **Time** ____________

Preparation for Condition 4

spreadsheet

Introduction to Condition 4

Sample Ratings Task

type-in 99

Project Ratings Task

type-in the subject number

In this ratings task, there are 15 programming procedures. The programming procedures are:

- program design
- functional decomposition
- calling functions
- defining variables
- getting input
- generating output
- aligning output
- performing calculations
- testing and debugging
- working with bool expressions
- using if/switch statements
- using loops
- using class libraries
- designing geometric solutions
- reading from a file

There are 105 pairs of programming procedures for you to rate. We would like you to rate each pair according to the degree of overall relatedness between the two items in a pair. To rate the pairs, a scale of "Slightly Related" to "Highly Related" is provided. If you think that a pair of items is not at all related, then select "Not Related". After you made a choice, select the "Next Pair" button. You may use your own criteria to judge relatedness. When you have finished, you will be prompted to exit the program and inform the researcher.

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

When the student is finished

Thank you for your time today. You will be asked to come back in November to do a similar ratings task. There is a sign-up sheet for you to select your time for November. We will send you out a reminder.

Data Backup Procedures

1. The rating files are located in the shortcut "Student Data for Ratings Task".

PROJECT 3

INSTRUCTIONS FOR STUDENTS IN BACKWARD THINKING TASK, CONDITION 1

Subject Number ____________ Date ____________ Time ____________

Preparation for Condition 1

spreadsheet

Introduction to Condition 1

Example of a Backward Thinking Task

1. Click on the shortcut to "Backward Thinking for Project three".
2. Click on the file for that subject's number ___________

The task that you are going to perform today is backward thinking. You did this task last month. In backward thinking, you describe your most recent step first and work backwards to the initial step that you did.

As a refresher, let's look at the example on the screen.

For example, if you were asked to tell me how you got to Poe 639 today from the parking lot, you might tell me something like:

[Read all of the steps to the subject]

Most recent step: I walked through the doorway of Poe 639
I got off the elevator
I got onto the elevator
I walked into Poe
I crossed the street
I got out of the car
Initial step: I parked the car in the parking lot

Do you have any questions?

Sample Backward Thinking Task

Now let's move on to a sample backward thinking task. You've worked with this example before. What are the steps that you would go through to pick up a package from the post office? Start with thanking the clerk and work backwards to the initial step of parking in the post office parking lot.

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could thank the clerk?

Do you have any questions?

Go ahead and type your steps into the file.

Let me know when you are finished.

[Turn Page]

Backward Thinking Task for Project three

Now you're ready to begin the backwards thinking task for project three. The project description is provided for your review.

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

What are the steps you would go (or did go) through to complete project 3, starting with the most recent step first and working backwards to the initial step? Start with dropping off the printout in Leazar and work backwards to the initial step of reading and understanding the project 3 specifications.

You do not need to include details which are specific to the software that you are using such as "I clicked on the mouse" or I selected "Save from the File menu".

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could drop-off the printout?

Do you have any questions?

Go ahead and type your steps into the file. Let me know when you are finished.

When the student is finished

Thank you for your time today. You will be asked to come back in December to take the post-test. This is similar to the test that you took in August. There is a sign-up sheet for you to select your time for December. We will send you out a reminder.

Data Backup Procedures

1. Save the student's file to the hard drive first by selecting Save from the File Menu in Word.
2. Save the student's file to the diskette by selecting Save as... from the File Menu in Word, then select the A drive where it says "Save in", then select the project 3 folder.
3. Put the diskette back into the diskette case.

PROJECT 3

INSTRUCTIONS FOR STUDENTS IN RATINGS TASK, CONDITION 2

Subject Number ____________ Date ____________ Time ____________

Preparation for Condition 2

spreadsheet

Introduction to Condition 2

____

-
-

Sample Ratings Task

type-in 99

Project Ratings Task

type-in the subject number

In this ratings task, there are 18 programming procedures. The programming procedures are:

•	program design	•	using if/switch statements
•	functional decomposition	•	using loops
•	calling functions	•	using class libraries
•	defining variables	•	reading from a file
•	getting input	•	using arrays
•	generating output	•	sorting
•	aligning output	•	searching
•	performing calculations	•	using structs
•	testing and debugging		
•	working with bool expressions		

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

When the student is finished

Thank you for your time today. You will be asked to come back in December to take the post-test. This is similar to the test that you took in August. There is a sign-up sheet for you to select your time for December. We will send you out a reminder.

Data Backup Procedures

1. The rating files are located in the shortcut “Student Data for Ratings Task”.

PROJECT 3

INSTRUCTIONS FOR STUDENTS IN BACKWARD THINKING TASK, CONDITION 3

Subject Number ____________ **Date** ____________ **Time** ____________

Preparation for Condition 3

spreadsheet

Introduction to Condition 3

____

-
-

Example of a Backward Thinking Task

1. Click on the shortcut to "Backward Thinking for Project three".
2. Click on the file for that **subject's number ___________**

The task that you are going to perform today is backward thinking. You did this task last month. In backward thinking, you describe your most recent step first and work backwards to the initial step that you did.

As a refresher, let's look at the example on the screen.

For example, if you were asked to tell me how you got to Poe 639 today from the parking lot, you might tell me something like:

[Read all of the steps to the subject]

Most recent step: I walked through the doorway of Poe 639 I got off the elevator I got onto the elevator I walked into Poe I crossed the street I got out of the car **Initial step**: I parked the car in the parking lot

Do you have any questions?

Sample Backward Thinking Task

Now let's move on to a sample backward thinking task. You've worked with this example before. What are the steps that you would go through to pick up a package from the post office? Start with thanking the clerk and work backwards to the initial step of parking in the post office parking lot.

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could thank the clerk?

Do you have any questions?

Go ahead and type your steps into the file.

Let me know when you are finished.

[Turn Page]

Backward Thinking Task for Project 3

Now you're ready to begin the backwards thinking task for project 3. The project description is provided for your review.

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

What are the steps that you went through to complete project 3, starting with the most recent step first and working backwards to the initial step? Start with dropping off the printout in Leazar and work backwards to the initial step of reading and understanding the project 3 specifications.

You do not need to include details which are specific to the software that you are using such as "I clicked on the mouse" or I selected "Save from the File menu".

[Move the cursor to the line under the most recent step]

To get you started, what had to happen before you could drop-off the printout?

Do you have any questions?

Go ahead and type your steps into the file. Let me know when you are finished.

When the student is finished

Thank you for your time today. You will be asked to come back in December to take the post-test. This is similar to the test that you took in August. There is a sign-up sheet for you to select your time for December. We will send you out a reminder.

Data Backup Procedures

1. Save the student's file to the hard drive first by selecting Save from the File Menu in Word.
2. Save the student's file to the diskette by selecting Save as... from the File Menu in Word, then select the A drive where it says "Save in", then select the project 3 folder.
3. Put the diskette back into the diskette case.

PROJECT 3

INSTRUCTIONS FOR STUDENTS IN RATINGS TASK, CONDITION 4

Subject Number ____________ Date ____________ Time ____________

Preparation for Condition 4

spreadsheet

Introduction to Condition 4

Sample Ratings Task

type-in 99

type-in the subject number

In this ratings task, there are 18 programming procedures. The programming procedures are:

• program design	• using if/switch statements
• functional decomposition	• using loops
• calling functions	• using class libraries
• defining variables	• reading from a file
• getting input	• using arrays
• generating output	• sorting
• aligning output	• searching
• performing calculations	• using structs
• testing and debugging	
• working with bool expressions	

Your answers will not be shared with either the course instructor, grader, or teaching assistants. Your answers are for the purposes of the research study only and do not affect your course grade.

When the student is finished

Thank you for your time today. You will be asked to come back in December to take the post-test. This is similar to the test that you took in August. There is a sign-up sheet for you to select your time for December. We will send you out a reminder.

Data Backup Procedures

1. The rating files are located in the shortcut "Student Data for Ratings Task".

INSTRUCTIONS FOR POST-TEST DATA COLLECTION SESSION

[Read aloud to student]

The purpose of the post-test is to get an end of the semester measure of your C++ computer programming knowledge. The post-test contains ten “True” or “False” questions and three short answer questions. The test is designed to be completed in one hour or less.

The ease of use questionnaire asks your opinion of the ease of use of the task that you performed three times on the computer over the semester. There are five questions for you to answer. Because it is your opinion, there are no right or wrong answers.

Your post-test score will not be shared with either the course instructor, grader, or teaching assistants. Your score on the post-test is for the purposes of the research study only and does not affect your course grade. Your opinions will also remain confidential.

Do you have any questions?

[Give student the post-test and the ease of use questionnaire, and check name off list in spreadsheet printout]

[When student has finished, give him or her the debriefing hand-out]

DEBRIEFING HANDOUT

"How Students Learn Programming"

Dear Student,

Thank you for your participation in my research study. Your help is very much appreciated. I will send a note to the instructor (Dulberg) letting him know that you should receive the 2 points extra credit. You have definitely earned it!

The purpose of this study is to compare 2 techniques in terms of their ability to capture your "mental model" of C++ computer programming at different times over the course of the semester. A "mental model" is your cognitive representation of a task or domain.

Half of the students rated pairs of programming concepts in terms of their relatedness. The number of programming concepts increased over the semester to match the concepts that were taught in the classroom.

The rest of the students described the steps to complete each project, but started with the last step that they took to complete the project and worked backwards to reading and understanding the project specifications. This task is called backward thinking.

Each of these techniques has their pluses and minuses, but have not been compared and contrasted in a controlled research setting. The study that you participated in is a longitudinal study because it was conducted over the course of the semester.

Just to remind you, your data is confidential and will only be reported in aggregate form. No individual names will be associated with the data.

If you have any questions, please feel free to send me an e-mail (skahler@us.ibm.com) or call me at 254-0316.

Again, thank you for your participation. You did a great job!

Sincerely,

Susan Kahler

APPENDIX F: BACKWARD THINKING PROTOCOL SCORING KEYS

Backward Thinking for Project 1 Grading Sheet

Step #	Points student has earned	Step and Point Value
10 5pts		Submit source code Circle one: -5 pts if missing -2.5 pts if out of order, must be before Dropoff printout in Leazer
09 16 pts		Test program with a variety of test cases Circle one: -16 pts if missing -8 pts if out of order, must be before “Submit Source Code”
08 12 pts		Make sure output is formatted correctly Circle one: -12 pts if missing - 6 pts if out of order, must be before “Test program with a variety of test cases”
07 5 pts		Get total & sales tax to work Circle one: -5 pts if missing -2.5 pts if out of order, must be before “Make sure output is formatted correctly”
06 12 pts		Get discounts to work correctly Circle one: -12 pts if missing - 6 pts if out of order, must be before “Get total & sales tax to work”
05 5 pts		Get undiscounted total correct Circle one: -5 pts if missing -2.5 pts if out of order, must be before “ Get discounts to work correctly”
04 5 pts		Get individual item totals correct and displayed Circle one: -5 pts if missing -2.5 pts if out of order, must be before “Get undiscounted total correct”
03 5 pts		Get user input Circle one: -5 pts if missing -2.5 pts if out of order, must be before”Get individual item totals correct and displayed”
02 5 pts		Display banner Circle one: -5 pts if missing -2.5 pts if out of order, must be before “Get user input”
01 30 pts		Come up with design for project 1, functional decomposition Circle one: -30 pts if missing -15 points if out of order, must be before “Display banner”

Score _____________ out of 100 pts

# of Steps Out of Order _____________

# of Steps in Student's file_____________
(Do not include Initial Step and Most Recent Step)

# of Steps Missing _____________

Guidelines:

1. For step 1, give the student credit if they say words like "write steps down", "plan", "thought about project", "draw", in addition to design or functional decomposition.

2. If the students says " I did the calculations" but doesn't break them into the appropriate steps (4-6) then assume that the steps are there but out of order. So, instead of receiving 22 points, they would receive 11 points.

Backward Thinking for Project 2 Grading Sheet

Step #	Points student has earned	Step and Point Value
08 5pts		Submit source code Circle one: -5 pts if missing -2.5 pts if out of order, must be before Dropoff printout in Leazer
07 15 pts		Test program with a variety of test cases Circle one: -15 pts if missing -7.5 pts if out of order, must be before "Submit source code"
06 10 pts		Get low symbol to draw correctly (or Get high symbol to draw correctly) Circle one: -10 pts if missing - 5 pts if out of order, must be before "Test program with a variety of test cases"
05 10 pts		Get high symbol to draw correctly (or Get low symbol to draw correctly) Circle one: -10 pts if missing -5 pts if out of order, must be before " Test program with a variety of test cases"
04 10 pts		Get clipping to work correctly Circle one: -10 pts if missing -5 pts if out of order, must be before "Get high or low symbol to draw correctly"
03 10 pts		Get low's to plot without clipping (or Get high's to plot without clipping) Circle one: -10 pts if missing -5 pts if out of order, must be before "Get clipping to work correctly"
02 10 pts		Get high's to plot without clipping (or Get low's to plot without clipping) Circle one: -10 pts if missing -5 pts if out of order, must be before "Get clipping to work correctly"
01 30 pts		Come up with design for project 2, functional decomposition Circle one: -30 pts if missing -15 points if out of order, must be before "Get high's or low's to plot without clipping"

Score ______________ out of 100 pts　　　　# of Steps Out of Order ______________

# of Steps in Student's file______________　　　　# of Steps Missing ______________
(Do not include Initial Step and Most Recent Step)

Guidelines:

1. For step 1, give the student credit if they say words like “write steps down”, “plan”, “thought about project”, “draw”, in addition to design or functional decomposition.

2. If the student mentions that they used the sample in the project description, they are doing steps 2 and 3.

Backward Thinking for Project 3 Grading Sheet

Step #	Points	Step and Point Value
10 5 pts		Submit source code Circle one: -5 pts if missing -2.5 pts if out of order, must be before Dropoff printout in Leazer
09 16 pts		Test program with a variety of test cases Circle one: -16 pts if missing - 8 pts if out of order, must be before “Submit source code”
08 7 pts		Get menu (looping) to work correctly Circle one: - 7 pts if missing - 3.5 pts if out of order, must be before “Test program with a variety of test cases”
07 7 pts		Get bank and betting to work correctly Circle one: - 7 pts if missing - 3.5 pts if out of order, must be before “Get menu (looping) to work correctly”
06 7 pts		GET WINNING HANDS TO DETERMINE WINNER Circle one: - 7 pts if missing - 3.5 pts if out of order, must be before “Get bank and betting to work correctly”
05 7 pts		Get hand to display correctly or Get point totals to work correctly or Get hand to sort correctly Circle one: - 7 pts if missing -3.5 pts if out of order, must be before “Get winning hands to determine winner”
04 7 pts		Get hand to display correctly or Get point totals to work correctly or Get hand to sort correctly Circle one: -7 pts if missing -3.5 pts if out of order, must be before “Get winning hands to determine winner”
03 7 pts		Get hand to display correctly or Get point totals to work correctly or Get hand to sort correctly Circle one: -7 pts if missing -3.5 pts if out of order, must be before ”Get winning hands to determine winner”
02 7 pts		Get Class to work correctly (shuffling and dealing cards) Circle one: -7 pts if missing -3.5 pts if out of order, must be before “Get hand to display correctly”

01 30 pts		Come up with design for project 3, functional decomposition Circle one: -30 pts if missing -15 points if out of order, must be before “Get class to work correctly (shuffling and dealing cards)”

Score ____________ out of 100 pts   # of Steps Out of Order ____________

# of Steps in Student’s file____________   # of Steps Missing ____________
(Do not include Initial Step and Most Recent Step)

Guidelines:

1. For step 1, give the student credit if they say words like “write steps down”, “plan”, “thought about project”, “draw”, in addition to design or functional decomposition.

2. Steps 3,4,5 can be in any order.

3. The menu code refers to asking for another game.

4. Steps 2, 3-5 were done in homework #7 which is due Sunday 11/19. This is the first time that parts of the project were due as part of a homework assignment. The project due date is 11/28.