# Prototype Decision Support System for a Differential Diagnosis of Psychotic, Mood, and Organic Mental Disorders: Part II

Robert T. Plant, Stephen Murrell and Henry R. Moreno

*Med Decis Making* 1994; 14; 273
DOI: 10.1177/0272989X9401400310

# Prototype Decision Support System for a Differential Diagnosis of Psychotic, Mood, and Organic Mental Disorders:

## Part II

ROBERT T. PLANT, PhD, STEPHEN MURRELL, DPhil,
HENRY R. MORENO, PhD

This paper presents a prototype parallel-processing decision support system, based on the ALICE graph reduction machine, for the differential diagnosis of psychotic, mood, and organic mental disorders in accordance with the American Psychiatric Association's revised third edition of the *Diagnostic and Statistical Manual of Mental Disorders*. The paper extends the authors' earlier work where the same domain was implemented on an expert-system shell using a rule-based representation. *Key words*: mental disorders; decision support system; parallel processing, artificial intelligence. **(Med Decis Making 1994;14:273–288)**

In this paper we present a parallel-processing approach to the implementation of a decision support system for the differential diagnosis of psychotic, mood, and organic mental disorders, as shown in the guidelines of the revised third edition of the American Psychiatric Association *Diagnosis and Statistical Manual of Mental Disorders (DSM-III-R)* [It should be noted that DSM-III-R is a transitional version of the manual, which is to be superseded by DSM-IV; we use DSM-III-R because at the time of writing it is the most widely used version. The techniques shown in this paper are not dependent upon the details of the manual.]

The paper extends the earlier work of the authors in developing knowledge-based systems for the differential diagnosis of three separate but interconnected types of mental aberrations: 1) psychotic symptoms, 2) mood disturbances, and 3) organic mental disturbances.[1] *Psychotic symptoms* "include gross impairment in reality testing as evidenced by delusions, hallucinations, incoherence or marked loosening of associations, catatonic stupor or excitement, or grossly disorganized behavior."[2] *Mood disturbances* "are marked by a persistently depressed, elevated, expansive, or irritable mood."[2] *Organic mental disorders* "arise from a specific organic factor that is judged to be etiologically related to the disturbance or, in the case of Dementia and Delirium, all non-organic disorders that could account for symptoms have been eliminated."[2] Because several disorders may fit the individual diagnostic picture, a list of possible diagnoses is given; hence the term "differential diagnosis."

The goal of this paper is to extend the techniques developed in the creation of a rule-based production-system prototype implementation of psychotic, mood, and organic mental disorders, following the flowcharts given in DSM-III-R and reduced into the minimal decision table forms, as presented by Moreno and Plant.[1] We give an alternative implementation for the same problem, on a graph-reduction ALICE* machine,[3,4] a parallel-processing environment that demonstrates several advantages over the sequential nature of the shell-based approach presented in our earlier paper[1] and similar sequential implementations by other researchers.[5,6] We introduce the concept of graph reduction and parallel processing. This is then developed through a small example in which the diagnosis of a subset of mood disturbances is implemented using the graph-reduction technique. The following section discusses a large-scale implementation for all three aspects of the DSM-III-R flowcharts, focusing upon the advantages and disadvantages of such an approach. Extensions to the use of parallel processing in psychiatric and medical systems and the implications of these techniques are then presented.

It should be noted that our goal in developing the graph-reduction parallel-processing implementation of the decision support system was to illustrate the possibilities for parallel-processing technologies in the

Received September 8, 1993, from the Department of Computer Information Systems (RTP, HRM) and the Department of Computer Science (SM), University of Miami, Coral Gables, Florida. Revision accepted for publication December 22, 1993.

Address correspondence and reprint requests to Dr. Plant: Department of Computer Information Systems, University of Miami, Coral Gables, FL 33124.

*We do not use ALICE itself, but a local implementation (MALICE) which follows the original very closely.

areas addressed by mental health professionals. The aim was not to replace these professionals, but to indicate research directions that can be addressed by mental health workers. It is very important to be aware that the accuracy of the diagnoses is dependent upon the accuracy of the inputs, so that although the computer implementation removes the possibility of human error in the computations, the results are still dependent upon the individual clinical judgement of the practitioner. Furthermore, a mechanical system should never be the sole basis for diagnosis; its conclusions should always be subject to review by a qualified person. The danger that the prognostications of a computer may be seen as infallible or more reliable than those of a human in decision-making situations is always present, not only in medical systems but in all systems of this type.[7]

The system as presented here is seen by the authors as a very strong prototype and not intended for use as a final product.

### DIFFERENTIAL-DIAGNOSIS DECISION SUPPORT SYSTEM

In this paper we utilize a decision-table form of intermediate representation, in which the DSM-III-R decision trees were converted into mechanically perfect decision tables.[1] The tables were reduced into their minimal forms, such that all redundant rules were removed. The tables are presented in appendix B.

## Parallel Processing through Graph Reduction

A program in an ALICE system consists not of the traditional ordered sequence of uniform instructions, but of a set of "packets" connected in a graph, or tree-like structure. Each packet represents either an operation to be performed or an item or structure of data. The connections, or arcs, between packets encode dependency, data-flow paths, orders of execution, and shapes of data structures.

An ALICE system itself consists of a number of logically identical processing agents sharing and independently executing the same program in parallel. These processing agents and their interfaces with the shared program store are designed in such a way that the execution time for any operation will have no component that is dependent upon the number of processors currently operating, and ideally that an individual processing agent can, with minimal delay, be connected to or disconnected from a system without disrupting the operations of that system. A consequence of this design is that if a single processing agent is engaged in a complex computation, a second agent may, at any time, be "plugged in," and (provided that there is enough work for two agents) halve the required processing time. Similarly, an agent may be "unplugged" from a system in which it is no longer required and reconnected to another, or even used as an individual work station.

### OPERATIONS OF A PROCESSING AGENT

When a processing agent is running, it will continually cycle through the familiar "fetch–execute" sequence: first it finds in the shared program store (called the "packet pool") any packet that is suitable for execution, and removes that packet, or marks it in some way, so that no other agent will select the same one. The agent then inspects the contents of the packet to find which operation it represents and what the parameters of that operation are to be (they may be either other packets or simple constants), and performs the operation by executing a sequence of instructions found in a preloaded program store.

Performing the operations indicated by a single packet will generally involve some or all of the following:

- Extracting information from argument packets
- Performing simple, low-level operations (such as arithmetic calculations)
- Updating information in other packets
- Creating new packets to represent new data structures, or newly required sub-computations
- Removing obsolete packets
- Causing external actions (such as user input or output)
- Replacing the original packet; as ALICE is a graph-reduction machine, this must be an "update-in-place" operation that will leave the surrounding graph structure intact, and the newly created subgraph installed as an integral part of that structure, completely replacing the original packet.

The following two sections illustrate the performance of these operations on a small mood-disorders example.

### PRODUCTION OF A GRAPH-REDUCTION PROGRAM

Here we present an approach to the use of graph reduction in the creation of knowledge-based systems. To facilitate this, we develop a knowledge-based system from a decision table, a form of intermediate representation often used by knowledge engineers in the development of knowledge-based systems.[8]

The approach we follow is to take the decision-table form as an input to a program that translates that table into a graph-reduction program that can be run on an ALICE machine (figure 1). We then consider these processes in greater detail.

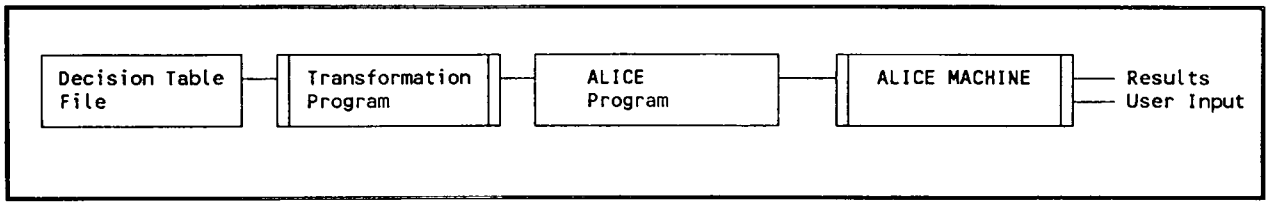


FIGURE 1. Decision table to graph-reduction transformation process.

## A Graph-reduction Program for Mood Disorders

The domain upon which our system is based in this first example is a small decision table from the set of decision tables that together combine to form the decision tree of DSM-III-R. It is presented here with only a limited discussion of its functionality in DSM-III-R, as its primary use is to help illustrate several important aspects of the graph-reduction approach to knowledge-based system implementation.

The decision table focuses upon an aspect of mood disorders and consists of only three conditions and five possible actions (figure 2).

Having a decision table from which to commence our development has significant advantages from the perspective of system verification and validation, and is an area of active research[9–12]; however, a full discussion of this is beyond the scope of this paper.

The decision table is processed by a Pascal program that mechanically translates the table into an ALICE program. This has several advantages, in that the correctness of the system is maintained and automatic transformation saves labor and time for the knowledge engineer and aids knowledge engineers who have little experience in developing graph-reduction systems. The full ALICE program resulting from transforming the decision table can be found in appendix A.

The ALICE program in appendix A is composed of 11 parts, each of which describes how to reduce a different type of packet: "initial," "query," "negative," "and2," "and3," "or2," "or3," "conclude," "iswanted," "propagate2," and "propagate3."

The first part:

```
To reduce initial:
  new(c1) id=query st=unready arg1="c1: Exclusive Depressive Synd"
  new(c2) id=query st=unready arg1="c2: Major Syndrome"
  new(c3) id=query st=unready arg1="c3: Two years of Dysthmia"
  new(c1n) id=negative st=unready arg1=c1
  new(c2n) id=negative st=unready arg1=c2
  new(c3n) id=negative st=unready arg1=c3
  new(k2) id=and2 st=unready arg1=c1n arg2=c2
  new(k3) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
  new(k4) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
  new(r1) id=or3 st=unready arg1=k2 arg2=k3 arg3=k4
  new(o1) id=conclude st=unready arg1=r1 arg2="a1: Major Depressive Episode"
  new(o2) id=conclude st=unready arg1=k2 arg2="a2: Depressive Disorder NOS"
  new(o3) id=conclude st=unready arg1=k3 arg2="a3: Dysthymia & Ma...depression"
  new(o4) id=conclude st=unready arg1=k4 arg2="a4: Major Depression"
  new(x1) id=propagate3 st=unready arg1=o1 arg2=o2 arg3=o3
  new(g0) id=propagate2 st=unready arg1=x1 arg2=o4
  rewrite id=iswanted arg1="starting" arg2=g0
```

is designed to construct the graph over which the graph-reduction system will be run. In this system we create the graph from the queries backwards to the conclusions, in a backward-chaining style. First, three condition packets are created (figure 3).

Each packet, when and if it is picked for reduction, is reduced according to the sets of rules provided in the ALICE program. Just as the section headed "To reduce initial" describes how the initial default seed packet is replaced to produce the machine's correct initial configuration, a later section of the program labeled "To reduce Query" will describe in detail how each of these new query packets should be reduced.

| | 1 | 5 | 7 | 8 |
|---|---|---|---|---|
| Exclusive Depressive Synd | Y | N | N | N |
| Major Syndrome | - | Y | N | N |
| Two Years of Dysthymia | - | - | Y | N |
| No Mood Disorder | X | | | |
| Major Depressive Episode | | | X | X |
| Depressive Disorder NOS | | X | | |
| Dysthymia & Major Depression | | | X | |
| Major Depression | | | | X |

FIGURE 2. A decision table for a graph reduction system.

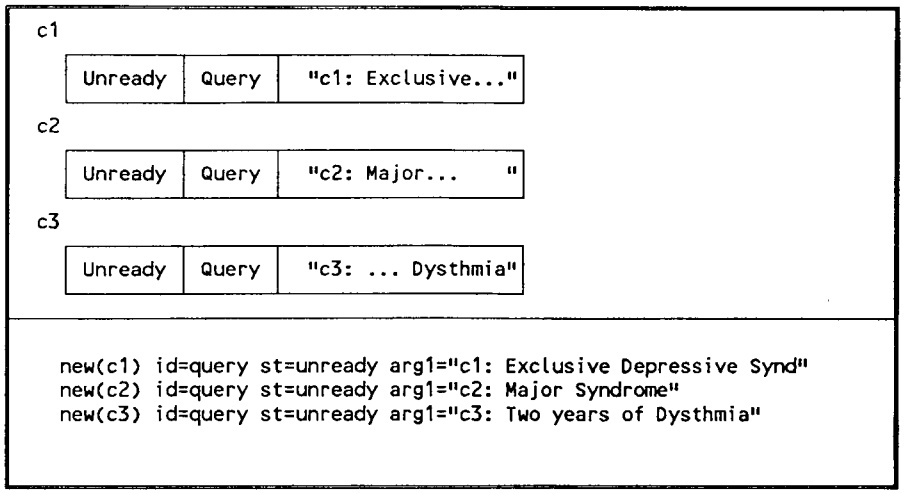


FIGURE 3. The condition packets.

The identifiers c1, c2, and c3 simply provide a means of referring back to these packets later.

The status field of a packet determines when a packet may be selected for reduction. Only packets that have a status of "ready" may ever be reduced. These query packets will remain unprocessed until their statuses are changed.

If a query packet is ever picked for reduction, the rules (shown in full in appendix A) specify that the associated string should be printed, and the user is invited to type a response. If the response is "YES," then the query packet is changed from a computational packet to a data packet representing the value TRUE. If the answer is "NO," then the value is FALSE. Any other response is ignored, and the packet is made available for a second reduction, thus causing the question to be asked again later.

The second aspect of system creation specified in "to reduce initial" is the creation of the *second packet layer* (figure 4).

This second layer provides negated forms of the conditions. If one of these packets is ever picked for reduction, it will be unable to proceed until its argument (the condition packet that it refers to) has already been reduced to a data packet. Until that happens, this packet is suspended, and the argument packet's status is changed to "ready" so that it may be selected. This is how demand for a computation is propagated through the graph of packets.

When a "negative" packet is eventually reduced, it is rewritten as a data packet with the logical opposite value to that of its condition.

The third layer, which represents the left-hand sides of the rules, calculates the conjunction of the relevant

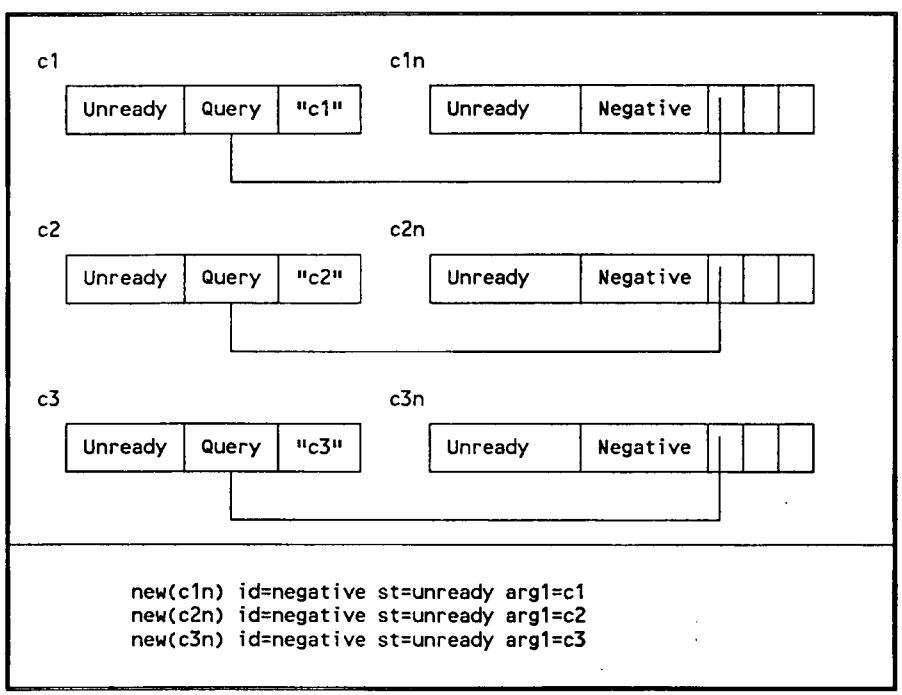


FIGURE 4. Second-layer propagation packets.

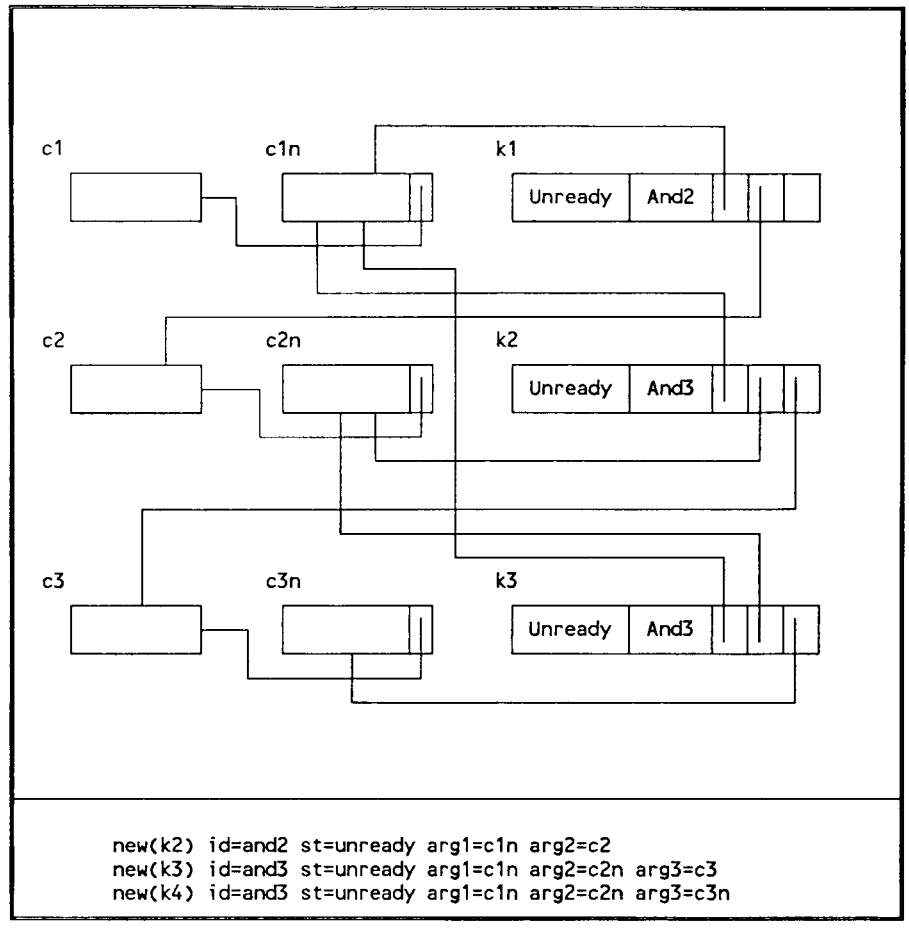


FIGURE 5. Third-layer conjunction packets.

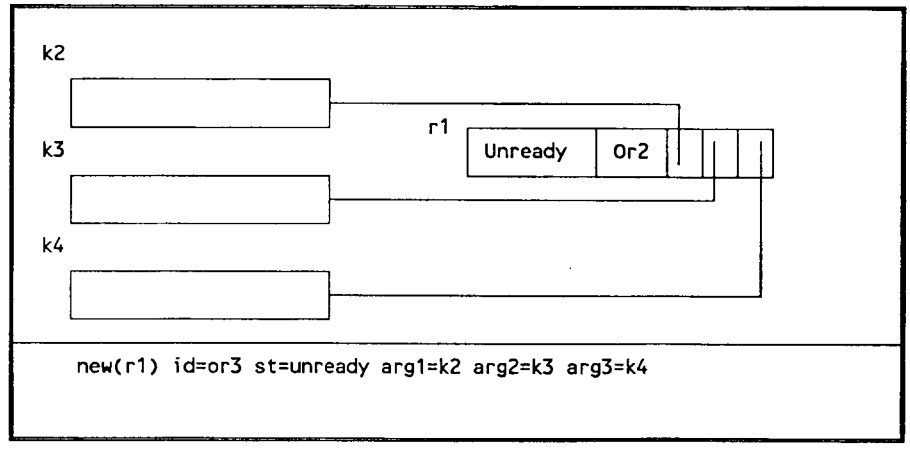


FIGURE 6. Fourth-layer disjunction packets.

conditions (e.g., the packet K3 shown in figure 5 represents the condition $\sim C1 \wedge \sim C2 \wedge C3$).

Similarly, the fourth layer, which exists only for right-hand sides that are activated by more than one conjunction of conditions, consists of logical disjunctions (figure 6).

"AND" and "OR" packets behave in fundamentally the same way as the "Negative" packets; their arguments are changed from "unready" to "ready" if they have not already been reduced to data, and once the arguments are data, these packets are also rewritten as either TRUE or FALSE data packets according to the operations they represent. In this implementation, complete evaluation of Booleans is performed. Once one of the arguments of an "AND" packet reduces to FALSE, the other packets are still evaluated. This is just a simplification for the purpose of illustration in this prototype, and certainly not a fixed feature of the graph-reduction method.

Having created all of the necessary left-hand-side conditions with their logical conjunctions and disjunctions, we can now create packets that represent

the right-hand sides of the rules, the conclusions (figure 7). The four packets shown in figure 8 represent the four possible conclusions of the decision table given in figure 2.

Finally, we have two packets that enable the system, once initiated, to propagate the activations through the graph.

### PROGRAM EXECUTION

Once the graph structure has been created, the ALICE system can proceed to reduce the graph based upon the input data from the user. An example of such a user-machine dialogue is of the form:

```
> Query c1: Exclusive Depressive Syndrome? y
> Query c2: Major Syndrome? y
> Query c3: Two years of Dysthmia? y
>> Conclusion a2: Major Depressive Episode
>> Conclusion a1: Dysthmia & major Depression
```

Whenever a processor is free, a packet with a ready status is selected at random; initially there are only two possibilities:

```
new(x1) id=propagate3 st=unready arg1=o1 arg2=o2 arg3=o3
new(g0) id=propagate2 st=unready arg1=x1 arg2=o4
```

In a machine with a single processor, one of these ready packets is selected (in a multi-processor system as many packets as there are processors may be selected simultaneously); for example, packet x1 whose purpose is to propagate execution, could be selected. This causes each of its argument packets also to become ready. Thus packets o1, o2, and o3 change status to "ready," and are placed in the pool of packets ready to be reduced (which also contains the other propagation packet g0 as it remains in the ready state). The activation of x1 propagates activation of the packets, which continues until it reaches a packet that does not rely on any others; in our example, perhaps, this would be packet C2, which could be completely reduced, causing the question "c2:Major Syndrome" to be asked. The reductions continue until all ready packets have been reduced, with those conclusions whose arguments were reduced to TRUE having been output as results.

## A Differential-diagnosis Graph-reduction Program for Psychiatric Disorders

In the preceding section we illustrated the process of graph reduction as applied to a small aspect of psychiatric mood disorders. This section discusses the use of these techniques upon all three aspects of the DSM-III-R flow charts.

The processes described earlier in the paper to produce the ALICE graph-reduction program for mood disorders were followed in an identical sequence. However, instead of the input to the transformation-and-reduction process being one decision table, we utilized ten decision tables, as shown in appendix B (*n.b.*, The organic B table is split into four sections for clarity). The multiple decision tables were also connected in a graph, as the control is passed from one

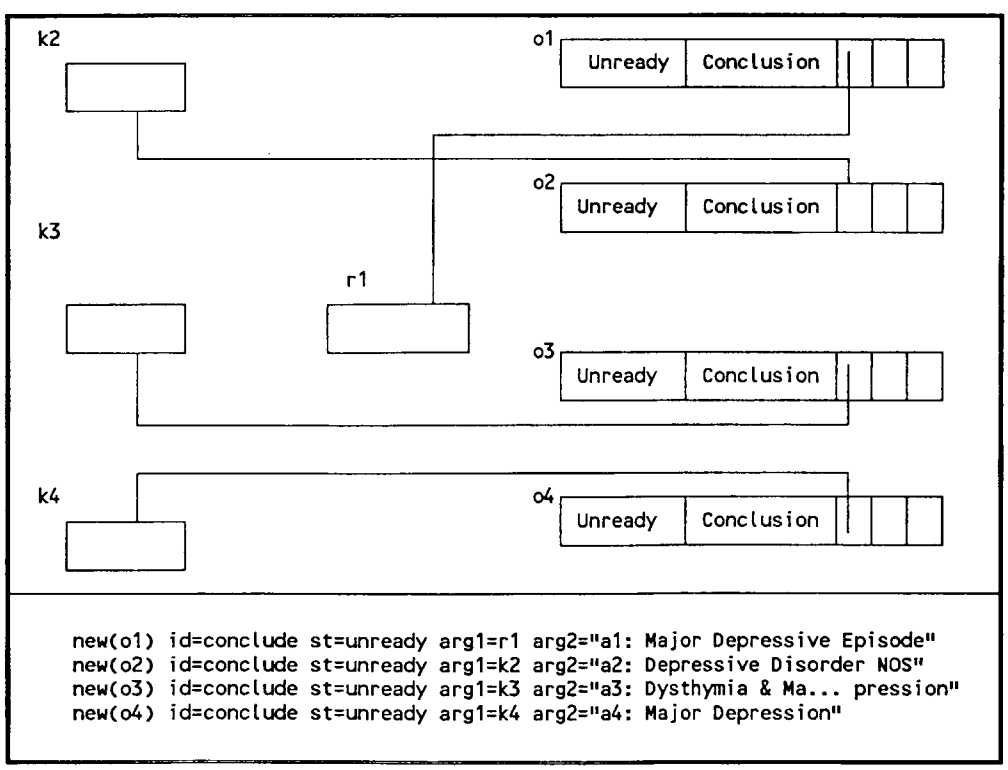


FIGURE 7. Creation of the right-hand-side conclusion packets.

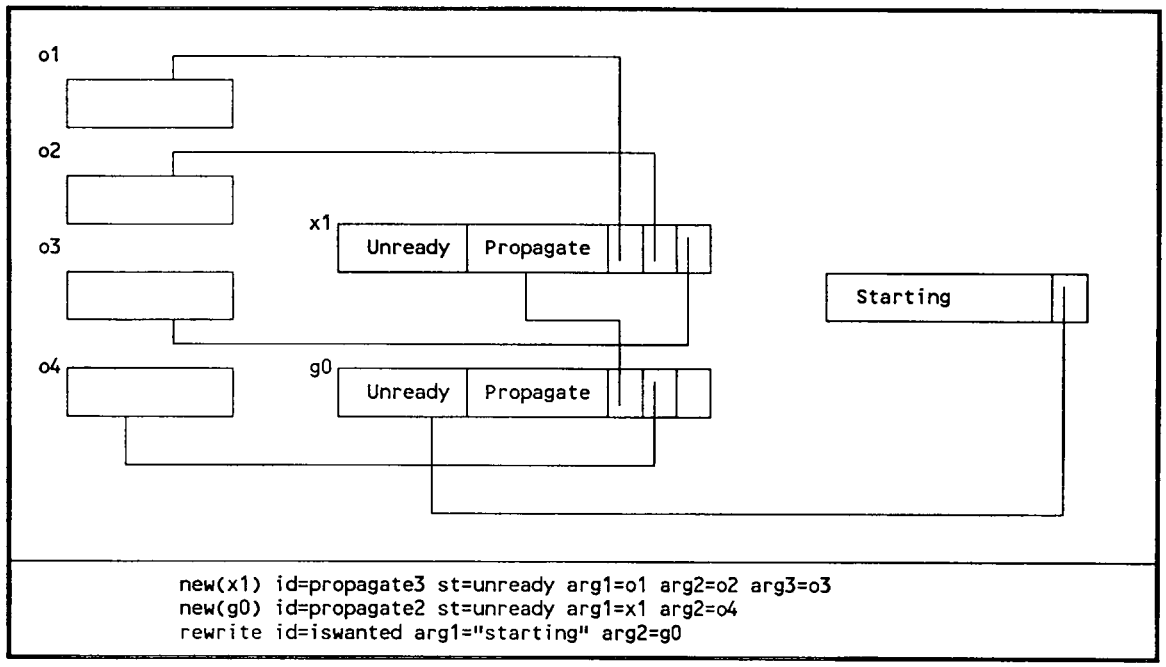


FIGURE 8. Propagation packets and initial packet.

table to another as the differential diagnosis proceeds. The transformation program that produces the ALICE graph-reduction program takes each of the decision tables in turn, creating the appropriate packets, linkages, and propagation layers to model the decision-table graph. The resultant ALICE program is provided in appendix C.

## SYSTEM OPERATION

Having created the ALICE program, the system is run in a fashion identical to that of sequential programming systems. The user is unaware of the parallel nature of the reduction mechanism that is being performed by the ALICE architecture. An example of a user dialogue with the system is of the form:

```
> Query c1: Signs? yes
> Query c2: Organic Factor? yes
> Query c3: Month? no
> Query c4: Week? yes
> Query c41: Concurrent? yes
> Query c42: Duration? yes
> Query c45: Two Weeks? no
> Query c43: Impaired? yes
> Query c44: More than Six Months? yes
>> Conclusion a43: Schizophreniform Disorder
```

where the user answers queries from the system until a resultant conclusion is produced by the system. The mnemonics, e.g., c1, c2, refer to the conditions in the corresponding decision table. The prototype system as illustrated here has only limited sophistication with respect to the man–machine interface; however, this is not a significant problem and could easily be addressed in subsequent versions of the system. The aim of this paper was to focus upon the principle of parallel processing and the use of graph reduction on an ALICE machine, in relation to the sophisticated domain of psychiatric disorders. A more complex example of system interaction can be seen through the following example:

```
> Query c1: Signs? yes
> Query c2: Organic Factor? yes
> Query c11: Evidence? yes
> Query c12: Disturbance? yes
> Query c13: Other Symptoms? no
> Query c14: Impairment? no
> Query c15: Change? yes
>> Conclusion a11: Delirium
>> Conclusion c15: Organic Personality Syndrome
> Query c101: Prominent Delusions? yes
> Query c102: Prominent Hallucinations? yes
> Query c103: Elevated or Expansive Mood? no
> Query c104: Anxiety? no
> Query c105: Organic Mental Disorder? no
> Query c106: Psychoactive Sub Etiology? yes
> Query c107: Before Use? yes
> Query c108: After Use? yes
>> Conclusion a102: Organic Hallucinosis
>> Conclusion a107: Psychoactive Substance Induced Intoxication
>> Conclusion a108: Psychoactive Substance Withdrawal
```

In this, a number of organic disorders were determined to be present for a given set of input criteria. In order to assess the validity of the system's results, the system was run against all paths in the decision tables and the results determined to be correct with respect to those tables. Further, as the tables were determined to be correct and consistent with respect to the DSM-III-R decision flow charts, we can state that the system is verified and valid. This is a much stronger statement than we were able to make in the implementation of the same domain decision tables in the rule-based form.[1] This was the result of the system's being coded by hand and "crafted" from the decision tables. Further, the syntax and size of the

rule-based VP-Expert programs made verification of that form of implementation a very difficult task. Thus, the burden of demonstrating correctness has been removed from the implementation stage to the decision-table-design stage, a significant benefit.

### SYSTEM DYNAMICS AND PERFORMANCE CONSIDERATIONS

The creation of large graph-reduction programs in conjunction with the associated ALICE parallel-processing architectures has several advantages from the consideration of system performance. The first of these is that of timing, as the time required (for a single processor) to reduce the graph is directly proportional to the size of the original decision table, and therefore the worst-case time for deductions is linear in relation to the problem size. This, however, can be improved through the use of partitioning when a confirmation rather than an initial diagnosis is required. Then individual conclusions are isolated such that only one conclusion is active; if there are enough processors in the system to perform all of the appropriate reductions concurrently, the time to perform deductions within these constraints is proportional to the depth of the AND/OR tree controlling the active conclusion. Further, as each of the packets can reference three other packets, that depth is approximately the base-three logarithm of the number of conditions in each AND tree, plus the base-three logarithm of the number of AND trees in the OR tree. Thus, the best-case time for a deduction is logarithmic in the problem size.

## Conclusions

The aim of this paper was to consider an alternative approach to the implementation of mood, psychotic, and organic mental disorders through the use of an ALICE graph-reduction parallel-processing program. The advantages of this approach stem from the ability of the system designer to create large, complete, and complex graphs in an easy and straightforward manner. The ability of the developer to create a working parallel-processing program from a series of decision tables opens up a wide variety of other psychiatric, medical, and decision support domains to which the technique can be applied.

The general advantages of the use of decision support technology in the fields of mental health and medicine have been extensively documented. See Jakab for extended references[13] and Burszajn et al.,[14] Gutheil et al.,[15] and Moreno[1] for a discussion of decision support systems in psychiatry. However, the use of parallel processing for decision support systems has had little attention. The technique introduced in this paper has the ability to perform computations in logarithmic time, which is of significant advantage, and when combined with the ability to easily construct graph-reduction systems, should lead to the utilization of this technology in other domains.

Finally, we wish to stress that the system as presented here is not intended as a replacement for the clinician but rather as an indication of an advanced technology suitable for the mental health care profession. Although it appears that the system is mechanically perfect, it is lacking in the human judgment that is essential for true expertise. The prototype indeed showed that considerable work in the decision support systems subfields of certainty factor analysis, explanations, representations, and hardware is needed prior to full exploitation of this technology by the practicing clinician.

The authors thank Susan S. Moreno, MD, and Terry A. Pulito, MD, of the Department of Psychiatry, School of Medicine, University of Miami.

## References

1. Moreno HR, Plant RT. A prototype decision support system for differential diagnosis of psychotic, mood, and organic mental disorders. Med Decis Making. 1993;13:43–8.
2. Diagnosis and Statistical Manual of Mental Disorders. 3rd ed. rev. Washington, DC: American Psychiatric Association, 1987.
3. Darlington J, Reeve M. ALICE: a multiprocessor reduction machine for the parallel evaluation of applicative languages. ACM/MIT Conf. on Functional Programming Languages and Computer Architecture, New Hampshire, 1981.
4. Townsend P. Flagship hardware and implementation. ICL Tech J. 1987;5:575–94.
5. Maurer K, Biehl K, Kuhner C, Loffler W. On the way to expert systems. Comparing DSM-III computer diagnoses with CATEGO (ICD) diagnoses in depressive and schizophrenic patients. Eur Arch Psychiatry Neurol Sci. 1989;239:127–32.
6. Werner G. Methuselah: an expert system for diagnosis in geriatric psychiatry. Comput Biomed Res. 1987;20:477–88.
7. Buchanan BG, Shortliffe EH. Rule-based Expert Systems. Reading, MA: Addison Wesley, 1984.
8. Plant RT. A methodology for knowledge acquisition in the development of knowledge-based systems, PhD thesis, University of Liverpool, Liverpool, England, 1994.
9. O'Leary DE, ed. Collected Papers of AAAI Workshops on Validation and Verification 1988–92. Reading, MA: Wiley, 1994.
10. Preece AD, Shinghal R, Batarekh A. Verifying expert systems: a logical framework and a practical tool. Expert Systems with Applications. New York: Pergamon, 1992:421–36.
11. Rushby J. Quality measures and assurance for AI software. Langley, VA: NASA, 1988; NASA Contact Report NASI-17067.
12. Culbert C. Verification and validation of knowledge-based systems. Special issue: Expert Systems with Applications. Vol 1, No 3. New York: Pergamon, 1990.
13. Jakab I. Artificial intelligence in medicine and psychiatry: new developments in the 1990's. Proc. Third Annu Symp Int Assoc Knowledge Engineers. Washington, DC: November 16–19, 1992: 241–61.
14. Bursztajn HJ, Feinbloom RI, Hamm RM, Brodsky A. Medical Choices, Medical Changes. London, England: Routledge, 1990.
15. Gutheil TG, Burstazjn HJ, Brodsky A, Alexander V. Decision Making in Psychiatry and the Law. Baltimore: Williams & Wilkins, 1991.

## APPENDIX A

```
To reduce initial:
  new(c1) id=query st=unready arg1="c1: Exclusive Depressive Synd"
  new(c2) id=query st=unready arg1="c2: Major Syndrome"
  new(c3) id=query st=unready arg1="c3: Two years of Dysthmia"
  new(c1n) id=negative st=unready arg1=c1
  new(c2n) id=negative st=unready arg1=c2
  new(c3n) id=negative st=unready arg1=c3
  new(k2) id=and2 st=unready arg1=c1n arg2=c2
  new(k3) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
  new(k4) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
  new(r1) id=or3 st=unready arg1=k2 arg2=k3 arg3=k4
  new(o1) id=conclude st=unready arg1=r1 arg2="a1: Major Depressive Episode"
  new(o2) id=conclude st=unready arg1=k2 arg2="a2: Depressive Disorder NOS"
  new(o3) id=conclude st=unready arg1=k3 arg2="a3: Dysthymia & Ma...epression"
  new(o4) id=conclude st=unready arg1=k4 arg2="a4: Major Depression"
  new(x1) id=propagate3 st=unready arg1=o1 arg2=o2 arg3=o3
  new(g0) id=propagate2 st=unready arg1=x1 arg2=o4
  rewrite id=iswanted arg1="starting" arg2=g0

to reduce query:
  write string "Query"
  write string arg1
  write string "?"
  read char x
  read char y
  k=0
  if (x=89) or (x=121)
    k=1
    rewrite id=true st=data arg1=arg1
  if (x=78) or (x=110)
    k=1
    rewrite id=false st=data arg1=arg1
  if k=0
    rewrite id=query st=ready arg1=arg1

to reduce positive:
  require arg1
  if (arg1.id)=true
    rewrite id=true st=data arg1=arg1.arg1
  else
    rewrite id=false st=data arg1=arg1.arg1

to reduce negative:
  require arg1
  if (arg1.id)=true
    rewrite id=false st=data arg1=arg1.arg1
  else
    rewrite id=true st=data arg1=arg1.arg1

to reduce and2:
  require arg1
  require arg2
  if ((arg1.id)=true) and ((arg2.id)=true)
    rewrite id=true st=data arg1=arg1 arg2=arg2
  else
    rewrite id=false st=data arg1=arg1 arg2=arg2

to reduce and3:
  require arg1
  require arg2
  require arg3
  if ((arg1.id)=true) and ((arg2.id)=true) and ((arg3.id)=true)
    rewrite id=true st=data arg1=arg1 arg2=arg2 arg3=arg3
  else
    rewrite id=false st=data arg1=arg1 arg2=arg2 arg3=arg3

to reduce or2:
  require arg1
  require arg2
  if ((arg1.id)=true) or ((arg2.id)=true)
    rewrite id=true st=data arg1=arg1 arg2=arg2
  else
    rewrite id=false st=data arg1=arg1 arg2=arg2

to reduce or3:
  require arg1
  require arg2
  require arg3
  if ((arg1.id)=true) or ((arg2.id)=true) or ((arg3.id)=true)
    rewrite id=true st=data arg1=arg1 arg2=arg2 arg3=arg3
  else
    rewrite id=false st=data arg1=arg1 arg2=arg2 arg3=arg3

to reduce conclude:
  require arg1
  if ((arg1.id)=true)
    write string "Conclusion"
    write string arg2
    write string "."
    write char 10
    rewrite id=true st=data arg1=arg1
  else
    rewrite id=false st=data arg1=arg1

to reduce iswanted:
  write string "Are you interested in "
  write string arg1
  write string "?"
  read char x
  read char y
  if (x=89) or (x=121)
    rewrite id=wakeup arg1=arg2
  else
    if (x=78) or (x=110)
      rewrite id=useless st=data rc=0
    else
      rewrite id=iswanted arg1=arg1 arg2=arg2

to reduce wakeup:
  require arg1
  rewrite id=useless st=data rc=0

to reduce propagate2:
  require arg1
  require arg2
  rewrite id=useless st=data rc=0

to reduce propagate3:
  require arg1
  require arg2
  require arg3
  rewrite id=useless st=data rc=0
```

## APPENDIX B

| Signs | Y | Y | Y | Y |
|---|---|---|---|---|
| Organic Factor | N | N | N | Y |
| Month | Y | N | N | - |
| Week | - | Y | N | - |
| Chain Organic A | | | | X |
| Chain Psych 1 | X | | | |
| Chain Psych 2 | | X | | |
| Chain Psych 3 | | | X | |

*1. Main Decision Table (t0).*

| | 1 | 9 | 13 | 14 | 15 | 16 | 17 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Evidence | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| Disturbance | Y | Y | Y | Y | Y | Y | N | N | N | N | N |
| Other Symptoms | Y | N | N | N | N | N | - | - | - | - | - |
| Impairment | - | Y | N | N | N | N | Y | N | N | N | N |
| Memory | - | - | Y | Y | N | N | - | Y | Y | N | N |
| Change | - | - | Y | N | Y | N | - | Y | N | Y | N |
| Delirium | X | X | X | X | X | X | | | | | |
| No Additional Diagnosis | X | | | | | | | | | | X |
| Dementia | | X | | | | | X | | | | |
| Amnestic Syndrome | | | X | X | | | | X | X | | |
| Organic Personality Synd | | | X | | X | | | X | | X | |

*2. Organic Disorders Decision Table A (t1).*

| | 1 | 3 | 5 | 8 |
|---|---|---|---|---|
| Response 1 | Y | Y | N | N |
| Full Mood | Y | N | - | - |
| Symptoms 1 | - | - | Y | N |
| Psychotic Mood Disorder | X | | | |
| Brief Reactive Disorder | | X | | |
| Chain Psycho02 | | | X | |
| Chain Psycho03 | | | | X |

*3. Psychotic Disorders Decision Table 1 (t2).*

| Persistent Mood | Y | Y | Y | Y | Y |
|---|---|---|---|---|---|
| Organic Factor | Y | N | N | N | N |
| One or More Periods | - | Y | Y | N | N |
| At Least one Mood Syndrome | - | Y | N | - | - |
| Two Weeks | - | - | - | Y | N |
| Organic Tree | X | | | | |
| Chain Mood 1 | | | | X | |
| Chain Mood 2 | | | | | X |
| Chain Mood 3 | | X | | | |
| Chain Mood 4 | | | X | | |

*4. Mood-Driver Decision Table (t3).*

| | 1 | 3 | 5 | 9 | 10 | 17 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|
| Concurrent | Y | Y | Y | Y | Y | N | N | N |
| Duration | Y | Y | Y | N | N | - | - | - |
| Impaired | Y | Y | N | - | - | Y | Y | N |
| More Than Six Months | Y | N | - | - | - | Y | N | - |
| Two Weeks | - | - | - | Y | N | - | - | - |
| Psychotic Mood Disorder | | | | | X | | | |
| Schizoaffective Disorder | | | | X | | | | |
| Psychotic Disorder NOS | | | X | | | | | X |
| Schizophreniform Disorder | | X | | | | | X | |
| Schizophrenia | X | | | | | X | | |

*5. Psychotic Disorders Decision Table 2 (t4).*

| | 1 | 2 | 5 | 7 | 9 | 10 |
|---|---|---|---|---|---|---|
| Present | Y | Y | Y | Y | N | N |
| Duration | Y | Y | N | N | - | - |
| Persist | - | - | Y | N | - | - |
| Delusion | Y | N | - | - | Y | N |
| Delusional Disorder | X | | | | X | |
| Psychotic Disorder NOS | | X | X | | | X |
| Psychotic Mood Disorder | | | | X | | |

*6. Psychotic Decision Table 3 (t5).*

| | 1 | 5 | 7 | 8 |
|---|---|---|---|---|
| Exclusive Depressive Synd | Y | N | N | N |
| Major Syndrome | - | Y | N | N |
| Two Years of Dysthymia | - | - | Y | N |
| No Mood Disorder | X | | | |
| Major Depressive Episode | | | X | X |
| Depressive Disorder NOS | | X | | |
| Dysthymia & Major Depression | | | X | |
| Major Depression | | | | X |

*7. Mood Disorders Decision Table 1 (t6).*

| | 1 | 5 | 7 | 8 |
|---|---|---|---|---|
| Superimposed Dysthymia Syndrome | Y | N | N | N |
| Two Years of Dysthymia | - | Y | N | N |
| Depressed Mood | - | - | Y | N |
| No Mood Disorder | X | | | |
| Dysthymia | | X | | |
| Depressive Disorder NOS | | | | X |
| Adjusted Disorder | | | X | |

*8. Mood Disorders Decision Table 2 (t7).*

| | 1 | 5 | 6 | 8 |
|---|---|---|---|---|
| Exclusive Depressive Syndrome | Y | N | N | N |
| Major Syndrome NOT Superimposed | - | Y | Y | N |
| Two Years of Cyclothymia | - | Y | N | - |
| No Mood Disorder | X | | | |
| Bipolar Disorder NOS | | | | X |
| Manic Episode | | X | X | |
| Cyclothymia & Bipolar Disorder | | X | | |
| Bipolar Disorder | | | X | |

*9. Mood Disorders Decision Table 3 (t8).*

| | 1 | 3 | 4 |
|---|---|---|---|
| Hypomanic Syndrome | Y | N | N |
| Numerous Hypomanic Episodes | - | Y | N |
| No Mood Disorder | X | | |
| Hypomanic Episode | | X | X |
| Bipolar Disorder NOS | | | X |
| Cyclothymia & Bipolar Disorder | | X | |

*10. Mood Disorders Decision Table 4 (t9).*

*11. Organic Disorders Decision Table B (Part 1 of 4) (t10).*

| | 1 | 17 | 25 | 26 | 27 | 28 | 29 | 33 | 49 | 57 | 58 | 59 | 60 | 61 | 65 | 81 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Prominent Delusions | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| Prominent Hallucinations | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | N | N |
| Elevated or Expansive Mood | Y | Y | Y | Y | Y | Y | Y | N | N | N | N | N | N | N | Y | Y |
| Anxiety | Y | N | N | N | N | N | N | Y | N | N | N | N | N | N | Y | N |
| Organic Mental Disorder | - | Y | N | N | N | N | N | - | Y | N | N | N | N | N | - | Y |
| Psychoactive Sub Etiology | - | - | Y | Y | Y | Y | N | - | - | Y | Y | Y | Y | N | - | - |
| Before Use | - | - | Y | Y | N | N | - | - | - | Y | Y | N | N | - | - | - |
| After Use | - | - | Y | N | Y | N | - | - | - | Y | N | Y | N | - | - | - |
| Org Delusional Syndrome | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| Org Hallucinosis | X | X | X | X | X | X | X | X | X | X | X | X | X | X | | |
| Org Mood Syndrome | X | X | X | X | X | X | X | | | | | | | | X | X |
| Org Anxiety Syndrome | X | | | | | | | X | | | | | | | X | |
| No Addit'al Diagnosis Given | | X | | | | | | | X | | | | | | | X |
| Org Mental Disorder NOS | | | | X | | X | X | | | | X | | X | X | | |
| Psy Sub-induced Intoxic. | | | X | X | | | | | | X | X | | | | | |
| Psychoactive Sub Withdrawal | | | X | | X | | | | | X | | X | | | | |

*12. Organic Disorders Decision Table B (Part 2 of 4) (t10).*

| | 89 | 90 | 91 | 92 | 93 | 97 | 98 | 113 | 121 | 122 | 123 | 124 | 125 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Prominent Delusions | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| Prominent Hallucinations | N | N | N | N | N | N | N | N | N | N | N | N | N |
| Elevated or Expansive Mood | Y | Y | Y | Y | Y | N | N | N | N | N | N | N | N |
| Anxiety | N | N | N | N | N | Y | Y | N | N | N | N | N | N |
| Organic Mental Disorder | N | N | N | N | N | Y | - | Y | N | N | N | N | N |
| Psychoactive Sub Etiology | Y | Y | Y | Y | N | - | - | - | Y | Y | Y | Y | N |
| Before Use | Y | Y | N | N | - | - | - | - | Y | Y | N | N | - |
| After Use | Y | N | Y | N | - | - | - | - | Y | N | Y | N | - |
| Org Delusional Syndrome | X | X | X | X | X | X | X | X | X | X | X | X | X |
| Org Hallucinosis | | | | | | | | | | | | | |
| Org Mood Syndrome | X | X | X | X | X | | | | | | | | |
| Org Anxiety Syndrome | | | | | | X | X | | | | | | |
| No Addit'al Diagnosis Give | | | | | | | | | | | | | |
| Org Mental Disorder NOS | | X | | X | X | | | X | | X | | X | X |
| Psy Sub-induced Intoxic. | X | X | | | | | | | X | X | | | |
| Psychoactive Sub Withdrawa | X | | X | | | | | | X | | X | | |

| | 129 | 145 | 153 | 154 | 155 | 156 | 157 | 161 | 177 | 185 | 186 | 187 | 188 | 189 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Prominent Delusions | N | N | N | N | N | N | N | N | N | N | N | N | N | N |
| Prominent Hallucinations | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y |
| Elevated or Expansive Mood | Y | Y | Y | Y | Y | Y | Y | N | N | N | N | N | N | N |
| Anxiety | Y | N | N | N | N | N | N | Y | N | N | N | N | N | N |
| Organic Mental Disorder | - | Y | N | N | N | N | N | - | Y | N | N | N | N | N |
| Psychoactive Sub Etiology | - | - | Y | Y | Y | Y | N | - | - | Y | Y | Y | Y | N |
| Before Use | - | - | Y | Y | N | N | - | - | - | Y | Y | N | N | - |
| After Use | - | - | Y | N | Y | N | - | - | - | Y | N | Y | N | - |
| Org Delusional Syndrome | | | | | | | | | | | | | | |
| Org Hallucinosis | X | X | | | | | X | X | X | X | X | X | X | X |
| Org Mood Syndrome | X | X | X | | | | X | | | | | | | |
| Org Anxiety Syndrome | X | | X | | | | | X | | | | | | |
| No Addit'al Diagnosis Given | | X | | | | | | | X | | | | | |
| Org Mental Disorder NOS | | | | X | | X | X | | | | X | | X | X |
| Psy Sub-induced Intoxic. | | | X | X | | | | | | X | X | | | |
| Psychoactive Sub Withdrawal | | | X | | X | | | | | X | | X | | |

*13. Organic Disorders Decision Table B (Part 3 of 4) (t10).*

*14. Organic Disorders Decision Table B (Part 4 of 4) (t10).*

| | 193 | 209 | 217 | 218 | 219 | 220 | 221 | 225 | 241 | 249 | 250 | 251 | 252 | 253 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Prominent Delusions | N | N | N | N | N | N | N | N | N | N | N | N | N | N |
| Prominent Hallucinations | N | N | N | N | N | N | N | N | N | N | N | N | N | N |
| Elevated or Expansive Mood | Y | Y | Y | Y | Y | Y | N | N | N | N | N | N | N | N |
| Anxiety | Y | N | N | N | N | N | N | Y | N | N | N | N | N | N |
| Organic Mental Disorder | - | Y | N | N | N | N | N | - | Y | N | N | N | N | N |
| Psychoactive Sub Etiology | - | - | Y | Y | Y | Y | N | - | - | Y | Y | Y | Y | N |
| Before Use | - | - | Y | Y | N | N | - | - | - | Y | Y | N | N | - |
| After Use | - | - | Y | N | Y | N | - | - | - | Y | N | Y | N | - |
| Org Delusional Syndrome | | | | | | | | | | | | | | |
| Org Hallucinosis | | | | | | | | | | | | | | |
| Org Mood Syndrome | X | X | X | X | X | X | X | | | | | | | |
| Org Anxiety Syndrome | X | | | | | | | X | | | | | | |
| No Addit'al Diagnosis Give | | X | | | | | | | X | | | | | |
| Org Mental Disorder NOS | | | | X | | X | X | | | | X | | X | X |
| Psy Sub-induced Intoxic. | | | X | X | | | | | | X | X | | | |
| Psychoactive Sub Withdrawa | | | X | | X | | | | | X | | X | | |

## APPENDIX C

```
To reduce initial:
  new(c1) id=query st=unready arg1="c1: Signs"
  new(c2) id=query st=unready arg1="c2: Organic Factor"
  new(c3) id=query st=unready arg1="c3: Month"
  new(c4) id=query st=unready arg1="c4: Week"
  new(c1n) id=negative st=unready arg1=c1
  new(c2n) id=negative st=unready arg1=c2
  new(c3n) id=negative st=unready arg1=c3
  new(c4n) id=negative st=unready arg1=c4
  new(k1) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
  new(x1) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
  new(k2) id=and2 st=unready arg1=x1 arg2=c4
  new(x2) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
  new(k3) id=and2 st=unready arg1=x2 arg2=c4n
  new(k4) id=and2 st=unready arg1=c1 arg2=c2
  new(o1) id=chain st=unready arg1=k4 arg2=table1 arg3="t1: Organic"
  new(o2) id=chain st=unready arg1=k1 arg2=table2 arg3="t2: Psych1"
  new(o3) id=chain st=unready arg1=k2 arg2=table4 arg3="t4: Psych2"
  new(o4) id=chain st=unready arg1=k3 arg2=table5 arg3="t5: Psych3"
  new(x3) id=propagate3 st=unready arg1=o1 arg2=o2 arg3=o3
  rewrite id=propagate2 arg1=x3 arg2=o4

To reduce table1:
  new(c1) id=query st=unready arg1="c11: Evidence"
  new(c2) id=query st=unready arg1="c12: Disturbance"
  new(c3) id=query st=unready arg1="c13: Other Symptoms"
  new(c4) id=query st=unready arg1="c14: Impairment"
  new(c5) id=query st=unready arg1="c14: Memory"
  new(c6) id=query st=unready arg1="c15: Change"
  new(c1n) id=negative st=unready arg1=c1
  new(c2n) id=negative st=unready arg1=c2
  new(c3n) id=negative st=unready arg1=c3
  new(c4n) id=negative st=unready arg1=c4
  new(c5n) id=negative st=unready arg1=c5
  new(c6n) id=negative st=unready arg1=c6
  new(k1) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
  new(x1) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
  new(k2) id=and2 st=unready arg1=x1 arg2=c4
  new(x2) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
  new(x3) id=and3 st=unready arg1=c4n arg2=c5 arg3=c6
  new(k3) id=and2 st=unready arg1=x2 arg2=x3
  new(x4) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
  new(x5) id=and3 st=unready arg1=c4n arg2=c5 arg3=c6n
  new(k4) id=and2 st=unready arg1=x4 arg2=x5
  new(x6) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
  new(x7) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
  new(k5) id=and2 st=unready arg1=x6 arg2=x7
  new(x8) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
  new(x9) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
  new(k6) id=and2 st=unready arg1=x8 arg2=x9
  new(k7) id=and3 st=unready arg1=c1 arg2=c2n arg3=c4
  new(x10) id=and3 st=unready arg1=c1 arg2=c2n arg3=c4n
  new(x11) id=and2 st=unready arg1=c5 arg2=c6
  new(k8) id=and2 st=unready arg1=x10 arg2=x11
  new(x12) id=and3 st=unready arg1=c1 arg2=c2n arg3=c4n
  new(x13) id=and2 st=unready arg1=c5 arg2=c6n
  new(k9) id=and2 st=unready arg1=x12 arg2=x13
  new(x14) id=and3 st=unready arg1=c1 arg2=c2n arg3=c4n
  new(x15) id=and2 st=unready arg1=c5n arg2=c6
  new(k10) id=and2 st=unready arg1=x14 arg2=x15
  new(x16) id=and3 st=unready arg1=c1 arg2=c2n arg3=c4n
  new(x17) id=and2 st=unready arg1=c5n arg2=c6n
  new(k11) id=and2 st=unready arg1=x16 arg2=x17
  new(x18) id=or3 st=unready arg1=k1 arg2=k2 arg3=k3
  new(x19) id=or3 st=unready arg1=k4 arg2=k5 arg3=k6
  new(r1) id=or2 st=unready arg1=x18 arg2=x19
  new(o1) id=conclude st=unready arg1=r1 arg2="a11: Delirium"
  new(r2) id=or2 st=unready arg1=k1 arg2=k11
  new(o2) id=conclude st=unready arg1=r2 arg2="a12: No Additional Diagnosis"
  new(r3) id=or2 st=unready arg1=k2 arg2=k7
  new(o3) id=conclude st=unready arg1=r3 arg2="a13: Dementia"
  new(x20) id=or3 st=unready arg1=k3 arg2=k4 arg3=k8
  new(r4) id=or2 st=unready arg1=x20 arg2=k9
  new(o4) id=conclude st=unready arg1=r4 arg2="a14: Amnestic Syndrome"
  new(x21) id=or3 st=unready arg1=k3 arg2=k5 arg3=k8
  new(r5) id=or2 st=unready arg1=x21 arg2=k10
  new(o5) id=conclude st=unready arg1=r5 arg2="a15: Organic Personality Synd"
  new(x22) id=or3 st=unready arg1=k1 arg2=k2 arg3=k3
  new(x23) id=or3 st=unready arg1=k4 arg2=k5 arg3=k6
  new(x24) id=or3 st=unready arg1=k7 arg2=k8 arg3=k9
  new(x25) id=or2 st=unready arg1=k10 arg2=k11
  new(x26) id=or3 st=unready arg1=x22 arg2=x23 arg3=x24
  new(r6) id=or2 st=unready arg1=x26 arg2=x25
  new(o6) id=chain st=unready arg1=r6 arg2=table10 arg3="t10: Organic B"
  new(x27) id=propagate3 st=unready arg1=o1 arg2=o2 arg3=o3
  new(x28) id=propagate3 st=unready arg1=o4 arg2=o5 arg3=o6
  rewrite id=propagate2 arg1=x27 arg2=x28

To reduce table2:
  new(c1) id=query st=unready arg1="c21: Response 1"
  new(c2) id=query st=unready arg1="c22: Full Mood"
  new(c3) id=query st=unready arg1="c23: Symptoms 1"
  new(c1n) id=negative st=unready arg1=c1
  new(c2n) id=negative st=unready arg1=c2
  new(c3n) id=negative st=unready arg1=c3
  new(k1) id=and2 st=unready arg1=c1 arg2=c2
  new(k2) id=and2 st=unready arg1=c1 arg2=c2n
  new(k3) id=and2 st=unready arg1=c1n arg2=c3
  new(k4) id=and2 st=unready arg1=c1n arg2=c3n
  new(o1) id=chain st=unready arg1=k1 arg2=table3 arg3="t3: Psychotic Mood
    Disorder"
  new(o2) id=conclude st=unready arg1=k2 arg2="a22: Brief Reactive Disorder"
  rewrite id=propagate2 arg1=o1 arg2=o2

To reduce table3:
  new(c1) id=query st=unready arg1="c31: Persistent Mood"
  new(c2) id=query st=unready arg1="c32: Organic Factor"
  new(c3) id=query st=unready arg1="c33: One or More Periods"
  new(c4) id=query st=unready arg1="c34: At Least One Mood Sydrome"
  new(c5) id=query st=unready arg1="c35: Two Weeks"
  new(c1n) id=negative st=unready arg1=c1
  new(c2n) id=negative st=unready arg1=c2
  new(c3n) id=negative st=unready arg1=c3
  new(c4n) id=negative st=unready arg1=c4
  new(c5n) id=negative st=unready arg1=c5
  new(k1) id=and2 st=unready arg1=c1 arg2=c2
  new(x1) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
  new(k2) id=and2 st=unready arg1=x1 arg2=c4
  new(x2) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
  new(k3) id=and2 st=unready arg1=x2 arg2=c4n
  new(x3) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
  new(k4) id=and2 st=unready arg1=x3 arg2=c5
  new(x4) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
  new(k5) id=and2 st=unready arg1=x4 arg2=c5n
  new(o1) id=chain st=unready arg1=k1 arg2=table1 arg3="t1: Organic Tree"
  new(o2) id=chain st=unready arg1=k4 arg2=table6 arg3="t6: Chain Mood 1"
  new(o3) id=chain st=unready arg1=k5 arg2=table7 arg3="t7: Chain Mood 2"
  new(o4) id=chain st=unready arg1=k2 arg2=table8 arg3="t8: Chain Mood 3"
  new(o5) id=chain st=unready arg1=k3 arg2=table9 arg3="t9: Chain Mood 4"
  rewrite id=propagate3 arg1=o1 arg2=o2 arg3=o5

To reduce table4:
  new(c1) id=query st=unready arg1="c41: Concurrent"
  new(c2) id=query st=unready arg1="c42: Duration"
  new(c3) id=query st=unready arg1="c43: Impaired"
  new(c4) id=query st=unready arg1="c44: More Than Six Months"
  new(c5) id=query st=unready arg1="c45: Two Weeks"
  new(c1n) id=negative st=unready arg1=c1
  new(c2n) id=negative st=unready arg1=c2
  new(c3n) id=negative st=unready arg1=c3
  new(c4n) id=negative st=unready arg1=c4
  new(c5n) id=negative st=unready arg1=c5
  new(x1) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
  new(k1) id=and2 st=unready arg1=x1 arg2=c4
  new(x2) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
  new(k2) id=and2 st=unready arg1=x2 arg2=c4n
  new(k3) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
  new(k4) id=and3 st=unready arg1=c1 arg2=c2n arg3=c5
  new(k5) id=and3 st=unready arg1=c1 arg2=c2n arg3=c5n
  new(k6) id=and3 st=unready arg1=c1n arg2=c3 arg3=c4
  new(k7) id=and3 st=unready arg1=c1n arg2=c3 arg3=c4n
  new(k8) id=and2 st=unready arg1=c1n arg2=c3n
  new(o1) id=conclude st=unready arg1=k5 arg2="a41: Psychotic Mood Disorder"
  new(o2) id=conclude st=unready arg1=k4 arg2="a42: Schizoaffective Disorder"
  new(r3) id=or2 st=unready arg1=k3 arg2=k8
  new(o3) id=conclude st=unready arg1=r3 arg2="a43: Psychotic Disorder NOS"
  new(o4) id=conclude st=unready arg1=k7 arg2="a43: Schizophreniform Disorder"
  new(o5) id=conclude st=unready arg1=k6 arg2="a44: Schizophrenia"
  rewrite id=propagate3 arg1=o1 arg2=o2 arg3=o5

To reduce table5:
  new(c1) id=query st=unready arg1="c51: Present"
```

```
new(c2) id=query st=unready arg1="c52: Duration"
new(c3) id=query st=unready arg1="c53: Persist"
new(c4) id=query st=unready arg1="c54: Delusion"
new(c1n) id=negative st=unready arg1=c1
new(c2n) id=negative st=unready arg1=c2
new(c3n) id=negative st=unready arg1=c3
new(c4n) id=negative st=unready arg1=c4
new(k1) id=and3 st=unready arg1=c1 arg2=c2 arg3=c4
new(k2) id=and3 st=unready arg1=c1 arg2=c2 arg3=c4n
new(k3) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
new(k4) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(k5) id=and2 st=unready arg1=c1n arg2=c4
new(k6) id=and2 st=unready arg1=c1n arg2=c4n
new(r1) id=or2 st=unready arg1=k1 arg2=k5
new(o1) id=conclude st=unready arg1=r1 arg2="a51: Delusional Disorder"
new(r2) id=or3 st=unready arg1=k2 arg2=k3 arg3=k6
new(o2) id=conclude st=unready arg1=r2 arg2="a52: Psychotic Disorder NOS"
new(o3) id=conclude st=unready arg1=k4 arg2="a53: Psychotic Mood Disorder"
rewrite id=propagate2 arg1=o1 arg2=o2
```

To reduce table6:

```
new(c1) id=query st=unready arg1="c61: Exclusive Depressive Syndrome"
new(c2) id=query st=unready arg1="c62: Major Syndrome NOT Superimposed"
new(c3) id=query st=unready arg1="c63: Two Years of Cyclothymia"
new(c4) id=query st=unready arg1="c64: No Mood Disorder"
new(c1n) id=negative st=unready arg1=c1
new(c2n) id=negative st=unready arg1=c2
new(c3n) id=negative st=unready arg1=c3
new(c4n) id=negative st=unready arg1=c4
new(k2) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(k3) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3n
new(k4) id=and2 st=unready arg1=c1n arg2=c2n
new(o1) id=conclude st=unready arg1=k4 arg2="a61: Bipolar Disorder NOS"
new(r2) id=or2 st=unready arg1=k2 arg2=k3
new(o2) id=conclude st=unready arg1=r2 arg2="a62: Manic Episode"
new(o3) id=conclude st=unready arg1=k2 arg2="a63: Cyclothymia & Bipolar
  Disorder"
new(o4) id=conclude st=unready arg1=k3 arg2="a64: Bipolar Disorder"
rewrite id=propagate2 arg1=o1 arg2=o2
```

To reduce table7:

```
new(c1) id=query st=unready arg1="c71: Superimposed Dysthymia Syndrome"
new(c2) id=query st=unready arg1="c72: Two Years of Dysthymia"
new(c3) id=query st=unready arg1="c73: Depressed Mood"
new(c1n) id=negative st=unready arg1=c1
new(c2n) id=negative st=unready arg1=c2
new(c3n) id=negative st=unready arg1=c3
new(k2) id=and2 st=unready arg1=c1n arg2=c2
new(k3) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(k4) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(o1) id=conclude st=unready arg1=c1 arg2="a71: No Mood Disorder"
new(o2) id=conclude st=unready arg1=k2 arg2="a72: Dysthymia"
new(o3) id=conclude st=unready arg1=k4 arg2="a73: Depressive Disorder NOS"
new(o4) id=conclude st=unready arg1=k3 arg2="a74: Adjusted Disorder"
rewrite id=propagate2 arg1=o1 arg2=o2
```

To reduce table8:

```
new(c1) id=query st=unready arg1="c81: Exclusive Depressive Synd"
new(c2) id=query st=unready arg1="c82: Major Syndrome"
new(c3) id=query st=unready arg1="c83: Two Years of Dysthymia"
new(c1n) id=negative st=unready arg1=c1
new(c2n) id=negative st=unready arg1=c2
new(c3n) id=negative st=unready arg1=c3
new(k2) id=and2 st=unready arg1=c1n arg2=c2
new(k3) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(k4) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(o1) id=conclude st=unready arg1=c1 arg2="a81: No Mood Disorder"
new(r2) id=or2 st=unready arg1=k3 arg2=k4
new(o2) id=conclude st=unready arg1=r2 arg2="a82: Major Depressive Episode"
new(o3) id=conclude st=unready arg1=k2 arg2="a83: Depressive Disorder NOS"
new(o4) id=conclude st=unready arg1=k3 arg2="a84: Dysthymia & Major Depression"
new(o5) id=conclude st=unready arg1=k4 arg2="a85: Major Depression"
rewrite id=propagate3 arg1=o1 arg2=o2 arg3=o5
```

To reduce table9:

```
new(c1) id=query st=unready arg1="c91: Hypomanic Syndrome"
new(c2) id=query st=unready arg1="c92: Numerous Hypomanic Episodes"
new(c1n) id=negative st=unready arg1=c1
new(c2n) id=negative st=unready arg1=c2
new(k2) id=and2 st=unready arg1=c1n arg2=c2
new(k3) id=and2 st=unready arg1=c1n arg2=c2n
new(o1) id=conclude st=unready arg1=c1 arg2="a91: No Mood Disorder"
new(r2) id=or2 st=unready arg1=k2 arg2=k3
new(o2) id=conclude st=unready arg1=r2 arg2="a92: Hypomanic Episode"
new(o3) id=conclude st=unready arg1=k3 arg2="a93: Bipolar Disorder NOS"
new(o4) id=conclude st=unready arg1=k2 arg2="a94: Cyclothymia & Bipolar
  Disorder"
rewrite id=propagate2 arg1=o1 arg2=o2
```

To reduce table10:

```
new(c1) id=query st=unready arg1="c101 : Prominent Delusions"
new(c2) id=query st=unready arg1="c102 : Prominent Hallucinations"
new(c3) id=query st=unready arg1="c103 : Elevated or Expansive Mood"
new(c4) id=query st=unready arg1="c104 : Anxiety"
new(c5) id=query st=unready arg1="c105 : Organic Mental Disorder"
new(c6) id=query st=unready arg1="c106 : Psychoactive Sub Etiology"
new(c7) id=query st=unready arg1="c107 : Before Use"
new(c8) id=query st=unready arg1="c108 : After Use"
new(c1n) id=negative st=unready arg1=c1
new(c2n) id=negative st=unready arg1=c2
new(c3n) id=negative st=unready arg1=c3
new(c4n) id=negative st=unready arg1=c4
new(c5n) id=negative st=unready arg1=c5
new(c6n) id=negative st=unready arg1=c6
new(c7n) id=negative st=unready arg1=c7
new(c8n) id=negative st=unready arg1=c8
new(x1) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
new(k1) id=and2 st=unready arg1=x1 arg2=c4
new(x2) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
new(x3) id=and2 st=unready arg1=c4n arg2=c5
new(k2) id=and2 st=unready arg1=x2 arg2=x3
new(x4) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
new(x5) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x6) id=and2 st=unready arg1=c7 arg2=c8
new(k3) id=and3 st=unready arg1=x4 arg2=x5 arg3=x6
new(x7) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
new(x8) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x9) id=and2 st=unready arg1=c7 arg2=c8n
new(k4) id=and3 st=unready arg1=x7 arg2=x8 arg3=x9
new(x10) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
new(x11) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x12) id=and2 st=unready arg1=c7n arg2=c8
new(k5) id=and3 st=unready arg1=x10 arg2=x11 arg3=x12
new(x13) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
new(x14) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x15) id=and2 st=unready arg1=c7n arg2=c8n
new(k6) id=and3 st=unready arg1=x13 arg2=x14 arg3=x15
new(x16) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3
new(x17) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
new(k7) id=and2 st=unready arg1=x16 arg2=x17
new(x18) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
new(k8) id=and2 st=unready arg1=x18 arg2=c4
new(x19) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
new(x20) id=and2 st=unready arg1=c4n arg2=c5
new(k9) id=and2 st=unready arg1=x19 arg2=x20
new(x21) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
new(x22) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x23) id=and2 st=unready arg1=c7 arg2=c8
new(k10) id=and3 st=unready arg1=x21 arg2=x22 arg3=x23
new(x24) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
new(x25) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x26) id=and2 st=unready arg1=c7 arg2=c8n
new(k11) id=and3 st=unready arg1=x24 arg2=x25 arg3=x26
new(x27) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
new(x28) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x29) id=and2 st=unready arg1=c7n arg2=c8
new(k12) id=and3 st=unready arg1=x27 arg2=x28 arg3=x29
new(x30) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
new(x31) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x32) id=and2 st=unready arg1=c7n arg2=c8n
new(k13) id=and3 st=unready arg1=x30 arg2=x31 arg3=x32
new(x33) id=and3 st=unready arg1=c1 arg2=c2 arg3=c3n
new(x34) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
new(k14) id=and2 st=unready arg1=x33 arg2=x34
new(x35) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
new(k15) id=and2 st=unready arg1=x35 arg2=c4
new(x36) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
new(x37) id=and2 st=unready arg1=c4n arg2=c5
new(k16) id=and2 st=unready arg1=x36 arg2=x37
new(x38) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
new(x39) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x40) id=and2 st=unready arg1=c7 arg2=c8
new(k17) id=and3 st=unready arg1=x38 arg2=x39 arg3=x40
new(x41) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
new(x42) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x43) id=and2 st=unready arg1=c7 arg2=c8n
new(k18) id=and3 st=unready arg1=x41 arg2=x42 arg3=x43
new(x44) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
```

new(x45) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x46) id=and2 st=unready arg1=c7n arg2=c8
new(k19) id=and3 st=unready arg1=x44 arg2=x45 arg3=x46
new(x47) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
new(x48) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x49) id=and2 st=unready arg1=c7n arg2=c8n
new(k20) id=and3 st=unready arg1=x47 arg2=x48 arg3=x49
new(x50) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3
new(x51) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
new(k21) id=and2 st=unready arg1=x50 arg2=x51
new(x52) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(x53) id=and2 st=unready arg1=c4 arg2=c5
new(k22) id=and2 st=unready arg1=x52 arg2=x53
new(x54) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(k23) id=and2 st=unready arg1=x54 arg2=c4
new(x55) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(x56) id=and2 st=unready arg1=c4n arg2=c5
new(k24) id=and2 st=unready arg1=x55 arg2=x56
new(x57) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(x58) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x59) id=and2 st=unready arg1=c7 arg2=c8
new(k25) id=and3 st=unready arg1=x57 arg2=x58 arg3=x59
new(x60) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(x61) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x62) id=and2 st=unready arg1=c7 arg2=c8n
new(k26) id=and3 st=unready arg1=x60 arg2=x61 arg3=x62
new(x63) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(x64) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x65) id=and2 st=unready arg1=c7n arg2=c8
new(k27) id=and3 st=unready arg1=x63 arg2=x64 arg3=x65
new(x66) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(x67) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x68) id=and2 st=unready arg1=c7n arg2=c8n
new(k28) id=and3 st=unready arg1=x66 arg2=x67 arg3=x68
new(x69) id=and3 st=unready arg1=c1 arg2=c2n arg3=c3n
new(x70) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
new(k29) id=and2 st=unready arg1=x69 arg2=x70
new(x71) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(k30) id=and2 st=unready arg1=x71 arg2=c4
new(x72) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(x73) id=and2 st=unready arg1=c4n arg2=c5
new(k31) id=and2 st=unready arg1=x72 arg2=x73
new(x74) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(x75) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x76) id=and2 st=unready arg1=c7 arg2=c8
new(k32) id=and3 st=unready arg1=x74 arg2=x75 arg3=x76
new(x77) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(x78) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x79) id=and2 st=unready arg1=c7 arg2=c8n
new(k33) id=and3 st=unready arg1=x77 arg2=x78 arg3=x79
new(x80) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(x81) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x82) id=and2 st=unready arg1=c7n arg2=c8
new(k34) id=and3 st=unready arg1=x80 arg2=x81 arg3=x82
new(x83) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(x84) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x85) id=and2 st=unready arg1=c7n arg2=c8n
new(k35) id=and3 st=unready arg1=x83 arg2=x84 arg3=x85
new(x86) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3
new(x87) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
new(k36) id=and2 st=unready arg1=x86 arg2=x87
new(x88) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3n
new(k37) id=and2 st=unready arg1=x88 arg2=c4
new(x89) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3n
new(x90) id=and2 st=unready arg1=c4n arg2=c5
new(k38) id=and2 st=unready arg1=x89 arg2=x90
new(x91) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3n
new(x92) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x93) id=and2 st=unready arg1=c7 arg2=c8
new(k39) id=and3 st=unready arg1=x91 arg2=x92 arg3=x93
new(x94) id=and3 st=unready arg1=c1n arg2=c2 arg3=c3n
new(x95) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x96) id=and2 st=unready arg1=c7 arg2=c8n
new(k40) id=and3 st=unready arg1=x94 arg2=x95 arg3=x96
new(x97) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(k41) id=and2 st=unready arg1=x97 arg2=c4
new(x98) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(x99) id=and2 st=unready arg1=c4n arg2=c5
new(k42) id=and2 st=unready arg1=x98 arg2=x99
new(x100) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(x101) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x102) id=and2 st=unready arg1=c7 arg2=c8
new(k43) id=and3 st=unready arg1=x100 arg2=x101 arg3=x102
new(x103) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(x104) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x105) id=and2 st=unready arg1=c7 arg2=c8n
new(k44) id=and3 st=unready arg1=x103 arg2=x104 arg3=x105
new(x106) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(x107) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x108) id=and2 st=unready arg1=c7n arg2=c8
new(k45) id=and3 st=unready arg1=x106 arg2=x107 arg3=x108
new(x109) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3
new(x110) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x111) id=and2 st=unready arg1=c7n arg2=c8n
new(k46) id=and3 st=unready arg1=x109 arg2=x110 arg3=x111
new(x112) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(x113) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
new(k47) id=and2 st=unready arg1=x112 arg2=x113
new(x114) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(k48) id=and2 st=unready arg1=x114 arg2=c4
new(x115) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(x116) id=and2 st=unready arg1=c4n arg2=c5
new(k49) id=and2 st=unready arg1=x115 arg2=x116
new(x117) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(x118) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x119) id=and2 st=unready arg1=c7 arg2=c8
new(k50) id=and3 st=unready arg1=x117 arg2=x118 arg3=x119
new(x120) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(x121) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x122) id=and2 st=unready arg1=c7 arg2=c8n
new(k51) id=and3 st=unready arg1=x120 arg2=x121 arg3=x122
new(x123) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(x124) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x125) id=and2 st=unready arg1=c7n arg2=c8
new(k52) id=and3 st=unready arg1=x123 arg2=x124 arg3=x125
new(x126) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(x127) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6
new(x128) id=and2 st=unready arg1=c7n arg2=c8n
new(k53) id=and3 st=unready arg1=x126 arg2=x127 arg3=x128
new(x129) id=and3 st=unready arg1=c1n arg2=c2n arg3=c3n
new(x130) id=and3 st=unready arg1=c4n arg2=c5n arg3=c6n
new(k54) id=and2 st=unready arg1=x129 arg2=x130
new(x131) id=or3 st=unready arg1=k1 arg2=k2 arg3=k3
new(x132) id=or3 st=unready arg1=k4 arg2=k5 arg3=k6
new(x133) id=or3 st=unready arg1=k7 arg2=k8 arg3=k9
new(x134) id=or3 st=unready arg1=k10 arg2=k11 arg3=k12
new(x135) id=or3 st=unready arg1=k13 arg2=k14 arg3=k15
new(x136) id=or3 st=unready arg1=k16 arg2=k17 arg3=k18
new(x137) id=or3 st=unready arg1=k19 arg2=k20 arg3=k21
new(x138) id=or3 st=unready arg1=k22 arg2=k23 arg3=k24
new(x139) id=or3 st=unready arg1=k25 arg2=k26 arg3=k27
new(x140) id=or2 st=unready arg1=k28 arg2=k29
new(x141) id=or3 st=unready arg1=x131 arg2=x132 arg3=x133
new(x142) id=or3 st=unready arg1=x134 arg2=x135 arg3=x136
new(x143) id=or3 st=unready arg1=x137 arg2=x138 arg3=x139
new(x144) id=or3 st=unready arg1=x141 arg2=x142 arg3=x143
new(r1) id=or2 st=unready arg1=x144 arg2=x140
new(o1) id=conclude st=unready arg1=r1 arg2="a101 : Org Delusional Syndrome"
new(x145) id=or3 st=unready arg1=k1 arg2=k2 arg3=k3
new(x146) id=or3 st=unready arg1=k4 arg2=k5 arg3=k6
new(x147) id=or3 st=unready arg1=k7 arg2=k8 arg3=k9
new(x148) id=or3 st=unready arg1=k10 arg2=k11 arg3=k12
new(x149) id=or3 st=unready arg1=k13 arg2=k14 arg3=k30
new(x150) id=or3 st=unready arg1=k31 arg2=k36 arg3=k37
new(x151) id=or3 st=unready arg1=k38 arg2=k39 arg3=k40
new(x152) id=or3 st=unready arg1=x145 arg2=x146 arg3=x147
new(x153) id=or3 st=unready arg1=x148 arg2=x149 arg3=x150
new(r2) id=or3 st=unready arg1=x152 arg2=x153 arg3=x151
new(o2) id=conclude st=unready arg1=r2 arg2="a102 : Org Hallucinosis"
new(x154) id=or3 st=unready arg1=k1 arg2=k2 arg3=k3
new(x155) id=or3 st=unready arg1=k4 arg2=k5 arg3=k6
new(x156) id=or3 st=unready arg1=k7 arg2=k15 arg3=k16
new(x157) id=or3 st=unready arg1=k17 arg2=k18 arg3=k19
new(x158) id=or3 st=unready arg1=k20 arg2=k21 arg3=k30
new(x159) id=or3 st=unready arg1=k31 arg2=k32 arg3=k36
new(x160) id=or3 st=unready arg1=k41 arg2=k42 arg3=k43
new(x161) id=or3 st=unready arg1=k44 arg2=k45 arg3=k46
new(x162) id=or3 st=unready arg1=x154 arg2=x155 arg3=x156
new(x163) id=or3 st=unready arg1=x157 arg2=x158 arg3=x159
new(x164) id=or3 st=unready arg1=x160 arg2=x161 arg3=k47
new(r3) id=or3 st=unready arg1=x162 arg2=x163 arg3=x164
new(o3) id=conclude st=unready arg1=r3 arg2="a103 : Org Mood Syndrome"
new(x165) id=or3 st=unready arg1=k1 arg2=k8 arg3=k15
new(x166) id=or3 st=unready arg1=k22 arg2=k23 arg3=k30
new(x167) id=or3 st=unready arg1=k32 arg2=k37 arg3=k41
new(x168) id=or3 st=unready arg1=x165 arg2=x166 arg3=x167
new(r4) id=or2 st=unready arg1=x168 arg2=k48

```
new(o4) id=conclude st=unready arg1=r4 arg2="a104 : Org Anxiety Syndrome"
new(x169) id=or3 st=unready arg1=k2 arg2=k9 arg3=k16
new(x170) id=or3 st=unready arg1=k31 arg2=k38 arg3=k42
new(r5) id=or3 st=unready arg1=x169 arg2=x170 arg3=k49
new(o5) id=conclude st=unready arg1=r5 arg2="a105 : No Addit'al Diagnosis
  Given"
new(x171) id=or3 st=unready arg1=k4 arg2=k6 arg3=k7
new(x172) id=or3 st=unready arg1=k11 arg2=k13 arg3=k14
new(x173) id=or3 st=unready arg1=k18 arg2=k20 arg3=k21
new(x174) id=or3 st=unready arg1=k24 arg2=k26 arg3=k28
new(x175) id=or3 st=unready arg1=k29 arg2=k33 arg3=k35
new(x176) id=or3 st=unready arg1=k36 arg2=k40 arg3=k44
new(x177) id=or3 st=unready arg1=k46 arg2=k47 arg3=k51
new(x178) id=or2 st=unready arg1=k53 arg2=k54
new(x179) id=or3 st=unready arg1=x171 arg2=x172 arg3=x173
new(x180) id=or3 st=unready arg1=x174 arg2=x175 arg3=x176
new(x181) id=or2 st=unready arg1=x177 arg2=x178
new(r6) id=or3 st=unready arg1=x179 arg2=x180 arg3=x181
new(o6) id=conclude st=unready arg1=r6 arg2="a106 : Org Mental Disorder
  NOS"
new(x182) id=or3 st=unready arg1=k4 arg2=k10 arg3=k11
new(x183) id=or3 st=unready arg1=k17 arg2=k18 arg3=k25
new(x184) id=or3 st=unready arg1=k26 arg2=k32 arg3=k33
new(x185) id=or3 st=unready arg1=k39 arg2=k40 arg3=k43
new(x186) id=or3 st=unready arg1=k44 arg2=k50 arg3=k51
new(x187) id=or3 st=unready arg1=x182 arg2=x183 arg3=x184
new(x188) id=or2 st=unready arg1=x185 arg2=x186
new(r7) id=or2 st=unready arg1=x187 arg2=x188
new(o7) id=conclude st=unready arg1=r7 arg2="a107 : Psy Sub-induced Intoxic."
new(x189) id=or3 st=unready arg1=k5 arg2=k10 arg3=k12
new(x190) id=or3 st=unready arg1=k17 arg2=k19 arg3=k25
new(x191) id=or3 st=unready arg1=k27 arg2=k32 arg3=k34
new(x192) id=or3 st=unready arg1=k39 arg2=k43 arg3=k45
new(x193) id=or2 st=unready arg1=k50 arg2=k52
new(x194) id=or3 st=unready arg1=x189 arg2=x190 arg3=x191
new(x195) id=or2 st=unready arg1=x192 arg2=x193
new(r8) id=or2 st=unready arg1=x194 arg2=x195
new(o8) id=conclude st=unready arg1=r8 arg2="a108 : Psychoactive Sub
  Withdrawal"
new(x196) id=propagate3 st=unready arg1=o1 arg2=o2 arg3=o5
new(x197) id=propagate3 st=unready arg1=o6 arg2=o7 arg3=o8
rewrite id=propagate2 arg1=x196 arg2=x197

to reduce query:
 write string "Query"
 write string arg1
 write string "?"
 read char x
 read char y
 k=0
 if (x=89) or (x=121)
  k=1
  rewrite id=true st=data arg1=0 arg2=arg1
 if (x=78) or (x=110)
  k=1
  rewrite id=false st=data arg1=0 arg2=arg1
 if k=0
  rewrite id=query st=ready arg1=arg1

to reduce negative:
 require arg1
 if (arg1.id)=true
  rewrite id=false st=data arg1=arg1.arg1 arg2=0
 else
  rewrite id=true st=data arg1=arg1.arg1 arg2=0

to reduce and2:
 require arg1
 require arg2
 if ((arg1.id)=true) and ((arg2.id)=true)
  rewrite id=true st=data arg1=arg1 arg2=arg2
 else
  rewrite id=false st=data arg1=arg1 arg2=arg2

to reduce and3:
 require arg1
 require arg2
 require arg3
 if ((arg1.id)=true) and ((arg2.id)=true) and ((arg3.id)=true)
  rewrite id=true st=data arg1=arg1 arg2=arg2 arg3=arg3
 else
  rewrite id=false st=data arg1=arg1 arg2=arg2 arg3=arg3

to reduce or2:
 require arg1
 require arg2
 if ((arg1.id)=true) or ((arg2.id)=true)
  rewrite id=true st=data arg1=arg1 arg2=arg2
 else
  rewrite id=false st=data arg1=arg1 arg2=arg2

to reduce or3:
 require arg1
 require arg2
 require arg3
 if ((arg1.id)=true) or ((arg2.id)=true) or ((arg3.id)=true)
  rewrite id=true st=data arg1=arg1 arg2=arg2 arg3=arg3
 else
  rewrite id=false st=data arg1=arg1 arg2=arg2 arg3=arg3

to reduce conclude:
 require arg1
 if ((arg1.id)=true)
  write string "Conclusion"
  write string arg2
  write string "."
  write char 10
  rewrite id=true st=data arg1=arg1 arg2=arg2
 else
  rewrite id=false st=data arg1=arg1 arg2=arg2

to reduce iswanted:
 write string "Are you interested in "
 write string arg1
 write string "?"
 read char x
 read char y
 if (x=89) or (x=121)
  rewrite id=wakeup arg1=arg2
 else
  if (x=78) or (x=110)
   rewrite id=useless st=data rc=0
  else
   rewrite id=iswanted arg1=arg1 arg2=arg2

to reduce chain:
 require arg1
 if (arg1.id)=true
  write string "[Chaining table "
  write string arg3
  write string "]"
  write char 10
  rewrite id=arg2
 else
  rewrite id=useless st=data rc=0

to reduce wakeup:
 require arg1
 rewrite id=useless st=data rc=0

to reduce propagate2:
 require arg1
 require arg2
 rewrite id=useless st=data rc=0

to reduce propagate3:
 require arg1
 require arg2
 require arg3
 rewrite id=useless st=data rc=0
```