to support their decisions with recourse to both, the problem is that these methods sometimes do not yield the same predictions. If a clinician prefers to follow one method generally and only employ the other on “special occasions”, the problem then becomes one of how to define this subset of occasions.

An analogous situation is emerging in the software engineering community with regard to the growing numbers of software metrics aimed at quantifying attributes of software products or processes. The human users of formal methods have the whole of their life experiences, specialist training and personal beliefs to draw upon. Although the model’s predictions are based on a large number of samples, these samples are of a relatively restricted type. The approach advocated by Meehl, in the case of clinical diagnoses, is to use one’s head in general, especially in those cases where statistical methods are inappropriate or yield predictions which are clearly incorrect. It is likely, after all, that there will arise many occasions for which a suitable statistical method is unavailable and the decision maker has no alternative. For the subset of occasions on which our regression based model is applicable, however, it should be remembered that there is a great deal of evidence to suggest that regression based models yield more accurate predictions than those given by human judges, including those judges with relevant training (Dawes, 1971; 1979; Slovic and Lichtenstein, 1971).

Providing the model proposed in this thesis consistently yields more accurate predictions than those based on naive human intuition, it should have scope for practical application. There is reason to expect that this claim might gain empirical support because participants’ extremely high confidence ratings and low levels of correctness suggest their intuitions were often fallible. Further research is required, however, before this claim can be made with any confidence. Perhaps a useful starting point for such research would be those attempts to represent using mathematical formulae the subset of occasions on which linear regression models
tend to outperform human judges (see for example: Goldberg, 1970).

9.3.8 Training Considerations

Staff training is often a key concern for organisations contemplating the adoption of new technology. It is perhaps this concern which has led proponents to claim that the training requirements for formal methods are trivial, and that prospective users quickly learn to appreciate the notations and underlying concepts which facilitate their application. It is claimed, for example, that a background in set theory and logic are the only necessary prerequisites for formal methods and that the mathematics required for formal specification is “easy” (Hall, 1990) or “straightforward” (Thomas, 1993). It is also claimed that prospective users will feel comfortable with the mathematics and symbology underlying formal methods, and be reading and writing specifications within a matter of days (Larsen et al., 1996; Potter, 1991). It is perhaps the realisation that the mathematics is not accessible to everyone, however, which leads Hall (1990, p.17) to add the proviso that “competent people who can cope with the necessary mathematical manipulations are the ones who must carry out safety critical projects”.

It was participants’ expertise, rather than linguistic factors, which accounted for the greatest variance in our experimental data and, hence, exerted a dominating influence on reasoning performance. It might be argued, therefore, that prospective users should receive extensive practical training in reasoning about formal specifications over a prolonged period before formal methods are applied on critical industrial projects. Academic courses tend to emphasize the teaching of grammatical issues in order that students may read and write specifications within a short period of time. The motivation for such courses may stem from undergraduate textbooks, which tend to focus on the grammatical symbology of specific notations rather than the deeper conceptual issues relating to the use of formal methods (Garlan, 1996). If
students are not encouraged to reason about specifications in academia, however, it should hardly be surprising that they will succumb to error when reasoning about formal specifications in industry. Students do not learn to guard against unwarranted inferences and are not encouraged to justify warranted ones.

Users clearly need to develop an understanding of formal grammar so that they are capable of reading and writing formal specifications. Based on the findings of this research, it might be argued that an adequate training programme would also provide practical experience of reasoning about specifications and alert users to linguistic constructs and conditions which typically evoke reasoning errors. The findings also suggest that formal methods training should deliver an appreciation of the various ways in which designs can be expressed, and how each of these might affect an audience's interpretative or reasoning processes. It would require a radical rethink of training culture and policies in some organisations, however, before an educational programme can be formulated to meet all of these requirements. Rather than viewing training as an exercise to be undergone by prospective users a few weeks before formal methods are due to be applied on a project, the correlations between expertise and performance reported in this research suggest that formal methods training should be viewed as a long term endeavour. It may take months, or even years, before users learn to identify those factors which are particularly likely to evoke non-logical reasoning, develop compensatory mental heuristics, and learn to favour these heuristics habitually in software engineering contexts.

"The only complete safeguard against reasoning ill, is the habit of reasoning well; familiarity with the principles of correct reasoning, and practice in applying those principles" (Mill, 1874/1986, p.513).

Politzer (1986; 1990) attributes errors observed in studies of human reasoning to people's inability to differentiate between the laws of logic and the incompati-
ible pragmatic laws of everyday reasoning. He suggests that simple instructions or training procedures which alert reasoners to differences between the two systems can facilitate performance. This view is consistent with Cohen's (1986) argument that, providing participants are trained to watch out for certain language features they will not persist in applying inappropriate reasoning strategies and, hence, will not succumb to error quite so often. It would be interesting to test whether such forms of training can reduce the human potential for reasoning errors in formalised contexts. Our results suggest that the tendency to favour non-logical everyday heuristics can, in many cases, be reduced through increased formal language experience and expertise. It may be through increased familiarity with a formal notation, therefore, that users learn to appreciate those points at which the laws of language and logic diverge, or learn to identify those combinations of language construct which lead to reasoning errors. It seems worthy of note that natural language based studies generally ascribe reasoning errors to the linguistic variables under analysis, such as inference type or term polarity, rather than the expertise of human reasoners.

It is argued that pragmatic knowledge develops alongside language acquisition skills from early adolescence onwards, whereas knowledge of logic does not fully develop until late adolescence or early adulthood (Inhelder and Piaget, 1958). It is also argued that it is only through environmental and educational experience that people learn to appreciate fully the logical meanings of the predicate quantifiers (Politzer, 1990) and propositional connectives (Neimark and Chapman, 1975). It is only through experience and training, therefore, that an individual can distinguish between those occasions in which a purely logical or an informal pragmatic approach is appropriate. The development of such knowledge appears to be a necessary prerequisite for deductive tasks of the kind administered in this research. This hypothesis is consistent with cognitive theories of bilingualism. It is argued, for example, that an individual has only a single psycholinguistic system in which knowledge of native and
secondary languages reside, and that neither language may be completely blocked or disabled in situations where both can be applied (Dalrymple-Alford, 1968). It is also claimed that it is only through increased language familiarity that an individual learns to ignore cues which evoke the inappropriate language and begins to reason without “interlingual interference” (Kiyak, 1982).

Unfortunately, the high propensity for error which our participants exhibited under certain experimental conditions is unlikely to fix itself. Software developers will continue to be biased by prior beliefs, to endorse illicit conversions, to misinterpret logical premisses, and to be biased towards negative or determinate conclusions. Although our model accounts for a very limited range of user characteristics in its present state, its application has the potential for identifying types of user likely to be influenced by non-logical heuristics and biases (as demonstrated in Chapter Eight). The model might therefore be used to justify staff selection or training decisions, particularly where it consistently predicts high probabilities of error for staff with a particular level of expertise on critical projects. Although it is unlikely that we will ever be able to guarantee completely error-free human reasoning, cognitive science can help us to identify the conditions under which developers are prone to err, to identify the types of error likely to be committed, to formulate measures for estimating their propensity for error under these conditions, and to develop corrective procedures for any errors that might occur. We must be prepared to take on board other relevant findings from cognitive science if we are to lessen the potential for human error in software development contexts and offer empirical evidence in support or refutation of the software community’s psychological claims.

9.3.9 Further Implications for Cognitive Science

A central tenet of traditional mental logic theory is that people are equipped with inference rules analogous to those used in formal logic and would, under ideal cir-
cumstances, always apply the appropriate rules to allow a logical conclusion to be reached (Braine, 1978; Inhelder and Piaget, 1958; Rips, 1994). It seems surprising that participants strayed from fundamental rules of logic quite so often during our studies in view of, first, their logical training, second, the fact that the materials were expressed explicitly in symbolic logical terms, third, the fact that the correct solutions were presented in the form of multiple-choice options and, fourth, the fact that reasoners were allowed to complete the tasks without the time pressures often associated with laboratory based tests. Given that our participants appeared unable to reason in a truth-functional manner about explicitly logical problems which clearly called for logical lines of thought - that is, in conditions as close to "ideal" as one could reasonably expect - it is difficult to see how the results of this research could support traditional mental logic theory. The theory has also been criticised in the cognitive science literature (Evans, 1993b; Johnson-Laird and Byrne, 1991; Manktelow and Over, 1990), and in the philosophy literature (Ayer, 1971; Cohen, 1944; Strawson, 1966). Such findings appear to cast doubt on traditional mental logic theory. It is in view of these findings that we reiterate Kant.

"Logic does not really contain the rules in accordance with which man actually thinks but the rules for how man ought to think. For man often uses his understanding and thinks otherwise than he ought to think and use his understanding. Logic thus contains the objective laws of the understanding and of reason" (Kant, in Young, 1992, p.13).

As a means of defence against these criticisms, mental logic theory has been reformulated to account for the non-logical forms of inference that people make on a frequent basis in everyday life and under experimental conditions. Contemporary mental logic theory argues that people's inbuilt logical rules coexist closely with procedures for drawing non-logical inferences (Braine, 1994; Braine and O'Brien, 1991).
It argues that reasoning is guided by logical principles when the demands of the task are within the scope of people's basic mental logic skills, and is otherwise guided by pragmatic heuristics which could be oriented towards the semantic content of a task rather than its structural form (O'Brien, 1993; Rips, 1989). O'Brien (1995) claims that reasoners only resort to pragmatic heuristics when they are uncertain of how logical analysis can lead to a plausible conclusion. In response to any criticism that human reasoning is primarily dependent on the content of problems rather than their logical structure, contemporary mental logic might point to a wealth of empirical data which suggests that cognitive processes usually focus on argumentative form rather than semantic content but that content sometimes conflicts with prior beliefs, making it more difficult for reasoners to distinguish between conclusions that are logically valid and those merely believed to be valid. Although the results of this research could not fully corroborate a logico-pragmatic theory of this kind, it seems worthy of note that our findings are generally consistent with O'Brien's predictions. The case for contemporary mental logic is further supported by the fact that people appear to commit non-logical errors under highly specialised circumstances, yet are profoundly logical at other times in laboratory based studies and on a frequent basis in everyday life; a fact which is often obscured in some experimental results.

Independent strands of cognitive research have, during the past three decades, proposed at least five different approaches to theories of human reasoning: mental models (Johnson-Laird and Byrne, 1993), mental logic (Braine, 1994), heuristics and biases (Kahneman et al., 1991), domain sensitive schemas (Cheng and Holyoak, 1985), and pragmatics (Levinson, 1983). Although members of the cognitive science community have generally subscribed to one or another of these theories, few appear willing to entertain the hypothesis that an integrated model might characterise the tremendous diversity in human reasoning processes more accurately.
"There is a strong sense in which thinking research has failed to capture the dynamic qualities of everyday thought. We think that this is a result of the often-lamented, fragmentary nature of the field. As researchers have tried to come to grips with the phenomena of thinking. They have carved them into bite-sized chunks and undone a whollistic conception of a set of processes working together” (Eysenck and Keane, 1990, p.461).

That so few appear willing to subscribe to an integrated theory of the kind suggested by Eysenck and Keane may be attributable to the difficulties involved in testing such theories using existing empirical procedures. The problem is exemplified in the claim made by Evans et al. (1993) that an integrated "logico-pragmatic" model of the type commended by O'Brien (1993; 1995) would be too general and unparsimonious to be tested empirically. Whilst the recent pragmatic extensions to traditional mental logic theory might make it less amenable to empirical validation, however, the current limits of our research methods cannot undermine the possibility that an integrated model of this nature might, in fact, reflect human reasoning processes more reliably than existing fragmentary theories. Given that no existing theory appears sufficient to account for the tremendous diversity exhibited in laboratory based manifestations of human reasoning, it seems far more likely that human reasoning is guided by both logical and non-logical processes, and processes oriented towards both the syntax and semantics of reasoning problems; not simply one or the other in isolation.

"In everyday life we use a rich mix of deductive and inductive reasoning and problem-solving strategies. The one shades into the other. Therefore, the big question we need to answer is as follows: Is a unified theory of thinking possible?” (Eysenck and Keane, 1990, p.461).
Many of the errors observed during this research are consistent with the hypothesis that participants analysed the formal operators at only a syntactic level and assumed an informal semantics cued by the realistic nature of the material, similar to that used for the equivalent logical operators in everyday language. This may explain, at least in part, why the thematic groups were outperformed by the abstract groups during our studies of disjunctive, conjunctive and quantified reasoning. Although this finding would count against traditional mental logic theory, it is supported by its contemporary revisions which claim that people’s mental repertoires of logical rules coexist closely with procedures that can reach conclusions beyond those sanctionable by logic alone. Providing this argument is correct, cognitive science must begin to devise empirical methods for testing integrated theories if it is to provide a comprehensive account of human reasoning processes.

9.4 Summary

Based on cognitive theories devised to explain errors committed by reasoners without logical training in natural language based contexts, our empirical studies suggest that the users of formal methods are prone to make the same mistakes. The results suggest that a reasoner’s ability to distinguish between the concepts of logical necessity and plausible contingency in formalised contexts can be influenced significantly by linguistic properties of a specification, such as the degree of meaningful content or the type of inference to be drawn, and by psychological characteristics of the reasoner, such as their length of experience or degree of expertise. Given that the results suggest non-logical encoding, processing and response biases, the psychological causes of software developers’ reasoning errors would appear to be deep-rooted. Software development has always been driven by human reasoning and it is likely to remain so, at least in the foreseeable future. The potential for human error will
therefore persist, despite the use of formal methods, and the software development process will remain vulnerable to the fallibility of human judgement.

Although everyday reasoning heuristics often encourage people to venture beyond explicit information to reach correct decisions, application of the same heuristics under strictly logical conditions appears to elicit "flawed judgement". This is ironic because it is precisely this ability to solve problems by looking beyond given information which makes people more intelligent than computers (Funder, 1987). Cognitive studies often express logical tasks in natural language guises with everyday content and contexts. Given that pragmatic lines of thought appear almost to be invited by the experimenters in such circumstances, it may be unfair to expect reasoners to adopt purely logical lines of thought and unfair to use logic as the normative system against which reasoning is assessed. It could also be argued that the logical terms, such as "or" and "some", which occur in these studies are ambiguous in the sense that reasoners are not told whether to interpret these terms according to their pragmatic or logical meanings. One might expect that the expression of the same terms in formal logic, "\( \lor \)" and "\( \exists \)", would eliminate any such ambiguity and lead to fewer reasoning errors. The results of this research suggest that the ambiguities cannot be eliminated in an absolute sense; the users of formal methods are, under certain circumstances, liable to reason about formal expressions in the same ways as their natural language counterparts.

Given that the rates of error in our formal logic based studies are generally lower than those observed for logically equivalent tasks in natural language based studies, there is some support for the claim that formalisation leads to improved reasoning. It is disconcerting, nevertheless, to think that the users of formal methods will exhibit similar and, in some situations, increased potentials for error in critical industrial contexts, where solutions are rarely offered explicitly in the form of multiple-choice options, where formal expressions typically contain more com-
plex combinations of logical operators, and where the repercussions of erroneous reasoning can be much more severe than in laboratory based experiments. Some of the specifications given to participants in this research were for supposedly safety critical systems. Although the application of formal methods might improve overall confidence in the integrity of these systems (Wing, 1990), the results of this research suggest that even highly trained users are prone to systematic errors when reasoning about the specifications for these systems. Perhaps the kind of question that the software community should therefore be asking itself is: Can we afford to risk even one developer failing to reason correctly about a critical system specification, let alone the rates observed in this research? Reasoning errors which escape human detection in this manner, even after careful deliberation, should be of particular concern because they are also likely to be committed by the human checkers involved in manually reviewing or certifying critical systems.

This research has explored the use of an empirical approach for testing the psychological claims relating to software engineering technologies. Based on the results from a series of experiments, we have formulated a predictive model for quantifying how far expressions from formal specifications are liable to evoke human reasoning errors and biases. If we now focus attention on those areas which our model suggests are error-prone and explore alternative design representations which avoid these constructs, this may help to reduce the potential for human error in the software development process. In order to achieve our research aims we have borrowed relevant empirical knowledge and procedures from cognitive science. This has helped us to identify specific conditions under which users are liable to error and bias when reasoning about formal specifications containing negatives, conditionals, disjunctives, conjunctives and quantifiers. In so doing, the feasibility of a cognitive approach to evaluating formal specifications has been demonstrated, which is at least as important as the results themselves.
References


Appendix A

Task Sheets
Experiment 1: An Initial Investigation

Computer Science Research Experiment:
Reasoning About Formal Software Specifications.

Student/Staff/Other (please specify): 

Z experience: ... years ... months

Age: 

Z courses attended:

Course: 

Please complete all tasks to the best of your ability (without reference to textbooks).

The experiment should take no longer than 30 minutes to complete.

If you guess the answer to a task please indicate where you do so.

Task 1

The requirements for a software operation 'InOut' are as follows:

"If the operation receives an 'A' as input then it will output a '4'."

The following Z schema is the operation's formal specification.

\[
\text{InOut} \\
\text{in? : Letter} \\
\text{out! : N} \\
\text{in? = A \Rightarrow out! = 4}
\]

Which inputs and outputs would help you to test whether 'InOut' is working correctly? Please circle your choice(s).

\[
\begin{align*}
in? = A & \quad \text{(A)} \\
\text{out! = 4} & \quad \text{(B)} \\
in? = S & \quad \text{(C)} \\
\text{out! = 7} & \quad \text{(D)}
\end{align*}
\]
Task 2

Part A

The following is an abstract state schema for a computerised library system.

\[ \text{[Copy, Book, Reader]} \]

\[ \text{maxloans : N} \]

---

**Library**

- stock : Copy \( \rightarrow \) Book
- issued : Copy \( \rightarrow \) Reader
- shelved : F Copy
- readers : F Reader

\[ \text{shelved} \cup \text{dom issued} = \text{dom stock} \]
\[ \text{shelved} \cap \text{dom issued} = \emptyset \]
\[ \text{ran issued} \subseteq \text{readers} \]
\[ \neg \exists r : \text{readers} \cdot \neg \left( \#(\text{issued} \uparrow \{ r \}) > \text{maxloans} \right) \]

---

Translate the predicate part of this schema to natural English.
Part B

The following is an English description of a required software operation.

"Operation ‘Compute Value’ outputs the sum of its two inputs squared."

Translate this description to an appropriate form in Z.

\[ \text{Compute Value} \]
Task 3

The following is an English description of a required software operation.

"The operation 'Toggle' exchanges the current status of a switch."

In your opinion, which of the following Z schemas best describes the operation’s behaviour? (Please circle the letter of your choice).

\[
SWITCH ::= \text{on} | \text{off}
\]

\[
\begin{array}{ll}
\text{Toggle} & \text{Toggle} \\
\text{s, s'} : SWITCH & \text{s, s'} : SWITCH \\
\text{s'} \neq \text{s} & \begin{align*}
(s = \text{off} \land s' = \text{on}) \lor \\
(s = \text{on} \land s' = \text{off})
\end{align*} \\
\end{array}
\]

(A)  

\[
\begin{array}{ll}
\text{Toggle} & \text{Toggle} \\
\text{s, s'} : SWITCH & \text{s, s'} : SWITCH \\
\text{s = on} \Rightarrow s' = \text{off} & (s = \text{on} \lor s = \text{off}) \Rightarrow \\
\text{s = off} \Rightarrow s' = \text{on} & (s' = \text{on} \lor s' = \text{off})
\end{array}
\]

(C)  

(D)

Can you justify your choice?

..........................................................................................................................  
..........................................................................................................................  
..........................................................................................................................
Task 4

For this task you should assume that a shape can only be one colour and that the following Z definitions are given.

\[
\begin{align*}
SHAPE & ::= square \mid circle \mid triangle \mid ... \\
COLOUR & ::= red \mid green \mid blue \mid ...
\end{align*}
\]

\[
\begin{align*}
\text{shape} : SHAPE \\
\text{colour} : COLOUR
\end{align*}
\]

Part A

The following expression was taken from the predicate part of a Z schema:

\[
(shape = \text{circle}) \Rightarrow (colour = \text{blue})
\]

Based on this expression alone, if \text{shape} = \text{circle} what can you say about the value of \text{colour} ? (Please circle the letter of your choice).

(A) colour \neq \text{blue} \\
(B) colour = \text{blue} \\
(C) colour = \text{green} \\
(D) Nothing

Part B

The following expression was taken from the predicate part of a Z schema:

\[
(shape = \text{circle}) \Rightarrow (colour = \text{blue})
\]

Based on this expression alone, if \text{colour} = \text{red} what can you say about the value of \text{shape} ? (Please circle the letter of your choice).

(A) \text{shape} \neq \text{circle} \\
(B) \text{shape} = \text{circle} \\
(C) \text{shape} = \text{square} \\
(D) Nothing
Part C

The following expression was taken from the predicate part of a Z schema:

\((shape = triangle) \Rightarrow (colour = red)\)

Based on this expression alone, if \(shape = square\) what can you say about the value of \(colour\)? (Please circle the letter of your choice).

(A) \(colour \neq blue\)
(B) \(colour = green\)
(C) \(colour \neq red\)
(D) Nothing

Part D

The following expression was taken from the predicate part of a Z schema:

\((shape = square) \Rightarrow (colour = green)\)

Based on this expression alone, if \(colour = green\) what can you say about the value of \(shape\)? (Please circle the letter of your choice).

(A) \(shape = square\)
(B) \(shape \neq square\)
(C) \(shape = triangle\)
(D) Nothing

Part E

The following expression was taken from the predicate part of a Z schema:

\(\neg(shape = circle) \Rightarrow (colour = blue)\)

Based on this expression alone, if \(colour \neq blue\) what can you say about the value of \(shape\)? (Please circle the letter of your choice).

(A) \(shape = square\)
(B) \(shape \neq circle\)
(C) \(shape = circle\)
(D) Nothing

Thank you for participating in this experiment.

If you would like to know your score please write your e-mail address here: .................................

Please return completed forms to:

Rick Vinter. School of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Experiment 2: Conditional Reasoning
(Abstract Formal Logic Group)

Computer Science Research Experiment:
Reasoning About Formal Software Specifications.

About Yourself

Occupation: Staff / Student / Other (please specify) ....................... Age: ....
Organisation: ........................................... Course (if applicable): ....................
Z experience: .... years Other formal notations known: ......................
How would you assess your knowledge of the Z notation? Novice/Proficient/Expert
Which types of formal logic have you studied? Propositional calculus / Predicate calculus / Boolean algebra / Other (please specify) ..........................

Instructions

In each of the tasks that follow, you will be shown a Z operational schema and a description of the operation's execution. You will be asked to determine which one of four given statements follow from the information given. Please circle the letter of your choice. You will also be asked to give a confidence rating, which should indicate how far you believe your answer to be correct. Please complete all tasks to the best of your ability, without reference to textbooks. The experiment should take no longer than 30 minutes to complete.

You may assume that the following definitions are global:

\[
\begin{align*}
\text{SHAPE} & ::= \text{square} \mid \text{circle} \mid \text{triangle} \mid \text{rectangle} \\
\text{COLOUR} & ::= \text{red} \mid \text{green} \mid \text{blue} \mid \text{white}
\end{align*}
\]

\[\text{ShapeAndColour} \]

\[\begin{array}{l}
\text{shape} : \text{SHAPE} \\
\text{colour} : \text{COLOUR}
\end{array}\]

Tasks

(1) If \(\text{shape} = \text{circle}\) before its execution, what can you say about the value of \(\text{colour}'\) after operation \(\text{SetColour}\) has executed?

\[
\text{SetColour} \\
\Delta \text{ShapeAndColour} \\
(\text{shape} = \text{circle}) \Rightarrow (\text{colour}' = \text{blue}) \\
\text{shape}' = \text{shape}
\]

Confidence rating: Not confident / Guess / Confident

(2) If \(\text{colour}' = \text{blue}\) after its execution, what can you say about the value of \(\text{shape}\) before operation \(\text{SetColour}\) has executed?

\[
\text{SetColour} \\
\Delta \text{ShapeAndColour} \\
(\text{shape} = \text{circle}) \Rightarrow (\text{colour}' \neq \text{blue}) \\
\text{shape}' = \text{shape}
\]

Confidence rating: Not confident / Guess / Confident
(3) If \( \text{shape} \neq \text{triangle} \) before its execution, what can you say about the value of \( \text{colour'} \) after operation \( \text{SetColour} \) has executed?

\[
\begin{array}{ll}
\text{SetColour} \\
\text{\( \Delta \text{ShapeAndColour} \)} \\
\text{(shape = triangle) } \Rightarrow \text{ (colour'} \neq \text{ green)} \\
\text{shape'} = \text{ shape} \\
\end{array}
\]

Confidence rating: Not confident / Guess / Confident

(A) \( \text{colour'} \neq \text{ white} \)  
(B) \( \text{colour'} \neq \text{ green} \)  
(C) \( \text{colour'} = \text{ green} \)  
(D) Nothing

(4) If \( \text{colour} \neq \text{ red} \) before its execution, what can you say about the value of \( \text{shape'} \) after operation \( \text{SetShape} \) has executed?

\[
\begin{array}{ll}
\text{SetShape} \\
\text{\( \Delta \text{ShapeAndColour} \)} \\
\text{(colour} \neq \text{ red}) \Rightarrow \text{ (shape'} \neq \text{ triangle)} \\
\text{colour'} = \text{ colour} \\
\end{array}
\]

Confidence rating: Not confident / Guess / Confident

(A) \( \text{shape'} = \text{ square} \)  
(B) \( \text{shape'} \neq \text{ triangle} \)  
(C) \( \text{shape'} = \text{ rectangle} \)  
(D) Nothing

(5) If \( \text{shape'} = \text{ rectangle} \) after its execution, what can you say about the value of \( \text{colour} \) before operation \( \text{SetShape} \) has executed?

\[
\begin{array}{ll}
\text{SetShape} \\
\text{\( \Delta \text{ShapeAndColour} \)} \\
\text{(colour} \neq \text{ white}) \Rightarrow \text{ (shape'} = \text{ rectangle)} \\
\text{colour'} = \text{ colour} \\
\end{array}
\]

Confidence rating: Not confident / Guess / Confident

(A) \( \text{colour} = \text{ white} \)  
(B) \( \text{colour} \neq \text{ white} \)  
(C) \( \text{colour} \neq \text{ blue} \)  
(D) Nothing

(6) If \( \text{shape'} = \text{ triangle} \) after its execution, what can you say about the value of \( \text{colour} \) before operation \( \text{SetShape} \) has executed?

\[
\begin{array}{ll}
\text{SetShape} \\
\text{\( \Delta \text{ShapeAndColour} \)} \\
\text{(colour} \neq \text{ green}) \Rightarrow \text{ (shape'} \neq \text{ triangle)} \\
\text{colour'} = \text{ colour} \\
\end{array}
\]

Confidence rating: Not confident / Guess / Confident

(A) \( \text{colour} = \text{ white} \)  
(B) \( \text{colour} = \text{ green} \)  
(C) \( \text{colour} \neq \text{ blue} \)  
(D) Nothing

(7) If \( \text{shape'} \neq \text{ circle} \) after its execution, what can you say about the value of \( \text{colour} \) before operation \( \text{SetShape} \) has executed?

\[
\begin{array}{ll}
\text{SetShape} \\
\text{\( \Delta \text{ShapeAndColour} \)} \\
\text{(colour} \neq \text{ green}) \Rightarrow \text{ (shape'} \neq \text{ circle)} \\
\text{colour'} = \text{ colour} \\
\end{array}
\]

Confidence rating: Not confident / Guess / Confident

(A) \( \text{colour} \neq \text{ green} \)  
(B) \( \text{colour} = \text{ green} \)  
(C) \( \text{colour} \neq \text{ blue} \)  
(D) Nothing
(8) If \( \text{colour}' = \text{white} \) after its execution, what can you say about the value of \( \text{shape} \) before operation \( \text{SetColour} \) has executed?

\[
\begin{array}{l}
\text{SetColour} \\
\quad \Delta \text{ShapeAndColour} \\
\quad (\text{shape} = \text{triangle}) \Rightarrow (\text{colour}' = \text{white}) \\
\quad \text{shape}' = \text{shape} \\
\end{array}
\]

(A) \( \text{shape} \neq \text{triangle} \)  
(B) \( \text{shape} = \text{triangle} \)  
(C) \( \text{shape} = \text{square} \)  
(D) Nothing

Confidence rating: Not confident / Guess / Confident

(9) If \( \text{shape} \neq \text{square} \) before its execution, what can you say about the value of \( \text{colour}' \) after operation \( \text{SetColour} \) has executed?

\[
\begin{array}{l}
\text{SetColour} \\
\quad \Delta \text{ShapeAndColour} \\
\quad (\text{shape} = \text{square}) \Rightarrow (\text{colour}' = \text{red}) \\
\quad \text{shape}' = \text{shape} \\
\end{array}
\]

(A) \( \text{colour}' \neq \text{red} \)  
(B) \( \text{colour}' = \text{red} \)  
(C) \( \text{colour}' \neq \text{white} \)  
(D) Nothing

Confidence rating: Not confident / Guess / Confident

(10) If \( \text{shape}' \neq \text{square} \) after its execution, what can you say about the value of \( \text{colour} \) before operation \( \text{SetShape} \) has executed?

\[
\begin{array}{l}
\text{SetShape} \\
\quad \Delta \text{ShapeAndColour} \\
\quad (\text{colour} \neq \text{white}) \Rightarrow (\text{shape}' = \text{square}) \\
\quad \text{colour}' = \text{colour} \\
\end{array}
\]

(A) \( \text{colour} = \text{white} \)  
(B) \( \text{colour} = \text{green} \)  
(C) \( \text{colour} \neq \text{white} \)  
(D) Nothing

Confidence rating: Not confident / Guess / Confident

(11) If \( \text{colour} = \text{white} \) before its execution, what can you say about the value of \( \text{shape}' \) after operation \( \text{SetShape} \) has executed?

\[
\begin{array}{l}
\text{SetShape} \\
\quad \Delta \text{ShapeAndColour} \\
\quad (\text{colour} \neq \text{white}) \Rightarrow (\text{shape}' = \text{circle}) \\
\quad \text{colour}' = \text{colour} \\
\end{array}
\]

(A) \( \text{shape}' = \text{circle} \)  
(B) \( \text{shape}' \neq \text{circle} \)  
(C) \( \text{shape}' = \text{triangle} \)  
(D) Nothing

Confidence rating: Not confident / Guess / Confident

(12) If \( \text{colour} = \text{blue} \) before its execution, what can you say about the value of \( \text{shape}' \) after operation \( \text{SetShape} \) has executed?

\[
\begin{array}{l}
\text{SetShape} \\
\quad \Delta \text{ShapeAndColour} \\
\quad (\text{colour} \neq \text{blue}) \Rightarrow (\text{shape}' \neq \text{square}) \\
\quad \text{colour}' = \text{colour} \\
\end{array}
\]

(A) \( \text{shape}' = \text{circle} \)  
(B) \( \text{shape}' = \text{square} \)  
(C) \( \text{shape}' \neq \text{square} \)  
(D) Nothing

Confidence rating: Not confident / Guess / Confident
(13) If colour ≠ white before its execution, what can you say about the value of shape' after operation SetShape has executed?

<table>
<thead>
<tr>
<th>SetShape</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔShapeAndColour</td>
</tr>
<tr>
<td>(colour ≠ white) ⇒ (shape' = square)</td>
</tr>
<tr>
<td>colour' = colour</td>
</tr>
</tbody>
</table>

Confidence rating: Not confident / Guess / Confident

(14) If shape = rectangle before its execution, what can you say about the value of colour' after operation SetColour has executed?

<table>
<thead>
<tr>
<th>SetColour</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔShapeAndColour</td>
</tr>
<tr>
<td>(shape = rectangle) ⇒ (colour' ≠ red)</td>
</tr>
<tr>
<td>shape' = shape</td>
</tr>
</tbody>
</table>

Confidence rating: Not confident / Guess / Confident

(15) If colour' ≠ blue after its execution, what can you say about the value of shape before operation SetColour has executed?

<table>
<thead>
<tr>
<th>SetColour</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔShapeAndColour</td>
</tr>
<tr>
<td>(shape = circle) ⇒ (colour' ≠ blue)</td>
</tr>
<tr>
<td>shape' = shape</td>
</tr>
</tbody>
</table>

Confidence rating: Not confident / Guess / Confident

(16) If colour' ≠ red after its execution, what can you say about the value of shape before operation SetColour has executed?

<table>
<thead>
<tr>
<th>SetColour</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔShapeAndColour</td>
</tr>
<tr>
<td>(shape = rectangle) ⇒ (colour' = red)</td>
</tr>
<tr>
<td>shape' = shape</td>
</tr>
</tbody>
</table>

Confidence rating: Not confident / Guess / Confident

Thank you for participating in this experiment.

If you would like to know your score please write your e-mail address here: ..................................................

Please return completed forms to:

Rick Vinter. School of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Experiment 2: Conditional Reasoning
(Abstract Natural Language Group)

Computer Science Research Experiment:
Reasoning About Natural Language Arguments.

About Yourself

Occupation: Staff / Student / Other (please specify) ......................... Age: ......
Organisation: .................................. Course (if applicable): ......................
Have you studied any systems of formal logic before? Propositional calculus / Predicate calculus / Boolean algebra / Other (please specify) ........................................

Instructions

In each of the tasks that follow, you will be shown a description of a colours and shapes scenario. You will be asked to determine which one of four given statements follow from the scenario described. Please circle the letter of your choice. You will also be asked to give a confidence rating, which should indicate how far you believe your answer to be correct. Please complete all tasks to the best of your ability, without reference to textbooks. The experiment should take no longer than 30 minutes to complete.

Tasks

(1) If the shape is a circle then the colour is blue.
    The shape is a circle.
    Based on the above description, what can you say about colour?
    (A) the colour is not blue   
    (B) the colour is blue      
    (C) the colour is green    
    (D) nothing
    Confidence rating: Not confident / Guess / Confident

(2) If the shape is a circle then the colour is not blue.
    The colour is blue.
    Based on the above description, what can you say about shape?
    (A) the shape is a circle   
    (B) the shape is a rectangle
    (C) the shape is not a circle
    (D) nothing
    Confidence rating: Not confident / Guess / Confident

(3) If the shape is a triangle then the colour is not green.
    The shape is not a triangle.
    Based on the above description, what can you say about colour?
    (A) the colour is not white  
    (B) the colour is not green 
    (C) the colour is green     
    (D) nothing
    Confidence rating: Not confident / Guess / Confident
(4) If the colour is not red then the shape is not a triangle.  
The colour is not red.  

Based on the above description, what can you say about shape?  

(A) the shape is a square  
(B) the shape is not a triangle  
(C) the shape is a rectangle  
(D) nothing  

Confidence rating: Not confident / Guess / Confident

(5) If the colour is not white then the shape is a rectangle.  
The shape is a rectangle.  

Based on the above description, what can you say about colour?  

(A) the colour is white  
(B) the colour is not white  
(C) the colour is not blue  
(D) nothing  

Confidence rating: Not confident / Guess / Confident

(6) If the colour is not green then the shape is not a triangle.  
The shape is a triangle.  

Based on the above description, what can you say about colour?  

(A) the colour is white  
(B) the colour is green  
(C) the colour is not blue  
(D) nothing  

Confidence rating: Not confident / Guess / Confident

(7) If the colour is not green then the shape is not a circle.  
The shape is not a circle.  

Based on the above description, what can you say about colour?  

(A) the colour is not green  
(B) the colour is green  
(C) the colour is not blue  
(D) nothing  

Confidence rating: Not confident / Guess / Confident

(8) If the shape is a triangle then the colour is white.  
The colour is white.  

Based on the above description, what can you say about shape?  

(A) the shape is not a triangle  
(B) the shape is a triangle  
(C) the shape is a square  
(D) nothing  

Confidence rating: Not confident / Guess / Confident
(9) If the shape is a square then the colour is red.
    The shape is not a square.
    Based on the above description, what can you say about colour?
    (A) the colour is not red
    (B) the colour is red
    (C) the colour is not white
    (D) nothing
    Confidence rating: Not confident / Guess / Confident

(10) If the colour is not white then the shape is a square.
    The shape is not a square.
    Based on the above description, what can you say about colour?
    (A) the colour is white
    (B) the colour is green
    (C) the colour is not white
    (D) nothing
    Confidence rating: Not confident / Guess / Confident

(11) If the colour is not white then the shape is a circle.
    The colour is white.
    Based on the above description, what can you say about shape?
    (A) the shape is a circle
    (B) the shape is not a circle
    (C) the shape is a triangle
    (D) nothing
    Confidence rating: Not confident / Guess / Confident

(12) If the colour is not blue then the shape is not a square.
    The colour is blue.
    Based on the above description, what can you say about shape?
    (A) the shape is a circle
    (B) the shape is a square
    (C) the shape is not a square
    (D) nothing
    Confidence rating: Not confident / Guess / Confident

(13) If the colour is not white then the shape is a square.
    The colour is not white.
    Based on the above description, what can you say about shape?
    (A) the shape is a square
    (B) the shape is not a circle
    (C) the shape is not a square
    (D) nothing
    Confidence rating: Not confident / Guess / Confident
(14) If the shape is a rectangle then the colour is not red.
The shape is a rectangle.

Based on the above description, what can you say about colour?

(A) the colour is not blue
(B) the colour is not red
(C) the colour is red
(D) nothing

Confidence rating: Not confident / Guess / Confident

(15) If the shape is a circle then the colour is not blue.
The colour is not blue.

Based on the above description, what can you say about shape?

(A) the shape is not a rectangle
(B) the shape is a circle
(C) the shape is not a circle
(D) nothing

Confidence rating: Not confident / Guess / Confident

(16) If the shape is a rectangle then the colour is red.
The colour is not red.

Based on the above description, what can you say about shape?

(A) the shape is not a rectangle
(B) the shape is a rectangle
(C) the shape is a square
(D) nothing

Confidence rating: Not confident / Guess / Confident

Thank you for participating in this experiment.

If you would like to know your score please write your e-mail address here: ........................................

Please return completed forms to:

Rick Vinter. School of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Experiment 2: Conditional Reasoning
(Thematic Formal Logic Group)

Computer Science Research Experiment:
Reasoning About Formal Software Specifications.

About Yourself

Occupation: Staff / Student / Other (please specify) ..................... Age: ....
Organisation: ........................................... Course (if applicable): ......................
Z experience: ....... years Other formal notations known: ......................
How would you assess your knowledge of the Z notation? Novice/Proficient/Expert
Which types of formal logic have you studied? Propositional calculus / Predicate
calculus / Boolean algebra / Other (please specify) ......................

Instructions

In each of the tasks that follow, you will be shown a Z operational schema and a description of the operation's execution. You will be asked to determine which one of four given statements follow from the information given. Please circle the letter of your choice. You will also be asked to give a confidence rating, which should indicate how far you believe your answer to be correct. Please complete all tasks to the best of your ability, without reference to textbooks. The experiment should take no longer than 30 minutes to complete.

Tasks

(1) If selected_op = Maximise before its execution, what can you say about window_coords! after operation MaximiseWindow has executed?

```
MaximiseWindow

\exists ScreenManager
window_coords! : COORDS
selected_op? : OperationType

\text{selected_op? = Maximise} \Rightarrow \text{window_coords! = screen_coords}
```

(A) \text{window_coords!} = (0,648) \hspace{1cm} (C) \text{window_coords!} = \text{screen_coords}
(B) \text{window_coords!} \neq \text{screen_coords} \hspace{1cm} (D) \text{Nothing}

Confidence rating: \hspace{0.5cm} \square \hspace{0.2cm} \text{Not confident} \hspace{0.2cm} \square \hspace{0.2cm} \text{Guess} \hspace{0.2cm} \square \hspace{0.2cm} \text{Confident}

(2) If door_state' = door_state after its execution, what can you say about card_status? before operation DoorSecurityCheck has executed?

```
SecurityCheck

\Delta DoorInfo
\text{card_status? : Status}

\text{card_status? = Valid} \Rightarrow \neg (\text{door_state'} = \text{door_state})
```

(A) \neg \text{(card_status? = Valid)} \hspace{1cm} (C) \text{card_status? = Invalid}
(B) \text{card_status? = Valid} \hspace{1cm} (D) \text{Nothing}

Confidence rating: \hspace{0.5cm} \square \hspace{0.2cm} \text{Not confident} \hspace{0.2cm} \square \hspace{0.2cm} \text{Guess} \hspace{0.2cm} \square \hspace{0.2cm} \text{Confident}
(3) If \( \neg(reactor\_status! = Ok) \) after its execution, what can you say about coolertemp before operation \( ReactorTempCheck \) has executed?

\[
\begin{align*}
\text{ReactorTempCheck} & \quad \Xi\text{NuclearPlantStatus} \\
& \quad \text{reactor\_status! : Report} \\
& \quad \text{coolertemp} > \text{Maxtemp} \Rightarrow \neg(\text{reactor\_status!} = \text{Ok})
\end{align*}
\]

(A) \( \text{coolertemp} \leq \text{Maxtemp} \)  (C) \( \text{coolertemp} > \text{Mintemp} \)

(B) \( \text{coolertemp} > \text{Maxtemp} \)  (D) Nothing

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident

(4) If \( \neg(\text{current\_location?} = \text{target\_location?}) \) before its execution, what can you say about status! after operation \( GuidedMissileCheck \) has executed?

\[
\begin{align*}
\text{GuidedMissileCheck} & \quad \Xi\text{status! : Report} \\
& \quad \text{current\_location?, target\_location?: COORDINATES} \\
& \quad \neg(\text{current\_location?} = \text{target\_location?}) \Rightarrow \neg(\text{status!} = \text{Success})
\end{align*}
\]

(A) \( \neg(\text{status!} = \text{Success}) \)  (C) \( \text{status!} = \text{Success} \)

(B) \( \text{status!} = \text{Failure} \)  (D) Nothing

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident

(5) If \( queue' = \text{queue} \land (job?) \) after its execution, what can you say about \( \text{status(printer?)} \) before operation \( ProcessJob \) has executed?

\[
\begin{align*}
\text{ProcessJob} & \quad \Delta\text{PrintQueue} \\
& \quad \text{printer? : Printer} \\
& \quad \text{job? : PrintJob} \\
& \quad \neg(\text{status(printer?)} = \text{Unservicable}) \Rightarrow queue' = \text{queue} \land (\text{job?})
\end{align*}
\]

(A) \( \text{status(printer?)} = \text{Servicable} \)  (C) \( \neg(\text{status(printer?)} = \text{Unservicable}) \)

(B) \( \text{status(printer?)} = \text{Unservicable} \)  (D) Nothing

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident

(6) If \( \text{report!} = \text{MagHire} \) after its execution, what can you say about \( \text{age(cust?)} \) before operation \( HireVideo \) has executed?

\[
\begin{align*}
\text{HireVideo} & \quad \Xi\text{film? : Video} \\
& \quad \text{cust? : Member} \\
& \quad \text{report! : Report} \\
& \quad \neg(\text{age(cust?)} \geq \text{certificate(film?)}) \Rightarrow \neg(\text{report!} = \text{MagHire})
\end{align*}
\]

(A) \( \text{age(cust?)} = \text{certificate(film?)} \)  (C) \( \text{age(cust?)} \geq \text{certificate(film?)} \)

(B) \( \text{age(cust?)} < \text{certificate(film?)} \)  (D) Nothing

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident
(7) If \( \neg(\text{docstatus} = \text{Empty}) \) after its execution, what can you say about \text{doc}? before operation \text{InputText} has executed?

\[
\text{InputText}
\begin{align*}
\text{doc} & : \text{seq CHAR} \\
\text{docstatus} & : \text{Report}
\end{align*}
\]

\[\neg(\text{doc} = \{\}) \Rightarrow \neg(\text{docstatus} = \text{Empty})\]

(A) \text{doc} = \{\} \\
(B) \neg(\text{doc} = \{\}) \\
(C) \neg(\text{doc} \neq \{\}) \\
(D) \text{Nothing}

Confidence rating: \[\square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident}\]

(8) If \( DB' = DB \cup \text{newrec} \) after its execution, what can you say about \text{newrec}? before operation \text{AddNewRecord} has executed?

\[
\text{AddNewRecord}
\begin{align*}
\Delta \text{Database} \\
\text{newrec} & : \text{Record}
\end{align*}
\]

\[\text{newrec} \notin DB \Rightarrow (DB' = DB \cup \text{newrec})\]

(A) \(\neg(\text{newrec} \notin DB)\) \\
(B) \text{newrec} \notin DB \\
(C) \text{newrec} \in DB \\
(D) \text{Nothing}

Confidence rating: \[\square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident}\]

(9) If \( \neg(\text{password} = \text{Correct}) \) before its execution, what can you say about \text{LoggedUsers'} after operation \text{AccessSystem} has executed?

\[
\text{AccessSystem}
\begin{align*}
\Delta \text{UserDatabase} \\
\text{username} & : \text{dom userinfo} \\
\text{password} & : \text{Status}
\end{align*}
\]

\[\text{password} = \text{Correct} \Rightarrow (\text{LoggedUsers'} = \text{LoggedUsers} \cup \text{username'})\]

(A) \(\text{LoggedUsers'} = \text{LoggedUsers} \cup \text{username'}\) \\
(B) \(\neg(\text{LoggedUsers'} = \text{LoggedUsers} \cup \text{username'})\) \\
(C) \text{LoggedUsers'} = \text{username'} \\
(D) \text{Nothing}

Confidence rating: \[\square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident}\]

(10) If \( \neg(\text{members'} = \text{members} \cup \text{applicant'}) \) after its execution, what can you say about \text{applicant}? before operation \text{CheckFootballID} has executed?

\[
\text{CheckFootballID}
\begin{align*}
\Delta \text{FootballDB} \\
\text{applicant} & : \text{Person}
\end{align*}
\]

\[\neg(\text{applicant'} \in \text{banned}) \Rightarrow (\text{members'} = \text{members} \cup \text{applicant'})\]

(A) \text{applicant'} \in \text{banned} \\
(B) \text{applicant'} \notin \text{banned} \\
(C) \text{applicant'} \in \text{members} \\
(D) \text{Nothing}

Confidence rating: \[\square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident}\]
(11) If fullybooked(flight?) before its execution, what can you say about bookings' after operation BookSeat has executed?

<table>
<thead>
<tr>
<th>BookSeat</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔFlightDB</td>
</tr>
<tr>
<td>flight?: Destination → TIME</td>
</tr>
<tr>
<td>pass?: Passenger</td>
</tr>
<tr>
<td>¬fullybooked(flight?) ⇒ bookings' = bookings ∪ {flight? → pass?}</td>
</tr>
</tbody>
</table>

(A) bookings' = bookings ∪ {flight? → pass?}  (C) bookings' = {flight? → pass?}  
(B) (bookings' = bookings ∪ {flight? → pass?})  (D) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(12) If #register ≤ Maxstudents before its execution, what can you say about student? after operation AddStudent has executed?

<table>
<thead>
<tr>
<th>AddStudent</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔRegister</td>
</tr>
<tr>
<td>student?: Student</td>
</tr>
<tr>
<td>¬(#register ≤ Maxstudents) ⇒ ¬(student? ∈ register')</td>
</tr>
</tbody>
</table>

(A) ¬(student? ∈ register')  (C) student? ∈ register' |
| (B) student? ∉ register'  (D) Nothing |

Confidence rating: □ Not confident □ Guess □ Confident

(13) If ¬(book? ∈ stock) before its execution, what can you say about stock' after operation NewStock has executed?

<table>
<thead>
<tr>
<th>NewStock</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔLibraryStockDB</td>
</tr>
<tr>
<td>book?: Book</td>
</tr>
</tbody>
</table>

(A) ¬(stock' = stock ∪ book?)  (C) stock' = stock ∪ book? |
| (B) stock' = stock  (D) Nothing |

Confidence rating: □ Not confident □ Guess □ Confident

(14) If usercredit < price(item?) before its execution, what can you say about report! after operation VendItem has executed?

<table>
<thead>
<tr>
<th>VendItem</th>
</tr>
</thead>
<tbody>
<tr>
<td>≡ UserDetails</td>
</tr>
<tr>
<td>item?: GOOD</td>
</tr>
<tr>
<td>report!: Status</td>
</tr>
<tr>
<td>usercredit &lt; price(item?) ⇒ ¬(report! = SufficientCredit)</td>
</tr>
</tbody>
</table>

(A) report! = SufficientCredit  (C) ¬(report! = SufficientCredit) |
| (B) report! = InsufficientCredit  (D) Nothing |

Confidence rating: □ Not confident □ Guess □ Confident
(15) If \( (FS\_status? = \text{Ok}) \) before its execution, what can you say about \( FS' \) after operation \( \text{FileSystem} \) has executed?

\[
\begin{array}{c}
\text{FileSystem} \\
\Delta FS \\
FS\_status? : Status \\
\hline \\
FS\_status = \text{Ok} \Rightarrow \neg (FS' = FS)
\end{array}
\]

(A) \( FS' = FS \)  (C) \( FS' \neq FS \)
(B) \( \neg (FS' = FS) \)  (D) Nothing

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident

(16) If \( \neg (\text{comstatus!} = \text{Valid}) \) after its execution, what can you say about \( \text{command?} \) before operation \( \text{Assembler} \) has executed?

\[
\begin{array}{c}
\text{Assembler} \\
\exists \text{KnownCommands} \\
\text{command?} : \text{Opcode} \Rightarrow \text{Operand} \\
\text{comstatus!} : \text{Report} \\
\hline \\
\text{command?} \in \text{knowncoms} \Rightarrow \text{comstatus!} = \text{Valid}
\end{array}
\]

(A) \( \neg (\text{command?} \in \text{knowncoms}) \)  (C) \( \text{command?} \in \text{knowncoms} \)
(B) \( \text{command?} = \{\text{JUMP}, 1000\} \)  (D) Nothing

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident

Thank you for participating in this experiment.

If you would like to know your score please write your e-mail address here: ........................................

Please return completed forms to:

Rick Vinter. School of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Experiment 3: Disjunctive and Conjunctive Reasoning  
(Abstract Formal Logic Group)

Computer Science Research Experiment: 
Reasoning About Formal Software Specifications

About Yourself

Occupation: Staff / Student / Other (please specify) ......................... Age: ........
Organisation: ................................... Course (if applicable): .........................
Z experience: ...... years Other formal notations known: .........................
How would you assess your knowledge of the Z notation? Novice/Proficient/Expert
Which types of formal logic have you studied? Propositional calculus / Predicate calculus / Boolean algebra / Other (please specify) .........................

Instructions

In each of the tasks that follow, you will be shown a Z operational schema and a description of the operation's execution. You will be asked to determine which one of four given statements follow from the information given. Please circle the letter of your choice. You will also be asked to give a confidence rating, which should indicate how far you believe your answer to be correct. Please complete all tasks to the best of your ability, without reference to textbooks. The experiment should take no longer than 30 minutes to complete. You may assume that the following definitions are global.

\[
SHAPE ::= square \mid circle \mid triangle \mid rectangle
\]
\[
COLOUR ::= red \mid green \mid blue \mid white
\]

Tasks

(1) If \( \neg (\text{shape}!) = \text{triangle} \) what can you say about \( \text{colour}! \) in operation \( \text{GetShapeColour} \)?

\[
\begin{array}{c|c}
\text{GetShapeColour} & \text{Description} \\
\hline
\text{shape}!: \text{SHAPE} & (a) \ \text{colour}! = \text{green} \\
\text{colour}!: \text{COLOUR} & (b) \ \neg (\text{colour}! = \text{green}) \\
\text{shape}! = \text{triangle} \lor \text{colour}! = \text{green} & (c) \ \text{colour}! = \text{red} \\
& (d) \ \text{Nothing}
\end{array}
\]

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident

(2) Based on its description below, what can you say about the output from operation \( \text{GetShapeColour} \)?

\[
\begin{array}{c|c}
\text{GetShapeColour} & \text{Description} \\
\hline
\text{shape}!: \text{SHAPE} & (a) \ \text{shape}! = \text{square} \land \text{colour}! = \text{blue} \\
\text{colour}!: \text{COLOUR} & (b) \ \neg (\text{colour}! = \text{blue}) \\
\neg (\text{shape}! = \text{square}) \land \text{colour}! = \text{blue} & (c) \ \neg (\text{shape}! = \text{square}) \\
& (d) \ \text{Nothing}
\end{array}
\]

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident
(3) If \( \text{shape!} = \text{square} \) what can you say about \( \text{colour!} \) in operation \( \text{GetShapeColour} \)?

\[
\begin{array}{ll}
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\text{shape! = square} \lor \text{colour! = green}
\end{array}
\]

 Confidence rating: □ Not confident □ Guess □ Confident

(a) \( \text{colour! = blue} \)  
(b) \( \text{colour! = green} \)  
(c) \( \neg(\text{colour! = green}) \)  
(d) Nothing

(4) If \( \neg(\text{colour! = blue}) \) what can you say about \( \text{shape!} \) in operation \( \text{GetShapeColour} \)?

\[
\begin{array}{ll}
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\text{shape! = triangle} \lor \neg(\text{colour! = blue})
\end{array}
\]

 Confidence rating: □ Not confident □ Guess □ Confident

(a) \( \text{shape! = triangle} \)  
(b) \( \neg(\text{shape! = square}) \)  
(c) \( \neg(\text{shape! = triangle}) \)  
(d) Nothing

(5) Based on its description below, what can you say about the output from operation \( \text{GetShapeColour} \)?

\[
\begin{array}{ll}
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\text{colour! = blue}
\end{array}
\]

 Confidence rating: □ Not confident □ Guess □ Confident

(a) \( \neg(\text{shape! = triangle}) \)  
(b) \( \text{shape! = circle} \)  
(c) \( \text{shape! = circle} \land \text{colour! = blue} \)  
(d) Nothing

(6) Based on its description below, what can you say about \( \text{shape!} \) in operation \( \text{GetShapeColour} \)?

\[
\begin{array}{ll}
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(\text{shape!} = \text{rectangle} \lor \text{colour!} = \text{blue})
\end{array}
\]

 Confidence rating: □ Not confident □ Guess □ Confident

(a) \( \text{shape! = rectangle} \)  
(b) \( \neg(\text{shape!} = \text{rectangle}) \)  
(c) \( \text{shape! = square} \)  
(d) Nothing

(7) If \( \text{shape!} = \text{circle} \) what can you say about \( \text{colour!} \) in operation \( \text{GetShapeColour} \)?

\[
\begin{array}{ll}
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\text{colour! = blue} \lor \neg(\text{shape!} = \text{circle})
\end{array}
\]

 Confidence rating: □ Not confident □ Guess □ Confident

(a) \( \text{colour! = blue} \)  
(b) \( \neg(\text{colour! = blue}) \)  
(c) \( \text{colour! = green} \)  
(d) Nothing
(8) If $\neg (\text{colour} = \text{red})$ what can you say about $\text{shape}$ in operation $\text{GetShapeColour}$?

- $\text{GetShapeColour}$
  - $\text{shape} : \text{SHAPE}$
  - $\text{colour} : \text{COLOUR}$
  - $\text{shape} = \text{circle} \lor \neg (\text{colour} = \text{red})$

Confidence rating:  □ Not confident □ Guess □ Confident

(9) If $\text{shape} = \text{rectangle}$ what can you say about $\text{colour}$ in operation $\text{GetShapeColour}$?

- $\text{GetShapeColour}$
  - $\text{shape} : \text{SHAPE}$
  - $\text{colour} : \text{COLOUR}$
  - $\text{shape} = \text{rectangle} \lor \neg (\text{colour} = \text{blue})$

Confidence rating:  □ Not confident □ Guess □ Confident

(10) Based on its description below, what can you say about the output from operation $\text{GetShapeColour}$?

- $\text{GetShapeColour}$
  - $\text{shape} : \text{SHAPE}$
  - $\text{colour} : \text{COLOUR}$
  - $\text{colour} = \text{blue} \land \neg (\text{shape} = \text{circle})$

Confidence rating:  □ Not confident □ Guess □ Confident

(11) Based on its description below, what can you say about the output from operation $\text{GetShapeColour}$?

- $\text{GetShapeColour}$
  - $\text{shape} : \text{SHAPE}$
  - $\text{colour} : \text{COLOUR}$
  - $\neg (\text{shape} = \text{square})$

Confidence rating:  □ Not confident □ Guess □ Confident

(12) Based on its description below, what can you say about the output from operation $\text{GetShapeColour}$?

- $\text{GetShapeColour}$
  - $\text{shape} : \text{SHAPE}$
  - $\text{colour} : \text{COLOUR}$
  - $\neg (\text{colour} = \text{green})$

Confidence rating:  □ Not confident □ Guess □ Confident
(13) If \(\neg(colour! = \text{blue})\) what can you say about \(shape!\) in operation \textit{GetShapeColour}?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(colour! = \text{blue}) \lor shape! = \text{triangle}
\]

Confidence rating: □ Not confident □ Guess □ Confident

(14) If \(colour! = \text{green}\) what can you say about \(shape!\) in operation \textit{GetShapeColour}?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(colour! = \text{green}) \lor shape! = \text{square}
\]

Confidence rating: □ Not confident □ Guess □ Confident

(15) Based on its description below, what can you say about the output from operation \textit{GetShapeColour}?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
shape! = \text{circle} \land \text{colour!} = \text{white}
\]

Confidence rating: □ Not confident □ Guess □ Confident

(16) If \(\neg(colour! = \text{white})\) what can you say about \(shape!\) in operation \textit{GetShapeColour}?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\text{colour!} = \text{white} \lor \neg(shape! = \text{rectangle})
\]

Confidence rating: □ Not confident □ Guess □ Confident

(17) If \(colour! = \text{white}\) what can you say about \(shape!\) in operation \textit{GetShapeColour}?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
shape! = \text{circle} \lor \text{colour!} = \text{white}
\]

Confidence rating: □ Not confident □ Guess □ Confident
(18) Based on its description below, what can you say about colour! in operation GetShapeColour?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(\text{shape!} = \text{square} \lor \text{colour!} = \text{white})
\]

(a) colour! = white
(b) colour! = white \lor \text{shape!} = \text{square}
(c) \neg(\text{colour!} = \text{white})
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(19) If shape! = square what can you say about colour! in operation GetShapeColour?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(\text{shape!} = \text{square} \land \text{colour!} = \text{red})
\]

(a) colour! = red
(b) \neg(\text{colour!} = \text{red})
(c) colour! = blue
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(20) If \neg(\text{colour!} = \text{red}) what can you say about shape! in operation GetShapeColour?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(\text{shape!} = \text{circle}) \lor \neg(\text{shape!} = \text{rectangle})
\]

(a) \neg(\text{shape!} = \text{circle})
(b) \neg(\text{shape!} = \text{rectangle})
(c) \text{shape!} = \text{circle}
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(21) If shape! = square what can you say about colour! in operation GetShapeColour?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(\text{colour!} = \text{red}) \lor \text{shape!} = \text{square}
\]

(a) colour! = red
(b) \neg(\text{colour!} = \text{red})
(c) colour! = green
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(22) If shape! = circle what can you say about colour! in operation GetShapeColour?

\[
\text{GetShapeColour} \\
\text{shape! : SHAPE} \\
\text{colour! : COLOUR} \\
\neg(\text{colour!} = \text{white}) \lor \neg(\text{shape!} = \text{circle})
\]

(a) \neg(\text{colour!} = \text{white})
(b) colour! = white
(c) \neg(\text{colour!} = \text{white})
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident
(23) If \( \neg (\text{shape!} = \text{rectangle}) \) what can you say about \( \text{colour!} \) in operation \( \text{GetShapeColour} \)?

\[
\begin{align*}
\text{GetShapeColour} \\
\text{shape!} : \text{SHAPE} & \quad (a) \quad \text{colour!} = \text{green} \\
\text{colour!} : \text{COLOUR} & \quad (b) \quad \neg (\text{colour!} = \text{green}) \\
\neg (\text{colour!} = \text{green}) \lor \neg (\text{shape!} = \text{rectangle}) & \quad (c) \quad \text{colour!} = \text{red} \\
& \quad (d) \quad \text{Nothing}
\end{align*}
\]

Confidence rating: □ Not confident □ Guess □ Confident

(24) Based on its description below, what can you say about the output from operation \( \text{GetShapeColour} \)?

\[
\begin{align*}
\text{GetShapeColour} \\
\text{shape!} : \text{SHAPE} & \quad (a) \quad \neg (\text{shape!} = \text{triangle}) \\
\text{colour!} : \text{COLOUR} & \quad (b) \quad \text{shape!} = \text{triangle} \land \neg (\text{colour!} = \text{red}) \\
\text{shape!} = \text{triangle} & \quad (c) \quad \neg (\text{colour!} = \text{green}) \\
& \quad (d) \quad \text{Nothing}
\end{align*}
\]

Confidence rating: □ Not confident □ Guess □ Confident

(25) If \( \text{colour!} = \text{red} \) what can you say about \( \text{shape!} \) in operation \( \text{GetShapeColour} \)?

\[
\begin{align*}
\text{GetShapeColour} \\
\text{shape!} : \text{SHAPE} & \quad (a) \quad \text{shape!} = \text{square} \\
\text{colour!} : \text{COLOUR} & \quad (b) \quad \neg (\text{shape!} = \text{square}) \\
\neg (\text{colour!} = \text{red}) \lor \neg (\text{shape!} = \text{square}) & \quad (c) \quad \text{shape!} = \text{circle} \\
& \quad (d) \quad \text{Nothing}
\end{align*}
\]

Confidence rating: □ Not confident □ Guess □ Confident

(26) If \( \neg (\text{colour!} = \text{green}) \) what can you say about \( \text{shape!} \) in operation \( \text{GetShapeColour} \)?

\[
\begin{align*}
\text{GetShapeColour} \\
\text{shape!} : \text{SHAPE} & \quad (a) \quad \neg (\text{shape!} = \text{circle}) \\
\text{colour!} : \text{COLOUR} & \quad (b) \quad \neg (\text{shape!} = \text{rectangle}) \\
\neg (\text{colour!} = \text{green}) \lor \neg (\text{shape!} = \text{rectangle}) & \quad (c) \quad \text{shape!} = \text{rectangle} \\
& \quad (d) \quad \text{Nothing}
\end{align*}
\]

Confidence rating: □ Not confident □ Guess □ Confident

(27) Based on its description below, what can you say about the output from operation \( \text{GetShapeColour} \)?

\[
\begin{align*}
\text{GetShapeColour} \\
\text{shape!} : \text{SHAPE} & \quad (a) \quad \neg (\text{shape!} = \text{square}) \\
\text{colour!} : \text{COLOUR} & \quad (b) \quad \text{colour!} = \text{white} \land \text{shape!} = \text{square} \\
\neg (\text{colour!} = \text{white}) \land \neg (\text{shape!} = \text{square}) & \quad (c) \quad \text{colour!} = \text{white} \\
& \quad (d) \quad \text{Nothing}
\end{align*}
\]

Confidence rating: □ Not confident □ Guess □ Confident
(28) If `colour! = white` what can you say about `shape!` in operation `GetShapeColour`?

<table>
<thead>
<tr>
<th>GetShapeColour</th>
<th>(a) $\neg (shape! = square)$</th>
<th>(b) $\neg (shape! = circle)$</th>
<th>(c) $shape! = circle$</th>
<th>(d) Nothing</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>shape! : SHAPE</code></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>colour! : COLOUR</code></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\neg (shape! = circle \land colour! = white)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Confidence rating: □ Not confident □ Guess □ Confident

Thank you for participating in this experiment.

If you would like to know your score please write your e-mail address here: ..........................................

Please return completed forms to:

Rick Vinter. School of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Experiment 3: Disjunctive and Conjunctive Reasoning
(Thematic Formal Logic Group)

Computer Science Research Experiment:
Reasoning About Formal Software Specifications

About Yourself

Occupation: Staff / Student / Other (please specify) ......................... Age: ....
Organisation: ................................ Course (if applicable): ..............................
Z experience: .... years Other formal notations known: ..............................
How would you assess your knowledge of the Z notation? Novice/Proficient/Expert
Which types of formal logic have you studied? Propositional calculus / Predicate
calculus / Boolean algebra / Other (please specify) ..............................

Instructions

In each of the tasks that follow, you will be shown a Z operational schema and a description
of the operation's execution. You will be asked to determine which one of four given
statements follow from the information given. Please circle the letter of your choice. You
will also be asked to give a confidence rating, which should indicate how far you believe
your answer to be correct. Please complete all tasks to the best of your ability, without
reference to textbooks. The experiment should take no longer than 30 minutes to complete.

Tasks

(1) If \( \neg (\text{venue} = \text{England}) \) after its execution, what can you say about \( \text{officials}! \) in
operation \( \text{FootballChampionship} \)?

\[
\text{FootballChampionship}
\]
\[
\text{venue!} : \text{Location} \\
\text{officials!} : \text{RulingBody}
\]

\[
\text{venue!} = \text{England} \lor \text{officials!} = \text{UEFA}
\]

(a) \( \text{officials!} = \text{UEFA} \)  
(b) \( \neg (\text{officials!} = \text{UEFA}) \)  
(c) \( \text{officials!} = \text{English} \)  
(d) Nothing

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident

(2) What can you say about the effect of operation \( \text{HireVideo} \) on its after-state variables?

\[
\text{HireVideo}
\]
\[
\Delta \text{VideoShop}
\]

\[
\neg (\text{film'} \in \text{FilmsOnShelf}) \land \text{report'} = \text{OnLoan}
\]

(a) \( \text{film'} \in \text{FilmsOnShelf} \land \text{report'} = \text{OnLoan} \)  
(b) \( \neg (\text{film'} \in \text{FilmsOnShelf}) \)  
(c) \( \neg (\text{film'} \in \text{FilmsOnShelf}) \land \text{report'} = \text{OnLoan} \)  
(d) Nothing

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident
(3) If \( \text{film!} = \text{Babe} \) after its execution, what can you say about \( \text{ticket\_price!} \) in operation \( \text{Cinema} \)?

\[
\begin{align*}
\text{Cinema} \\
\text{film! : Film} \\
\text{ticket\_price! : N} \\
\text{film!} = \text{Babe} \lor \text{ticket\_price!} = 5
\end{align*}
\]

(a) \( \text{ticket\_price!} = 4 \)  
(b) \( \text{ticket\_price!} = 5 \)  
(c) \( \neg(\text{ticket\_price!} = 5) \)  
(d) Nothing

Confidence rating:  
□ Not confident  □ Guess  □ Confident

(4) If \( \neg(\text{title!} = \text{YourMajesty}) \) after its execution, what can you say about \( \text{person!} \) in operation \( \text{FormalTitle} \)?

\[
\begin{align*}
\text{FormalTitle} \\
\text{person! : Person} \\
\text{title! : Title} \\
\text{person!} = \text{Queen} \lor \neg(\text{title!} = \text{YourMajesty})
\end{align*}
\]

(a) \( \text{person!} = \text{King} \)  
(b) \( \text{person!} = \text{PrimeMinister} \)  
(c) \( \neg(\text{person!} = \text{Queen}) \)  
(d) Nothing

Confidence rating:  
□ Not confident  □ Guess  □ Confident

(5) What can you say about the effect of operation \( \text{JoinClub} \) on its after-state variables?

\[
\text{JoinClub} \\
\Delta\text{MembersDB} \\
\text{applicant? : PERSON} \\
\text{members'} = \text{members} \cup \{\text{applicant?}\}
\]

(a) \( \text{members'} = \text{members} \setminus \{\text{applicant?}\} \)  
(b) \( \text{applicant?} \notin \text{members'} \)  
(c) \( \text{applicant?} \notin \text{members'} \)  
(d) Nothing

Confidence rating:  
□ Not confident  □ Guess  □ Confident

(6) Based on its description below, what can you say about the output from operation \( \text{TimeAndDate} \)?

\[
\text{TimeAndDate} \\
\text{time! : TimeSystem} \\
\text{calendar! : CalendarSystem} \\
\neg(\text{time!} = \text{GMT} \lor \text{calendar!} = \text{Julian})
\]

(a) \( \text{time!} = \text{GMT} \)  
(b) \( \neg(\text{time!} = \text{GMT}) \)  
(c) \( \neg(\text{time!} = \text{WET}) \)  
(d) Nothing

Confidence rating:  
□ Not confident  □ Guess  □ Confident
(7) If forecast! = Rain after its execution, what can you say about current! in operation WeatherUpdate?

\[
\begin{align*}
\text{WeatherUpdate} \\
\text{current!}, \text{forecast!} : \text{Weather} \\
\text{current!} = \text{Sunny} \lor \neg (\text{forecast!} = \text{Rain})
\end{align*}
\]

(a) current! = Sunny  
(b) current! = Rain  
(c) \(\neg (\text{current!} = \text{Sunny})\)  
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(8) If \(\neg (\text{operand!} \in \text{ValidOperands})\) after its execution, what can you say about opcode! in operation Assembler?

\[
\begin{align*}
\text{Assembler} \\
\text{opcode!} : \text{Opcode} \\
\text{operand!} : \text{Operand} \\
\text{opcode!} \in \text{ValidOpCodes} \lor \text{operand!} \in \text{ValidOperands}
\end{align*}
\]

(a) opcode! = JUMP  
(b) opcode! ∈ ValidOpCodes  
(c) \(\neg (\text{opcode!} \in \text{ValidOpCodes})\)  
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(9) If card! = Switch after its execution, what can you say about account! in operation Bank?

\[
\begin{align*}
\text{Bank} \\
\text{card!} : \text{CreditCardType} \\
\text{account!} : \text{AccountType} \\
\text{card!} = \text{Switch} \lor \neg (\text{account!} = \text{Shared})
\end{align*}
\]

(a) account! = Shared  
(b) \(\neg (\text{account!} = \text{Shared})\)  
(c) account! = Single  
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(10) What can you say about the effect of operation SecurityCheck on its after-state variables?

\[
\begin{align*}
\text{SecurityCheck} \\
\Delta \text{Door} \\
\Delta \text{AlarmSystem} \\
\text{door_state!} = \text{Locked} \land \neg (\text{alarm_setting!} = \text{Disabled})
\end{align*}
\]

(a) \(\neg (\text{door_state!} = \text{Locked})\)  
(b) alarm_setting! = Disabled  
(c) \(\neg (\text{alarm_setting!} = \text{Disabled})\)  
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident
(11) What can you say about the effect of operation CheckStudentRegister on its after-state variables?

\[ \text{CheckStudentRegister} \]
\[ \Delta \text{Register} \]
\[ \neg (\# \text{register}' > \text{MaxStudents}) \]

(a) \#register' > MaxStudents \land student' \in register'  
(b) \neg (\# \text{register}' > \text{MaxStudents}) \land \neg (\text{student}' \in \text{register}')  
(c) \neg (\text{student}' \in \text{register}')  
(d) \text{Nothing}

Confidence rating: [ ] Not confident [ ] Guess [ ] Confident

(12) What can you say about the effect of operation DisablePrinter on its after-state variables?

\[ \text{DisablePrinter} \]
\[ \Delta \text{Printer} \]
\[ \neg (\text{printer\_status'} = \text{Online}) \]

(a) printer\_status' = Offline  
(b) queue\_status' = Empty  
(c) print\_queue' = () \land \neg (\text{printer\_status'} = \text{Online})  
(d) \text{Nothing}

Confidence rating: [ ] Not confident [ ] Guess [ ] Confident

(13) If \neg (\text{guest}! = \text{VIP}) after its execution, what can you say about room! in operation Hotel?

\[ \text{Hotel} \]
\[ guest! : \text{Customer} \]
\[ room! : \text{Room} \]
\[ \neg (\text{guest}! = \text{VIP}) \lor room! = \text{Single} \]

(a) \neg (room! = \text{Single})  
(b) room! = \text{Single}  
(c) room! = \text{Double}  
(d) \text{Nothing}

Confidence rating: [ ] Not confident [ ] Guess [ ] Confident

(14) If course! = Belfry after its execution, what can you say about prize! in operation GolfTournament?

\[ \text{GolfTournament} \]
\[ course! : \text{Venue} \]
\[ prize! : \text{Prize} \]
\[ \neg (\text{course}! = \text{Belfry}) \lor \text{prize}! = \text{RyderCup} \]

(a) \neg (\text{prize}! = \text{RyderCup})  
(b) prize! = \text{RyderCup}  
(c) prize! = \text{WalkerCup}  
(d) \text{Nothing}

Confidence rating: [ ] Not confident [ ] Guess [ ] Confident
(15) What can you say about the effect of operation \textit{MaximiseWindow} on its after-state variables?

\[
\begin{align*}
\text{MaximiseWindow} \\
\Delta \text{MenuOptions} \\
\Delta \text{ WindowManager}
\end{align*}
\]
\[
\text{selected\_op}' = \text{Maximise} \land \text{window\_coords}' = \text{screen\_coords}
\]

(a) \(\neg(\text{window\_coords}' = \text{screen\_coords})\)  \\
(b) \(\text{selected\_op}' = \text{Maximise}\)  \\
(c) \(\neg(\text{selected\_op}' = \text{Maximise})\)  \\
(d) \text{Nothing}

Confidence rating: □ Not confident □ Guess □ Confident

(16) If \(\neg(\text{password}' = \text{Correct})\) after its execution, what can you say about report' in operation \textit{AccessSystem}?

\[
\begin{align*}
\text{AccessSystem} \\
\text{password}' : \text{Status} \\
\text{report}' : \text{Report}
\end{align*}
\]
\[
\text{password}' = \text{Correct} \lor \neg(\text{report}' = \text{Unauthorised})
\]

(a) \(\text{report}' = \text{Unauthorised}\)  \\
(b) \(\text{report}' = \text{Authorised}\)  \\
(c) \(\neg(\text{report}' = \text{Unauthorised})\)  \\
(d) \text{Nothing}

Confidence rating: □ Not confident □ Guess □ Confident

(17) If \textit{nationality}' = \textit{Greece} after its execution, what can you say about \textit{person}' in operation \textit{Nationality}?

\[
\begin{align*}
\text{Nationality} \\
\text{person}' : \text{Name} \\
\text{nationality}' : \text{Country}
\end{align*}
\]
\[
\text{person}' = \text{Aristotle} \lor \text{nationality}' = \text{Greece}
\]

(a) \(\text{person}' = \text{Aristotle}\)  \\
(b) \(\neg(\text{person}' = \text{Aristotle})\)  \\
(c) \(\text{person}' = \text{Socrates}\)  \\
(d) \text{Nothing}

Confidence rating: □ Not confident □ Guess □ Confident

(18) Based on its description below, what can you say about the output from operation \textit{ArtGallery}?

\[
\begin{align*}
\text{ArtGallery} \\
\text{era}' : \text{Era} \\
\text{painter}' : \text{Person}
\end{align*}
\]
\[
\neg(\text{era}' = \text{Modern} \lor \text{painter}' = \text{VanGogh})
\]

(a) \(\text{painter}' = \text{Klimt}\)  \\
(b) \(\text{painter}' = \text{VanGogh}\)  \\
(c) \(\neg(\text{painter}' = \text{VanGogh})\)  \\
(d) \text{Nothing}

Confidence rating: □ Not confident □ Guess □ Confident
(19) If \( \text{text\_buffer}' = \{ \} \) after the execution of operation \text{CheckTextBuffer}, what can you say about \text{buffer\_status}'? 

\[
\text{CheckTextBuffer} \\
\Delta \text{KeyboardBuffer} \\
\neg(\text{text\_buffer}' = \{ \} \land \text{buffer\_status}' = \text{Empty})
\]

(a) \( \text{buffer\_status}' = \text{Empty} \)  
(b) \( \neg(\text{buffer\_status}' = \text{Empty}) \)  
(c) \( \text{buffer\_status}' = \text{Full} \)  
(d) Nothing 

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident 

(20) If \( \neg(\text{fuel\_system}' = \text{Injection}) \) after its execution, what can you say about \text{stereo}' in operation \text{PrototypeCar}? 

\[
\text{PrototypeCar} \\
\text{stereo}' : \text{ComponentType} \\
\text{fuel\_system}' : \text{FuelSystemType} \\
\neg(\text{stereo}' = \text{CD}) \lor \text{fuel\_system}' = \text{Injection}
\]

(a) \( \neg(\text{stereo}' = \text{CD}) \)  
(b) \( \text{stereo}' = \text{Cassette} \)  
(c) \( \text{stereo}' = \text{CD} \)  
(d) Nothing 

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident 

(21) If \( \text{number\_tills}' = 12 \) after its execution, what can you say about \text{pricing\_system}' in operation \text{Supermarket}? 

\[
\text{Supermarket} \\
\text{pricing\_system}' : \text{System} \\
\text{number\_tills}' : \mathbb{N} \\
\neg(\text{pricing\_system}' = \text{BarCode}) \lor \text{number\_tills}' = 12
\]

(a) \( \text{pricing\_system}' = \text{BarCode} \)  
(b) \( \neg(\text{pricing\_system}' = \text{BarCode}) \)  
(c) \( \text{pricing\_system}' = \text{Labels} \)  
(d) Nothing 

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident 

(22) If \( \text{highlights}' = \text{BBC} \) after its execution, what can you say about \text{live}' in operation \text{EventCoverage}? 

\[
\text{EventCoverage} \\
\text{live}', \text{highlights}' : \text{Organisation} \\
\neg(\text{live}' = \text{Sky}) \lor \neg(\text{highlights}' = \text{BBC})
\]

(a) \( \text{live}' = \text{Sky} \)  
(b) \( \text{live}' = \text{BBC} \)  
(c) \( \neg(\text{live}' = \text{Sky}) \)  
(d) Nothing 

Confidence rating: \( \square \) Not confident \( \square \) Guess \( \square \) Confident
(23) If \( \neg (\text{family!} = \text{Cat}) \) after its execution, what can you say about \( \text{animal!} \) in operation \( \text{Ancestry} \)?

\[
\begin{align*}
\text{Ancestry} & \\
\text{animal!, family! : Animal} & \\
\neg (\text{animal!} = \text{Lion}) \lor \neg (\text{family!} = \text{Cat}) & \\
\end{align*}
\]

(a) \( \text{animal!} = \text{Lion} \)  
(b) \( \neg (\text{animal!} = \text{Lion}) \)  
(c) \( \text{animal!} = \text{Tiger} \)  
(d) \( \text{Nothing} \)

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident

(24) What can you say about the effect of operation \( \text{GuidedMissileCheck} \) on its after-state variables?

\[
\begin{align*}
\text{GuidedMissileCheck} & \\
\Delta \text{Bearings} & \\
\text{target_loc? : COORDS} & \\
\text{current_loc' = target_loc?} & \\
\end{align*}
\]

(a) \( \neg (\text{current_loc'} = \text{target_loc}?) \land \text{mission'} = \text{Failure} \)  
(b) \( \text{current_loc'} = \text{target_loc}? \land \text{mission'} = \text{Success} \)  
(c) \( \neg (\text{current_loc'} = \text{target_loc}?) \)  
(d) \( \text{Nothing} \)

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident

(25) If \( \text{market!} = \text{Japan} \) after its execution, what can you say about \( \text{commodity!} \) in operation \( \text{StocksAndShares} \)?

\[
\begin{align*}
\text{StocksAndShares} & \\
\text{market! : Body} & \\
\text{commodity! : Product} & \\
\neg (\text{market!} = \text{Japan}) \lor \neg (\text{commodity!} = \text{OrangeJuice}) & \\
\end{align*}
\]

(a) \( \neg (\text{commodity!} = \text{CocoaBeans}) \)  
(b) \( \neg (\text{commodity!} = \text{OrangeJuice}) \)  
(c) \( \text{commodity!} = \text{Apples} \)  
(d) \( \text{Nothing} \)

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident

(26) If \( \neg (\text{processor!} = \text{Pentium}) \) after its execution, what can you say about \( \text{display!} \) in operation \( \text{ComputerHardware} \)?

\[
\begin{align*}
\text{ComputerHardware} & \\
\text{processor! : Chip} & \\
\text{display! : Screen} & \\
\neg (\text{processor!} = \text{Pentium}) \lor \neg (\text{display!} = \text{HighResolution}) & \\
\end{align*}
\]

(a) \( \text{display!} = \text{LowResolution} \)  
(b) \( \neg (\text{display!} = \text{HighResolution}) \)  
(c) \( \text{display!} = \text{HighResolution} \)  
(d) \( \text{Nothing} \)

Confidence rating: ☐ Not confident ☐ Guess ☐ Confident
(27) What can you say about the effect of operation ReactorFailure on its after-state variables?

\[ \text{ReactorFailure} \]
\[ \Delta \text{CoolingSystem} \]
\[ \Delta \text{Reactor} \]
\[ \neg(\text{coolertemp}' \leq \text{MaxTemp}) \land \neg(\text{core\_status}' = \text{Safe}) \]

(a) \(\neg(\text{core\_status}' = \text{Safe})\)  
(b) \(\text{coolertemp}' \leq \text{MaxTemp} \land \text{core\_status}' = \text{Safe}\)  
(c) \(\text{core\_status}' = \text{Safe}\)  
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

(28) If line\_status' = Unconnected after the execution of operation ConnectNewUser, what can you say about user'?

\[ \text{ConnectNewUser} \]
\[ \Delta \text{TelephoneNetwork} \]
\[ \neg(\text{user}' \notin \text{ConnectedUsers} \land \text{line\_status}' = \text{Unconnected}) \]

(a) \(\neg(\text{user}' \notin \text{ListedUsers})\)  
(b) \(\neg(\text{user}' \notin \text{ConnectedUsers})\)  
(c) \(\text{user}' \notin \text{ConnectedUsers}\)  
(d) Nothing

Confidence rating: □ Not confident □ Guess □ Confident

Thank you for participating in this experiment.

If you would like to know your score please write your e-mail address here: ..................................................

Please return completed forms to:

Rick Vinter. School of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Experiment 4: Quantified Reasoning  
(Abstract Formal Logic Group)

Computer Science Research Experiment:  
Reasoning About Formal Software Specifications.

About Yourself

Occupation: Staff / Student / Other (please specify): ......................... Age: ....
University / Organisation: ......................... Course (if applicable): .................
Z experience: ... years ... months  Z expertise rating:  Novice / Proficient / Expert
Other formal notations known (e.g. CCS, CSP, VDM):  .........................
Types of logic studied (e.g. propositional and predicate calculus, Boolean algebra): ....

--------------------------

Before We Begin

The Z expression $\forall t : T \cdot A(t) \Rightarrow B(t)$ corresponds most closely to which one of the following English translations?

(a) All As are Bs
(b) At least one (possibly all) As are Bs
(c) Possibly all As are Bs
(d) Some As are Bs

The Z expression $\exists t : T \cdot A(t) \wedge B(t)$ corresponds most closely to which one of the following English translations?

(a) At least one A is a B
(b) At least one (possibly all) As are Bs
(c) Exactly one A is a B
(d) Some As are Bs

The Z expression $\exists t : T \cdot A(t) \wedge \neg B(t)$ corresponds most closely to which one of the following English translations?

(a) At least one A is not a B
(b) At least one (possibly all) As are not Bs
(c) Exactly one A is not a B
(d) Some As are not Bs

The Z expression $\neg \exists t : T \cdot A(t) \wedge B(t)$ corresponds most closely to which one of the following English translations?

(a) None of the As are Bs
(b) At least one (possibly none) of the As are Bs
(c) Possibly none of the As are Bs
(d) Exactly one A is not a B

Instructions

In each of the tasks that follow, you will be shown two Z predicate expressions taken from an operational schema. You may assume that all of the named functions have been defined. You are asked to determine which one of four given statements follows logically from the information given. Please circle the letter of your choice. You will then be asked to give a confidence rating, which should indicate how far you believe your answer to be correct. Please complete all tasks to the best of your ability, without reference to textbooks. The experiment should take around 30 minutes to complete.
Tasks

(1) \( \forall x : X \cdot B(x) \Rightarrow C(x) \)
\( \forall x : X \cdot A(x) \Rightarrow B(x) \)

(a) \( \exists x : X \cdot A(x) \land C(x) \)
(b) \( \forall x : X \cdot A(x) \Rightarrow C(x) \)
(c) \( \neg \exists x : X \cdot A(x) \land C(x) \)
(d) No valid conclusion

(2) \( \neg \exists x : X \cdot A(x) \land B(x) \)
\( \forall x : X \cdot B(x) \Rightarrow C(x) \)

(a) \( \exists x : X \cdot C(x) \land \neg A(x) \)
(b) \( \forall x : X \cdot A(x) \Rightarrow C(x) \)
(c) \( \forall x : X \cdot C(x) \Rightarrow B(x) \)
(d) No valid conclusion

(3) \( \forall x : X \cdot A(x) \Rightarrow B(x) \)
\( \forall x : X \cdot B(x) \Rightarrow C(x) \)

(a) \( \forall x : X \cdot C(x) \Rightarrow A(x) \)
(b) \( \forall x : X \cdot A(x) \land A(x) \)
(c) \( \forall x : X \cdot B(x) \Rightarrow C(x) \)
(d) No valid conclusion

(4) \( \exists x : X \cdot B(x) \land C(x) \)
\( \exists x : X \cdot A(x) \land B(x) \)

(a) \( \exists x : X \cdot A(x) \land C(x) \)
(b) \( \forall x : X \cdot B(x) \land A(x) \)
(c) \( \forall x : X \cdot A(x) \Rightarrow C(x) \)
(d) No valid conclusion

(5) \( \exists x : X \cdot B(x) \land A(x) \)
\( \exists x : X \cdot B(x) \land C(x) \)

(a) \( \exists x : X \cdot C(x) \land A(x) \)
(b) \( \exists x : X \cdot C(x) \land A(x) \)
(c) \( \exists x : X \cdot B(x) \land \neg A(x) \)
(d) No valid conclusion

(6) \( \exists x : X \cdot A(x) \land B(x) \)
\( \exists x : X \cdot A(x) \land C(x) \)

(a) \( \exists x : X \cdot C(x) \land A(x) \)
(b) \( \exists x : X \cdot C(x) \land A(x) \)
(c) \( \forall x : X \cdot C(x) \Rightarrow A(x) \)
(d) No valid conclusion

(7) \( \forall x : X \cdot B(x) \Rightarrow C(x) \)
\( \exists x : X \cdot A(x) \land B(x) \)

(a) \( \neg \exists x : X \cdot A(x) \land C(x) \)
(b) \( \exists x : X \cdot A(x) \land C(x) \)
(c) \( \forall x : X \cdot A(x) \Rightarrow C(x) \)
(d) No valid conclusion

(8) \( \forall x : X \cdot A(x) \Rightarrow B(x) \)
\( \neg \exists x : X \cdot B(x) \land C(x) \)

(a) \( \exists x : X \cdot C(x) \land \neg A(x) \)
(b) \( \exists x : X \cdot C(x) \land A(x) \)
(c) \( \exists x : X \cdot C(x) \land A(x) \)
(d) No valid conclusion

(9) \( \exists x : X \cdot A(x) \land B(x) \)
\( \exists x : X \cdot B(x) \land C(x) \)

(a) \( \forall x : X \cdot C(x) \Rightarrow A(x) \)
(b) \( \exists x : X \cdot C(x) \land A(x) \)
(c) \( \exists x : X \cdot C(x) \land A(x) \)
(d) No valid conclusion

(10) \( \neg \exists x : X \cdot B(x) \land A(x) \)
\( \exists x : X \cdot C(x) \land B(x) \)

(a) \( \forall x : X \cdot C(x) \Rightarrow \neg A(x) \)
(b) \( \exists x : X \cdot C(x) \land A(x) \)
(c) \( \exists x : X \cdot C(x) \land \neg A(x) \)
(d) No valid conclusion

(11) \( \exists x : X \cdot A(x) \land B(x) \)
\( \exists x : X \cdot B(x) \land C(x) \)

(a) \( \forall x : X \cdot C(x) \land A(x) \)
(b) \( \exists x : X \cdot B(x) \land C(x) \)
(c) \( \exists x : X \cdot C(x) \land A(x) \)
(d) No valid conclusion

(12) \( \exists x : X \cdot B(x) \land \neg A(x) \)
\( \exists x : X \cdot B(x) \land \neg C(x) \)

(a) \( \exists x : X \cdot C(x) \land \neg A(x) \)
(b) \( \exists x : X \cdot A(x) \land \neg C(x) \)
(c) \( \forall x : X \cdot C(x) \Rightarrow A(x) \)
(d) No valid conclusion
(13) \forall x : X \cdot A(x) \Rightarrow B(x)
\neg \exists x : X \cdot C(x) \land B(x)
(a) \exists x : X \cdot C(x) \land A(x)
(b) \neg \exists x : X \cdot C(x) \land \neg A(x)
(c) \exists x : X \cdot C(x) \land \neg A(x)
(d) No valid conclusion

\[ \square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident} \]

(14) \neg \exists x : X \cdot B(x) \land \neg A(x)
\exists x : X \cdot B(x) \land C(x)
(a) \exists x : X \cdot C(x) \land \neg A(x)
(b) \neg \exists x : X \cdot C(x) \land A(x)
(c) \exists x : X \cdot C(x) \land A(x)
(d) No valid conclusion

\[ \square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident} \]

(15) \neg \exists x : X \cdot A(x) \land B(x)
\neg \exists x : X \cdot B(x) \land C(x)
(a) \forall x : X \cdot C(x) \Rightarrow A(x)
(b) \neg \exists x : X \cdot C(x) \land \neg A(x)
(c) \exists x : X \cdot C(x) \land \neg A(x)
(d) No valid conclusion

\[ \square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident} \]

(16) \neg \exists x : X \cdot A(x) \land B(x)
\forall x : X \cdot C(x) \Rightarrow B(x)
(a) \exists x : X \cdot C(x) \land A(x)
(b) \neg \exists x : X \cdot C(x) \land \neg A(x)
(c) \forall x : X \cdot C(x) \Rightarrow B(x)
(d) No valid conclusion

\[ \square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident} \]

(17) \forall x : X \cdot A(x) \Rightarrow B(x)
\forall x : X \cdot B(x) \Rightarrow C(x)
(a) \exists x : X \cdot C(x) \land A(x)
(b) \neg \exists x : X \cdot A(x) \land \neg A(x)
(c) \forall x : X \cdot C(x) \Rightarrow B(x)
(d) No valid conclusion

\[ \square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident} \]

(18) \exists x : X \cdot B(x) \land \neg A(x)
\forall x : X \cdot B(x) \Rightarrow C(x)
(a) \exists x : X \cdot C(x) \land \neg A(x)
(b) \exists x : X \cdot B(x) \land A(x)
(c) \forall x : X \cdot C(x) \Rightarrow A(x)
(d) No valid conclusion

\[ \square \text{ Not confident} \quad \square \text{ Guess} \quad \square \text{ Confident} \]
(25) $\exists x : X \bullet B(x) \land C(x)$  
$\forall x : X \bullet B(x) \Rightarrow A(x)$  
(a) $\exists x : X \bullet A(x) \land C(x)$  
(b) $\exists x : X \bullet A(x) \land \neg C(x)$  
(c) $\forall x : X \bullet A(x) \Rightarrow C(x)$  
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(28) $\forall x : X \bullet B(x) \Rightarrow C(x)$  
$\exists x : X \bullet B(x) \land A(x)$  
(a) $\forall x : X \bullet A(x) \Rightarrow C(x)$  
(b) $\neg \exists x : X \bullet A(x) \land C(x)$  
(c) $\exists x : X \bullet A(x) \land C(x)$  
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(26) $\neg \exists x : X \bullet B(x) \land C(x)$  
$\forall x : X \bullet B(x) \Rightarrow A(x)$  
(a) $\neg \exists x : X \bullet A(x) \land C(x)$  
(b) $\forall x : X \bullet A(x) \Rightarrow B(x)$  
(c) $\exists x : X \bullet A(x) \land \neg C(x)$  
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(29) $\exists x : X \bullet A(x) \land B(x)$  
$\forall x : X \bullet B(x) \Rightarrow C(x)$  
(a) $\exists x : X \bullet C(x) \land A(x)$  
(b) $\neg \exists x : X \bullet C(x) \land A(x)$  
(c) $\exists x : X \bullet B(x) \land \neg A(x)$  
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(27) $\forall x : X \bullet C(x) \Rightarrow B(x)$  
$\exists x : X \bullet A(x) \land \neg B(x)$  
(a) $\neg \exists x : X \bullet A(x) \land C(x)$  
(b) $\exists x : X \bullet A(x) \land \neg C(x)$  
(c) $\forall x : X \bullet A(x) \Rightarrow C(x)$  
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(30) $\neg \exists x : X \bullet B(x) \land C(x)$  
$\forall x : X \bullet A(x) \Rightarrow B(x)$  
(a) $\neg \exists x : X \bullet A(x) \land C(x)$  
(b) $\exists x : X \bullet A(x) \land \neg C(x)$  
(c) $\forall x : X \bullet B(x) \Rightarrow A(x)$  
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

Thank you for participating in this experiment.

If you would like to know your test score, please write your e-mail address here:

..................................................................................................................

Please return completed forms to:

Rick Vinter. Faculty of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Experiment 4: Quantified Reasoning  
(Thematic Formal Logic Group)  

Computer Science Research Experiment: 
Reasoning About Formal Software Specifications.

About Yourself

Occupation: Staff / Student / Other (please specify): .................  Age: ....
University / Organisation: ..................... Course (if applicable): ..................
Z experience: ... years ... months  Z expertise rating:  Novice / Proficient / Expert
Other formal notations known (e.g. CCS, CSP, VDM): ..............................
Types of logic studied (e.g. propositional and predicate calculus, Boolean algebra): ....

Before We Begin

The Z expression $\forall t : T \bullet A(t) \Rightarrow B(t)$ corresponds most closely to which one of the following English translations?

(a) All As are Bs  
(b) At least one (possibly all) As are Bs  
(c) Possibly all As are Bs  
(d) Some As are Bs

The Z expression $\exists t : T \bullet A(t) \land B(t)$ corresponds most closely to which one of the following English translations?

(a) At least one A is a B  
(b) At least one (possibly all) As are Bs  
(c) Exactly one A is a B  
(d) Some As are Bs

The Z expression $\exists t : T \bullet A(t) \land \neg B(t)$ corresponds most closely to which one of the following English translations?

(a) At least one A is not a B  
(b) At least one (possibly all) As are not Bs  
(c) Exactly one A is not a B  
(d) Some As are not Bs

The Z expression $\neg \exists t : T \bullet A(t) \land B(t)$ corresponds most closely to which one of the following English translations?

(a) None of the As are Bs  
(b) At least one (possibly none) of the As are Bs  
(c) Possibly none of the As are Bs  
(d) Exactly one A is not a B

Instructions

In each of the tasks that follow, you will be shown two Z predicate expressions taken from an operational schema. You may assume that all of the named functions have been defined. You are asked to determine which one of four given statements follows logically from the information given. Please circle the letter of your choice. You will then be asked to give a confidence rating, which should indicate how far you believe your answer to be correct. Please complete all tasks to the best of your ability, without reference to textbooks. The experiment should take around 40 minutes to complete.
Tasks

(1) ∀p : Person • human(p) ⇒ mortal(p)
∀p : Person • Greek(p) ⇒ human(p)
(7) ∃f : Food • edible(f) ∧ vegetable(f)
¬∃f : Food • vegetable(f) ∧ mineral(f)
(a) ∃f : Food • mineral(f) ∧ edible(f)
(b) ¬∃f : Food • mineral(f) ∧ edible(f)
(c) ∀f : Food • mineral(f) ⇒ edible(f)
(d) No valid conclusion

(2) ¬∃f : Food • orange(f) ∧ apple(f)
∀f : Food • apple(f) ⇒ fruit(f)
(a) ∃f : Food • fruit(f) ∧ ¬orange(f)
(b) ∀f : Food • orange(f) ⇒ fruit(f)
(c) ∀f : Food • fruit(f) ⇒ apple(f)
(d) No valid conclusion

(3) ∀p : Person • hard_worker(p) ⇒ rewarded(p)
∃p : Person • student(p) ∧ hard_worker(p)
(a) ¬∃p : Person • student(p) ∧ rewarded(p)
(b) ∃p : Person • student(p) ∧ rewarded(p)
(c) ∀p : Person • student(p) ⇒ rewarded(p)
(d) No valid conclusion

(4) ∀p : Person • computer_literate(p) ⇒ mathematician(p)
∀p : Person • programmer(p) ⇒ mathematician(p)
(a) ∀p : Person • programmer(p) ⇒ computer_literate(p)
(b) ¬∃p : Person • programmer(p) ∧ computer_literate(p)
(c) ∀p : Person • mathematician(p) ⇒ programmer(p)
(d) No valid conclusion

(5) ∃p : Person • smoker(p) ∧ rational(p)
¬∃p : Person • doctor(p) ∧ smoker(p)
(a) ¬∃p : Person • doctor(p) ∧ rational(p)
(b) ∃p : Person • smoker(p) ∧ doctor(p)
(c) ∀p : Person • doctor(p) ⇒ rational(p)
(d) No valid conclusion

(6) ∃p : Person • athlete(p) ∧ professional(p)
∃p : Person • athlete(p) ∧ amateur(p)
(a) ∃p : Person • amateur(p) ∧ professional(p)
(b) ¬∃p : Person • amateur(p) ∧ professional(p)
(c) ∃p : Person • athlete(p) ∧ ¬professional(p)
(d) No valid conclusion

(8) ∃p : Person • human(p) ∧ omnivore(p)
¬∃p : Person • vegetarian(p) ∧ omnivore(p)
(a) ¬∃p : Person • vegetarian(p) ∧ human(p)
(b) ∃p : Person • omnivore(p) ∧ vegetarian(p)
(c) ∃p : Person • vegetarian(p) ∧ human(p)
(d) No valid conclusion

(9) ∀p : Person • criminal(p) ⇒ deceitful(p)
¬∃p : Person • vicar(p) ∧ deceitful(p)
(a) ∃p : Person • vicar(p) ∧ criminal(p)
(b) ¬∃p : Person • vicar(p) ∧ criminal(p)
(c) ∃p : Person • vicar(p) ∧ ¬criminal(p)
(d) No valid conclusion

(10) ∃p : Person • movie_star(p) ∧ wealthy(p)
∃p : Person • wealthy(p) ∧ supermodel(p)
(a) ∀p : Person • supermodel(p) ⇒ movie_star(p)
(b) ∃p : Person • supermodel(p) ∧ movie_star(p)
(c) ¬∃p : Person • supermodel(p) ∧ movie_star(p)
(d) No valid conclusion

(11) ∃p : Person • novelist(p) ∧ poet(p)
∃p : Person • author(p) ∧ novelist(p)
(a) ∀p : Person • author(p) ⇒ ¬poet(p)
(b) ¬∃p : Person • author(p) ∧ poet(p)
(c) ∃p : Person • author(p) ∧ ¬poet(p)
(d) No valid conclusion

(12) ∀b : Being • God(b) ⇒ merciful(b)
¬∃b : Being • merciful(b) ∧ unforgiving(b)
(a) ∃b : Being • unforgiving(b) ∧ ¬God(b)
(b) ¬∃b : Being • unforgiving(b) ∧ God(b)
(c) ∃b : Being • unforgiving(b) ∧ God(b)
(d) No valid conclusion
(13) \( \exists p: \text{Person} \cdot \text{scientist}(p) \land \text{methodical}(p) \)
\( \neg \exists p: \text{Person} \cdot \text{drunkard}(p) \land \text{methodical}(p) \)

(a) \( \neg \exists p: \text{Person} \cdot \text{drunkard}(p) \land \text{scientist}(p) \)
(b) \( \exists p: \text{Person} \cdot \text{methodical}(p) \land \text{drunkard}(p) \)
(c) \( \exists p: \text{Person} \cdot \text{drunkard}(p) \land \text{scientist}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(14) \( \exists d: \text{Drug} \cdot \text{tablet}(d) \land \neg \text{harmful}(d) \)
\( \exists d: \text{Drug} \cdot \text{tablet}(d) \land \neg \text{prescribed}(d) \)

(a) \( \exists d: \text{Drug} \cdot \text{prescribed}(d) \land \neg \text{harmful}(d) \)
(b) \( \exists d: \text{Drug} \cdot \text{harmful}(d) \land \neg \text{prescribed}(d) \)
(c) \( \forall d: \text{Drug} \cdot \text{prescribed}(d) \Rightarrow \text{harmful}(d) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(15) \( \neg \exists m: \text{Material} \cdot \text{metal}(m) \land \text{wood}(m) \)
\( \exists m: \text{Material} \cdot \text{metal}(m) \land \text{conductor}(m) \)

(a) \( \exists m: \text{Material} \cdot \text{conductor}(m) \land \neg \text{wood}(m) \)
(b) \( \neg \exists m: \text{Material} \cdot \text{conductor}(m) \land \text{wood}(m) \)
(c) \( \exists m: \text{Material} \cdot \text{conductor}(m) \land \text{wood}(m) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(16) \( \neg \exists p: \text{Person} \cdot \text{biased}(p) \land \text{judge}(p) \)
\( \neg \exists p: \text{Person} \cdot \text{judge}(p) \land \text{commentator}(p) \)

(a) \( \forall p: \text{Person} \cdot \text{commentator}(p) \Rightarrow \text{biased}(p) \)
(b) \( \neg \exists p: \text{Person} \cdot \text{commentator}(p) \land \text{biased}(p) \)
(c) \( \exists p: \text{Person} \cdot \text{commentator}(p) \land \text{biased}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(17) \( \neg \exists p: \text{Person} \cdot \text{American}(p) \land \text{British}(p) \)
\( \forall p: \text{Person} \cdot \text{English}(p) \Rightarrow \text{British}(p) \)

(a) \( \neg \exists p: \text{Person} \cdot \text{English}(p) \land \text{American}(p) \)
(b) \( \exists p: \text{Person} \cdot \text{English}(p) \land \neg \text{American}(p) \)
(c) \( \forall p: \text{Person} \cdot \text{British}(p) \Rightarrow \text{English}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(18) \( \forall p: \text{Person} \cdot \text{bank_manager}(p) \Rightarrow \text{responsible}(p) \)
\( \forall p: \text{Person} \cdot \text{responsible}(p) \Rightarrow \text{trustworthy}(p) \)

(a) \( \exists p: \text{Person} \cdot \text{trustworthy}(p) \land \text{bank_manager}(p) \)
(b) \( \neg \exists p: \text{Person} \cdot \text{bank_manager}(p) \land \text{responsible}(p) \)
(c) \( \forall p: \text{Person} \cdot \text{trustworthy}(p) \Rightarrow \text{responsible}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(19) \( \exists v: \text{Vehicle} \cdot \text{train}(v) \land \neg \text{punctual}(v) \)
\( \forall v: \text{Vehicle} \cdot \text{train}(v) \Rightarrow \text{public_transport}(v) \)

(a) \( \exists v: \text{Vehicle} \cdot \text{public_transport}(v) \land \neg \text{punctual}(v) \)
(b) \( \exists v: \text{Vehicle} \cdot \text{train}(v) \land \text{punctual}(v) \)
(c) \( \forall v: \text{Vehicle} \cdot \text{public_transport}(v) \Rightarrow \text{punctual}(v) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(20) \( \exists v: \text{Vertebrate} \cdot \text{bird}(v) \land \neg \text{white}(v) \)
\( \forall v: \text{Vertebrate} \cdot \text{owl}(v) \Rightarrow \text{bird}(v) \)

(a) \( \exists v: \text{Vertebrate} \cdot \text{owl}(v) \land \neg \text{white}(v) \)
(b) \( \neg \exists v: \text{Vertebrate} \cdot \text{owl}(v) \land \text{white}(v) \)
(c) \( \exists v: \text{Vertebrate} \cdot \text{owl}(v) \land \text{white}(v) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(21) \( \neg \exists p: \text{Person} \cdot \text{coward}(p) \land \text{brave}(p) \)
\( \exists p: \text{Person} \cdot \text{bodyguard}(p) \land \text{brave}(p) \)

(a) \( \forall p: \text{Person} \cdot \text{bodyguard}(p) \Rightarrow \text{brave}(p) \)
(b) \( \exists p: \text{Person} \cdot \text{bodyguard}(p) \land \neg \text{coward}(p) \)
(c) \( \neg \exists p: \text{Person} \cdot \text{bodyguard}(p) \land \text{coward}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(22) \( \exists p: \text{Person} \cdot \text{capitalist}(p) \land \text{Russian}(p) \)
\( \forall p: \text{Russian}(p) \Rightarrow \text{communist}(p) \)

(a) \( \exists p: \text{Person} \cdot \text{communist}(p) \land \text{capitalist}(p) \)
(b) \( \neg \exists p: \text{Person} \cdot \text{communist}(p) \land \text{capitalist}(p) \)
(c) \( \exists p: \text{Person} \cdot \text{Russian}(p) \land \neg \text{capitalist}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(23) \( \neg \exists p: \text{Person} \cdot \text{disloyal}(p) \land \text{married}(p) \)
\( \exists p: \text{Person} \cdot \text{married}(p) \land \text{traitor}(p) \)

(a) \( \exists p: \text{Person} \cdot \text{traitor}(p) \land \neg \text{married}(p) \)
(b) \( \neg \exists p: \text{Person} \cdot \text{disloyal}(p) \land \text{traitor}(p) \)
(c) \( \exists p: \text{Person} \cdot \text{traitor}(p) \land \neg \text{disloyal}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(24) \( \forall p: \text{Person} \cdot \text{bribe_taker}(p) \Rightarrow \text{criminal}(p) \)
\( \exists p: \text{Person} \cdot \text{bribe_taker}(p) \land \text{policeman}(p) \)

(a) \( \forall p: \text{Person} \cdot \text{policeman}(p) \Rightarrow \text{criminal}(p) \)
(b) \( \neg \exists p: \text{Person} \cdot \text{policeman}(p) \land \text{criminal}(p) \)
(c) \( \exists p: \text{Person} \cdot \text{policeman}(p) \land \text{criminal}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident
(25) $\exists p : Person \bullet pacifist(p) \land patriotic(p)$
    $\neg \exists p : Person \bullet soldier(p) \land pacifist(p)$

(a) $\exists p : Person \bullet soldier(p) \land patriotic(p)$
(b) $\exists p : Person \bullet pacifist(p) \land soldier(p)$
(c) $\forall p : Person \bullet soldier(p) \Rightarrow patriotic(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(26) $\exists p : Person \bullet child(p) \land \neg adult(p)$
    $\exists p : Person \bullet adult(p) \land \neg driver(p)$

(a) $\exists p : Person \bullet driver(p) \land \neg child(p)$
(b) $\exists p : Person \bullet driver(p) \land child(p)$
(c) $\forall p : Person \bullet driver(p) \Rightarrow adult(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(27) $\forall p : Person \bullet introvert(p) \Rightarrow timid(p)$
    $\exists p : Person \bullet timid(p) \land \neg librarian(p)$

(a) $\exists p : Person \bullet librarian(p) \land introvert(p)$
(b) $\forall p : Person \bullet librarian(p) \Rightarrow introvert(p)$
(c) $\neg \exists p : Person \bullet librarian(p) \land introvert(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(28) $\exists a : Activity \bullet sport(a) \land olympic\_event(a)$
    $\exists a : Activity \bullet sport(a) \land safe(a)$

(a) $\exists a : Activity \bullet safe(a) \land olympic\_event(a)$
(b) $\forall a : Activity \bullet safe(a) \land olympic\_event(a)$
(c) $\exists a : Activity \bullet sport(a) \land \neg olympic\_event(a)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(29) $\forall p : Person \bullet bachelor(p) \Rightarrow unmarried(p)$
    $\forall p : Person \bullet bachelor(p) \Rightarrow male(p)$

(a) $\forall p : Person \bullet male(p) \Rightarrow bachelor(p)$
(b) $\forall p : Person \bullet male(p) \Rightarrow unmarried(p)$
(c) $\exists p : Person \bullet male(p) \land unmarried(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(30) $\exists p : Person \bullet athlete(p) \land runner(p)$
    $\forall p : Person \bullet athlete(p) \Rightarrow healthy(p)$

(a) $\exists p : Person \bullet healthy(p) \land runner(p)$
(b) $\exists p : Person \bullet healthy(p) \land \neg runner(p)$
(c) $\forall p : Person \bullet healthy(p) \Rightarrow runner(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(31) $\neg \exists p : Person \bullet churchgoer(p) \land atheist(p)$
    $\forall p : Person \bullet churchgoer(p) \Rightarrow devout(p)$

(a) $\neg \exists p : Person \bullet devout(p) \land atheist(p)$
(b) $\forall p : Person \bullet devout(p) \Rightarrow churchgoer(p)$
(c) $\exists p : Person \bullet devout(p) \land \neg atheist(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(32) $\forall p : Person \bullet honest(p) \Rightarrow hard\_worker(p)$
    $\exists p : Person \bullet politician(p) \land \neg hard\_worker(p)$

(a) $\neg \exists p : Person \bullet politician(p) \land honest(p)$
(b) $\exists p : Person \bullet politician(p) \land \neg honest(p)$
(c) $\forall p : Person \bullet politician(p) \Rightarrow honest(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(33) $\forall p : Person \bullet poor(p) \Rightarrow unlucky(p)$
    $\exists p : Person \bullet poor(p) \land gambler(p)$

(a) $\forall p : Person \bullet gambler(p) \Rightarrow unlucky(p)$
(b) $\neg \exists p : Person \bullet gambler(p) \land unlucky(p)$
(c) $\exists p : Person \bullet gambler(p) \land unlucky(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(34) $\neg \exists v : Vehicle \bullet car(v) \land boat(v)$
    $\exists v : Vehicle \bullet boat(v) \land aerodynamic(v)$

(a) $\exists v : Vehicle \bullet aerodynamic(v) \land \neg boat(v)$
(b) $\neg \exists v : Vehicle \bullet car(v) \land aerodynamic(v)$
(c) $\exists v : Vehicle \bullet aerodynamic(v) \land \neg car(v)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(35) $\forall p : Person \bullet Tory\_voter(p) \Rightarrow Conservative(p)$
    $\exists p : Person \bullet Conservative(p) \land \neg Labourite(p)$

(a) $\exists p : Person \bullet Labourite(p) \land Tory\_voter(p)$
(b) $\forall p : Person \bullet Labourite(p) \Rightarrow Tory\_voter(p)$
(c) $\neg \exists p : Person \bullet Labourite(p) \land Tory\_voter(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(36) $\forall p : Person \bullet home\_owner(p) \Rightarrow married(p)$
    $\exists p : Person \bullet bachelor(p) \land home\_owner(p)$

(a) $\neg \exists p : Person \bullet bachelor(p) \land married(p)$
(b) $\exists p : Person \bullet bachelor(p) \land married(p)$
(c) $\forall p : Person \bullet bachelor(p) \Rightarrow married(p)$
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident
(37) \( \exists p: \text{Person} \land \text{footballer}(p) \land \neg \text{healthy}(p) \land \forall p: \text{Person} \land \text{footballer}(p) \Rightarrow \text{athlete}(p) \)

(a) \( \exists p: \text{Person} \land \text{athlete}(p) \land \neg \text{healthy}(p) \)
(b) \( \exists p: \text{Person} \land \text{footballer}(p) \land \text{healthy}(p) \)
(c) \( \forall p: \text{Person} \land \text{athlete}(p) \Rightarrow \text{healthy}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(38) \( \exists p: \text{Person} \land \text{homeless}(p) \land \text{beggar}(p) \land \forall p: \text{Person} \land \text{beggar}(p) \Rightarrow \text{poor}(p) \)

(a) \( \exists p: \text{Person} \land \text{poor}(p) \land \text{homeless}(p) \)
(b) \( \neg \exists p: \text{Person} \land \text{poor}(p) \land \text{homeless}(p) \)
(c) \( \exists p: \text{Person} \land \text{beggar}(p) \land \text{homeless}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(39) \( \exists a: \text{Animal} \land \text{mammal}(a) \land \neg \text{loyal}(a) \land \forall a: \text{Animal} \land \text{dog}(a) \Rightarrow \text{mammal}(a) \)

(a) \( \exists a: \text{Animal} \land \text{dog}(a) \land \neg \text{loyal}(a) \)
(b) \( \neg \exists a: \text{Animal} \land \text{dog}(a) \land \text{loyal}(a) \)
(c) \( \exists a: \text{Animal} \land \text{dog}(a) \land \text{loyal}(a) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

(40) \( \neg \exists p: \text{Person} \land \text{millionaire}(p) \land \text{poor}(p) \land \forall p: \text{Person} \land \text{rich}(p) \Rightarrow \text{millionaire}(p) \)

(a) \( \neg \exists p: \text{Person} \land \text{rich}(p) \land \text{poor}(p) \)
(b) \( \exists p: \text{Person} \land \text{rich}(p) \land \neg \text{poor}(p) \)
(c) \( \forall p: \text{Person} \land \text{millionaire}(p) \Rightarrow \text{rich}(p) \)
(d) No valid conclusion

☐ Not confident ☐ Guess ☐ Confident

Thank you for participating in this experiment.

If you would like to know your test score, please write your e-mail address here:

............................................................

Please return completed forms to:

Rick Vinter. Faculty of Information Sciences, University of Hertfordshire, College Lane, Hatfield, Herts. AL10 9AB
Appendix B

Related Publications
Related Publications

This appendix describes in chronological order the various published papers and reports that have been produced as a result of this research programme.


This report describes the review that was used to identify a suitable grammatical framework within which to conduct cognitive experiments and to formulate metrics. It was this review which resulted in the decision to use the Z notation.


This report describes the initial exploratory investigation which was used to refine the original research aims and which influenced the design of the three main formalised studies.


Two seminal publications, Bowen and Hinchey (1994) and Hall (1990), each aim to dispel seven popular misconceptions associated with formal methods and to describe the possible benefits of formalisation. Based on empirical evidence from the initial study, this report presents seven reasons why formal methods might not necessarily lead to some of their commonly purported benefits.

This report documents the status of the research programme at the MPhil/PhD transfer stage and was submitted in partial fulfillment of the transfer requirements. It discusses the empirical studies which were complete, those in progress, and those planned studies whose designs had yet to be finalised at the transfer stage.


This report and joint symposium paper discuss the main findings from the initial investigation along with several preliminary findings from the formalised study of conditional reasoning. The implications of these findings are discussed in relation to the design of safety critical systems.


This report describes the main formalised study of conditional reasoning. The results point to a range of non-logical conditional reasoning errors which users are liable to commit in formalised contexts.

This document reports the main formalised study of disjunctive and conjunctive reasoning. Its results suggest that users of formal methods are often logical in reasoning about disjunctive and conjunctive statements in formalised contexts, but are liable to commit non-logical errors.


This document reports the main formalised study of quantified reasoning. Its results suggest that users of formal methods are liable to apply non-logical everyday reasoning heuristics when reasoning about categorically quantified statements in formalised contexts, similar to those exhibited in cognitive studies of syllogistic reasoning.


This report describes how the empirical data generated during the three main formalised studies was synthesised into a system of metrics for identifying potential sources of human reasoning difficulty in formal specifications.


This paper focuses on the way in which a descriptive model was formulated in terms of results obtained from the empirical study of conditional reasoning. It demonstrates how the model might be applied in software engineering contexts.