Empirical Model using Expert System Techniques in Hardware Failure of a System during Backup of Data

Abdullah Saad AlMalaise Al Ghamdi† , Syed Mutiullah Hussaini† † and Abdul Khadar Jilani†††

†Dept., Chair IS @ King Abdul Aziz University ,Jeddah –KSA.
††Faculty Member CS & IS Dept.,@ Umm Al-Qura University, Makkah –KSA.
†††Faculty Member CS @ Majmoah University, Al-Majmaah.

Summary
The using of the computer is widely spread today but the knowledge in the fault diagnosis is very limited. Thus, the fault diagnosis of the computer components is very important. Obviously, the risk of hardware failure is the most commonly talked-about reason to perform backups. Indeed, nothing will jolt someone into realizing the importance of backups more than an unrecoverable hard disk failure. Since the hard disk stores your main programs and data, it is the hardware whose failure hurts the most. It is also what gets the most attention, and rightly so. However, there are other hardware problems that can cause permanent data loss and some of these can be rather hard to figure out, since they don't seem like they should be responsible for the problem. For instance, memory error with so many systems today running without error detection or correction on their system memory, there is a chance of a memory error corrupting the data on the hard disk. It is rare for it to happen, but it does happen [1]. Accordingly, the need of creating an expert system to diagnose these faults is increasing. Here in this paper the design of an expert system that aims to help the user in diagnosing the failure in motherboard, RAM and CPU is represented.

Key words:
Expert system, Artificial Intelligence, Hardware.

1. Introduction

Artificial Intelligence (AI) is the area of computer science focusing on creating machines that can engage on behaviors that human consider intelligent. The dream of smart machines is becoming a reality after 50 years of research into AI programming techniques [2].

Expert system is an intelligent computer program that uses knowledge and inference procedures to solve problem which is difficult enough to require significant human expertise for their solution [2]. Expert system seeks and utilizes relevant information from their human users and from available knowledge bases in order to make recommendations. The expert system can store heuristic knowledge so, the user can interact with a computer to solve a certain problem [3].

Diagnosing a problem is very difficult because of the complexity of the systems and the constant change. The systems are complex functionally as a whole, and additional complexity is introduced through the interactions of heterogeneous components. The change comes about both internally through hardware and software updates and new components, and externally through environment. The behavior of the system is changing at different rates due to load, traffic mix and configuration [4].

A malfunction within the electronic circuits or electromechanical components of a computer system. Recovery from a hardware failure requires repair or replacement of the offending part [5]. Hardware devices can fail, but many drivers assume they do not. When confronted with real devices that misbehave, these assumptions can lead to driver or system failures. While major operating system and device vendors recommend that drivers detect and recover from hardware failures. We find that there are many drivers that will crash or hang when a device fails. Such bugs cannot easily be detected by regular stress testing because the failures are induced by the device and not the software load [6].

2. Literature review:

Mohd Daud Bin Isa and Othman Bin Sidek in 2000 developed an expert system called ‘PCDIASHOOT’ to assist and advice new technician or computer user to diagnosis and troubleshoot IBM PC compatible or clone. The developed system capabilities to diagnosis and troubleshoot various type of computer problem include of hardware and software problem. The system capabilities to train new technician and computer user to diagnosis and troubleshoot the computer IBM PC without expert help. The system also has capabilities to diagnosis and troubleshooting the computer with systematic and accurate way. As a result, they will produce the consistent result with the same consultation process. Also, the system would be easy to use, reliable and believable. It is because of using Kappa-PC that supports various component of expert system and has capabilities to prevent conflict resolution within the rule and rules itself during inferences

Manuscript received February 5, 2013
Manuscript revised February 20, 2013

process using forward chaining with depth first mod and implicit priority [3].

In 2009, a system model of fault diagnosis of the fire control computer and sensor subsystem, the method and process of fault diagnosis was introduced. In this expert system, they used the object-oriented production rule to represent the knowledge, which solves the bottleneck problem of the diagnostic knowledge acquisition effectively. The inferential process begins with the abnormal event and finally finds all of the possible faults and the faulty component. For some possible faulty components, which have large number of fault samples, the neural network model can be used to diagnose. The simulation training results show that the fault diagnosis expert system based on the combination of fault tree and neural network is rational and effective in fault diagnosis of the fire control system, realizes perfectly the combination of new knowledge and old one, and can grasp the state of systems dynamically [7].

The fuzzy expert system is suitable for solving an inaccurate and subjective problem as encountered by system test case selection. Diagnostic expert system for computer failures uses different knowledge representations and reasoning algorithms. An important branch of the research is dealing with modeling diagnostic rules. For example, fuzzy logic methods could supplement diagnostic of measurement lines and devices of computer system [8].

Guo-Zhong Zhou in 1993 presented an approach to fault diagnosis for large scale power systems is presented based on hierarchical distributed neural networks. Several independent neural networks in the same level are used to diagnose the faults within a substation, One higher level neural network which makes use of some outputs from lower level as its inputs is used to diagnose the faults in transmission lines A combined gradient learning algorithm with advantages of both gradient and conjugate gradient algorithms is employed for training the neural network.[8] This learning algorithm converges faster than the error back Propagation algorithm. Comparisons between the proposed approach and the single neural network approach are made for a model substation and a model Dower system. Simulation results show that this approach is very encouraging [9].

Expert systems were introduced by researchers in the Stanford Heuristic Programming Project, including the "father of expert systems" Edward Feigenbaum, with the Dendral and Mycin systems. Principal contributors to the technology were Bruce Buchanan, Edward Shortliffe, Randall Davis, William vanMelle, Carli Scott and others at Stanford. Expert systems were among the first truly successful forms of AI software.

Research is also very active in France, where researchers focus on the automation of reasoning and logic engines. The French Prolog computer language, designed in 1972, marks a real advance over expert systems like Dendral or Mycin: it is a shell, that's to say a software structure ready to receive any expert system and to run it. It integrates an engine using First-Order logic, with rules and facts. It's a tool for mass production of expert systems and was the first operational declarative language, later becoming the best-selling IA language in the world. However Prolog is not particularly user friendly and is an order of logic away from human logic.

In the 1980s, expert systems proliferated as they were recognized as a practical tool for solving real-world problems. Universities offered expert system courses and two thirds of the Fortune 1000 companies applied the technology in daily business activities. Interest was international with the Fifth Generation Computer Systems project in Japan and increased research funding in Europe. Growth in the field continued into the 1990s.

The development of expert systems was aided by the development of the symbolic processing languages Lisp and Prolog. To avoid re-inventing the wheel, expert system shells were created that had more specialized features for building large expert systems.

In 1981 the first IBM PC was introduced, with MS-DOS operating system. Its low price started to multiply users and opened a new market for computing and expert systems. In the 80's the image of IA was very good and people believed it would succeed within a short time. Many companies began to market expert systems shells from universities, renamed "generators" because they added to the shell a tool for writing rules in plain language and thus, theoretically, allowed to write expert systems without a programming language or any other software. The best known: Guru (USA) inspired by Mycin, Personal Consultant Plus (USA), Nexpert Object (developed by Neuron Data, company founded in California by three French), Genesia (developed by French public company Electricité de France and marketed by Steria), VP Expert (USA). But eventually the tools were only used in research projects. They did not penetrate the business market, showing that AI technology was not mature.

In 1986, a new expert system generator for PCs appeared on the market, derived from the French academic research: Intelligence Services, old by GSI-TECSI software company. This software showed a radical innovation: it used propositional logic ("Zeroth order logic") to execute expert systems, reasoning on a knowledge base written with everyday language rules, producing explanations and detecting logic contradictions between the facts. It was the first tool showing the AI defined by Edward Feigenbaum in his book about the Japanese Fifth Generation, Artificial Intelligence and Japan's Computer Challenge to the World (1983): "The machines will have reasoning power: they will automatically engineer vast amounts of knowledge to serve whatever purpose humans propose, from medical diagnosis to product design, from management decisions to education", "The reasoning animal has, perhaps inevitably, fashioned the reasoning machine", "the reasoning power of these machines matches or exceeds the reasoning power of the humans who instructed them and, in some cases, the reasoning power of any human performing such tasks". Intelligence Service was in fact "Pandora" (1985) a software developed for their thesis by two academic students of Jean-Louis Laurière, one of the most famous and prolific French AI researcher. Unfortunately, as this software was not developed by his own IT developers, GSI-TECSI was unable to make it evolve. Sales became scarce and marketing stopped after a few years [10].

3. Expert System in Hardware Failure:

An expert system: is a computer system that emulates the decision-making ability of a human expert. Expert systems are designed to solve complex problems by reasoning about knowledge, like an expert, and not by following the procedure of a developer as is the case in conventional programming. The first expert systems were created in the 1970s and then proliferated in the 1980s. Expert systems were among the first truly successful forms of AI software [10].

An expert system has a unique structure, different from traditional programs. It is divided into two parts, one fixed, independent of the expert system: the inference engine, and one variable: the knowledge base.

The following flowchart was used to build the expert system. By keeping a track of the chart we have created the meant expert system to solve the problems that occur in the Motherboard, CPU and RAM failure[12].

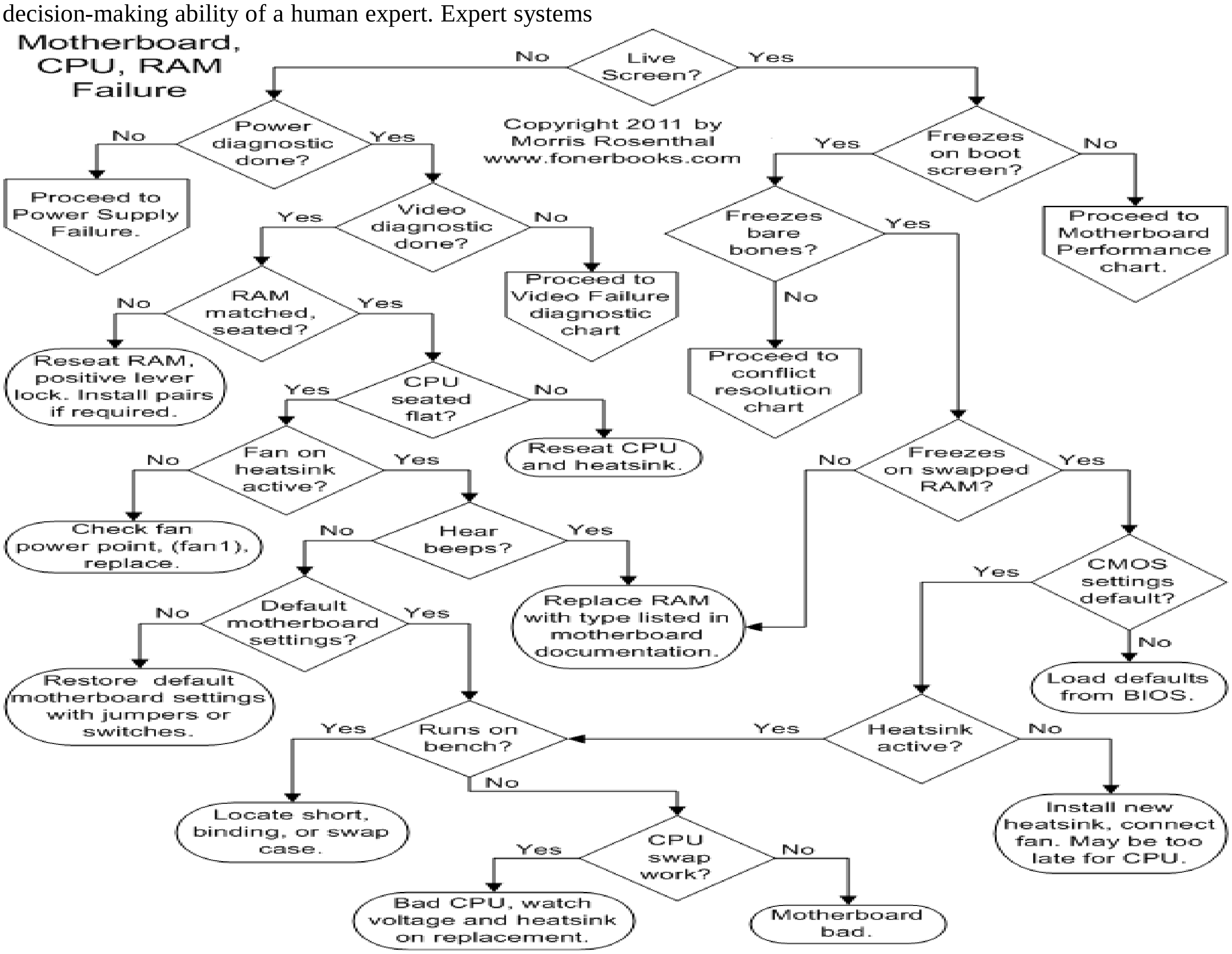


Figure 1: The flowchart used to build the expert system hardware failure.

In brief, the previous flowchart shows the way the program follows to check for the failure so that it can provide the appropriate solution. All knows the types of damages that could be caused for the hardware where a lot of these damages are caused by the external environment. Here are some common hardware failures: RAM Failures, Power Connector, Hard Disk, Overheating, LCD Failures, Motherboard and USB Box.

Most motherboard related failures are due to the "On-board" regulated supplies and component failure within those circuits. The on-board power supply circuit had partially failed and was overloading subsequent components else the problem would be with the capacitors which are defective in the first place. A motherboard failure on a laptop that is out of warranty would usually mean that it's time for a new laptop. The price of a new motherboard is usually higher than the current value of the laptop. Bad RAM is somehow harder to diagnose as similar symptoms may be caused by software problems, other hardware problems or even motherboard failure. However if you experience any of these symptoms, users should check for bad RAM before attempting any other troubleshooting.

Hardware and software work together to form a fully functional system, theoretically. However, rarely do you get a fully functional system all the time. There will almost certainly be malfunction of hardware, be it within the electronic circuits or even the whole component itself. Sometimes, the originating factor of the hardware failure is not the components of the system itself, but due to outside factors such as environmental disasters like fire, earthquakes and lightning storm. Recovery of failed hardware components is not a big problem by itself. It is basically identifying and replacing the problematic component. However, hardware failures are most deadly, when it affects daily routine and affecting critical personal or business data. This is especially true of the most important component of a computer system when it comes to storage of data, the hard disk [11]. The process of building an expert system is commonly known as knowledge engineering. This implies knowledge acquisition from a human or other source and coding it into the knowledge based of the expert system. The main phases in the knowledge engineering process are:

① The dialog process represented in figure 2 is similar to the task of a system designer discussing the requirements of the program with the client, in conventional programming. After acquiring knowledge engineer has to explicitly code it into the expert system knowledge base.

② After the coding stage, the human expert evaluates the expert system and gives feedback/critique to the knowledge engineer.

③ The knowledge engineer alerts the knowledge based in order to reflect the human experts comments. The process will iterate until the human expert finds the expert system satisfactory for the envisaged purpose

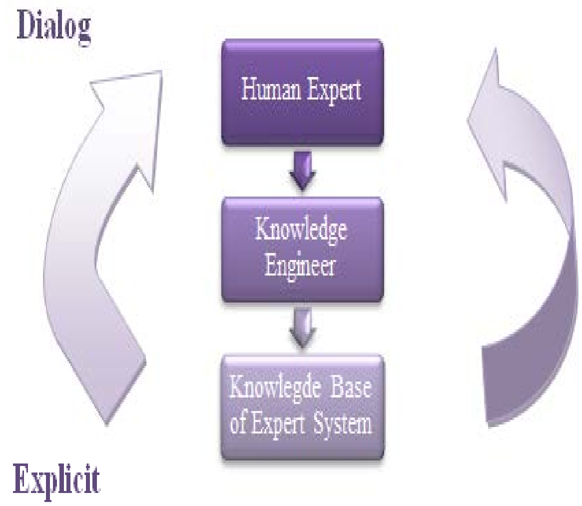


Figure 2: Basic concept of an expert system.

4. Application field:

Expert systems address areas where combinatory is enormous:

- Highly interactive or conversational applications, IVR, voice server, chatterbot
- Fault diagnosis, medical diagnosis
- Decision support in complex systems, process control, interactive user guide
- Educational and tutorial software
- Logic simulation of machines or systems
- Knowledge management
- Constantly changing software.

They can also be used in software engineering for rapid prototyping applications (RAD). Indeed, the expert system quickly developed in front of the expert shows him if the future application should be programmed. Indeed, any program contains expert knowledge and classic programming always begins with an expert interview. A program written in the form of expert system receives all the specific benefits of expert system, among others things it can be developed by anyone without computer training and without programming languages. But this solution has a defect: expert system runs slower than a traditional program because he consistently "thinks" when in fact classic software just follows paths traced by the programmer.

5. Examples of applications:

Expert systems are designed to facilitate tasks in the fields of accounting, medicine, process control, financial service, production, human resources, among others.

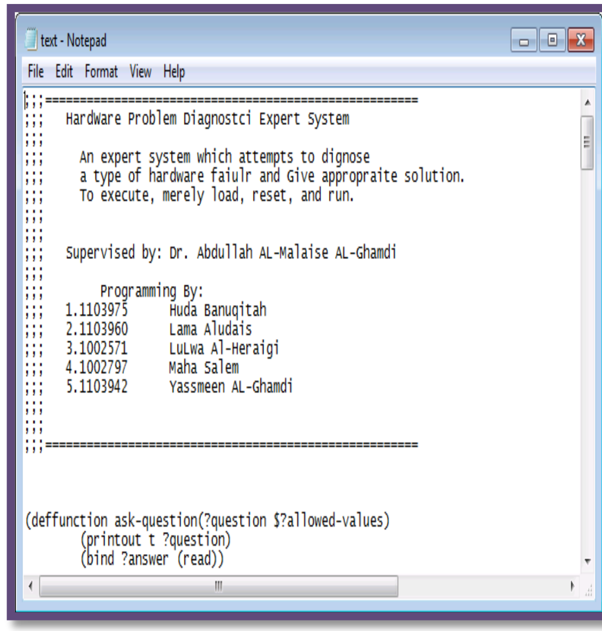
Typically, the problem area is complex enough that a simpler traditional algorithm cannot provide a proper solution. The foundation of a successful expert system depends on a series of technical procedures and development that may be designed by technicians and related experts. As such, expert systems do not typically provide a definitive answer, but provide probabilistic recommendations.

An example of the application of expert systems in the financial field is expert systems for mortgages. Loan departments are interested in expert systems for mortgages because of the growing cost of labour, which makes the handling and acceptance of relatively small loans less profitable. They also see a possibility for standardized, efficient handling of mortgage loan by applying expert systems, appreciating that for the acceptance of mortgages there are hard and fast rules which do not always exist with other types of loans. Another common application in the financial area for expert systems is in trading recommendations in various marketplaces. These markets involve numerous variables and human emotions which may be impossible to deterministically characterize, thus expert systems based on the rules of thumb from experts and simulation data are used. Expert system of this type can range from ones providing regional retail recommendations, like Wishabi, to ones used to assist monetary decisions by financial institutions and governments.

Another 1970s and 1980s application of expert systems, which we today would simply call AI, was in computer games. For example, the computer baseball games Earl Weaver Baseball and Tony La Russa Baseball each had highly detailed simulations of the game strategies of those two baseball managers. When a human played the game against the computer, the computer queried the Earl Weaver or Tony La Russa Expert System for a decision on what strategy to follow. Even those choices where some randomness was part of the natural system (such as when to throw a surprise pitch-out to try to trick a runner trying to steal a base) were decided based on probabilities supplied by Weaver or La Russa. Today we would simply say that "the game's AI provided the opposing manager's strategy" [10].

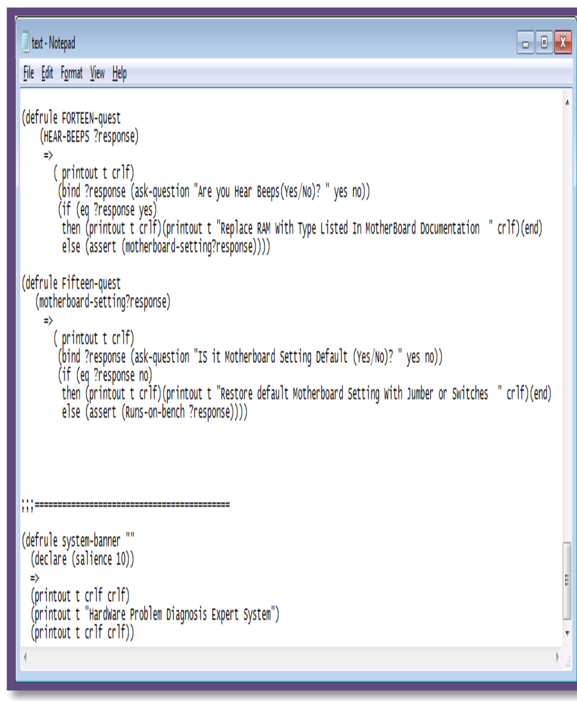
6. The Code:

After finding the solution, it was programmed with the CLIPS programming language. Snapshots of the program were taken as below.



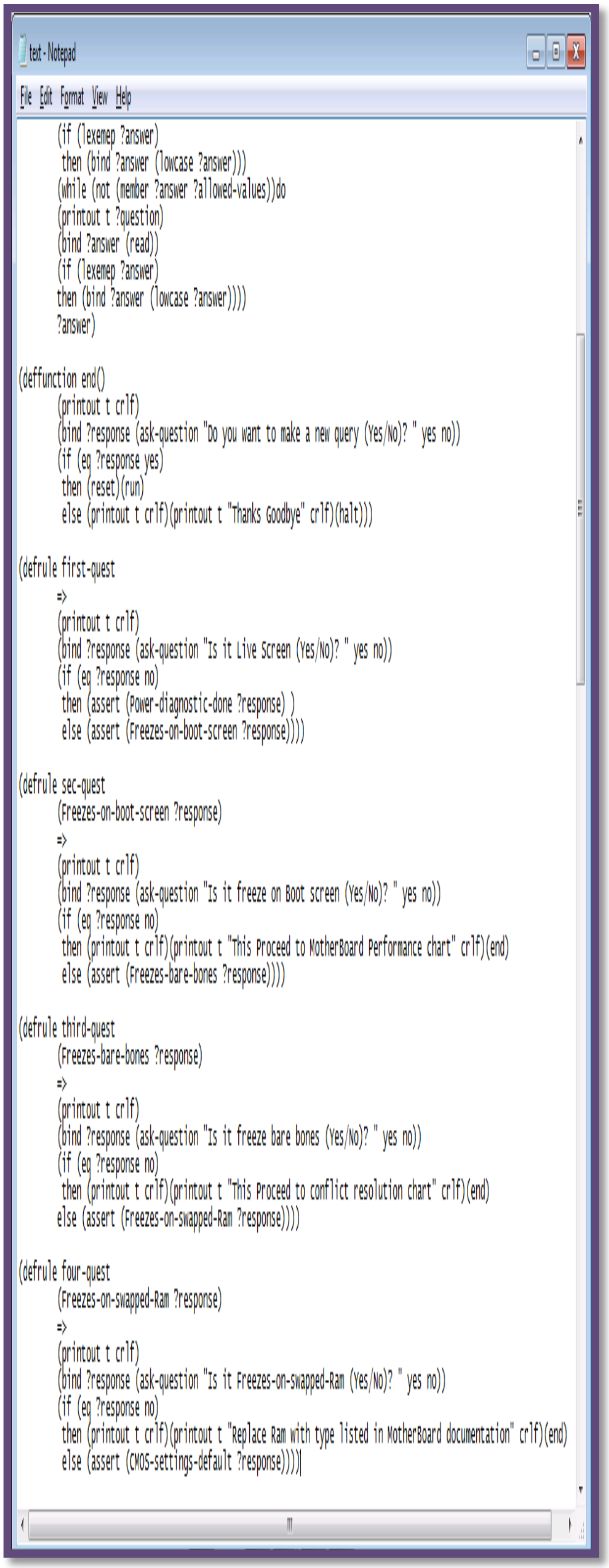
```
;;;=====================================================
;;;  Hardware Problem Diagnostci Expert System
;;;
;;;    An expert system which attempts to dignose
;;;    a type of hardware faiulr and Give appropraite solution.
;;;    To execute, merely load, reset, and run.
;;;
;;;
;;;
;;;  Supervised by: Dr. Abdullah AL-Malaise AL-Ghamdi
;;;
;;;       Programming By:
;;;  1.1103975      Huda Banuqitah
;;;  2.1103960      Lama Aludais
;;;  3.1002571      LuLwa Al-Heraigi
;;;  4.1002797      Maha Salem
;;;  5.1103942      Yassmeen AL-Ghamdi
;;;
;;;
;;;=====================================================


(deffunction ask-question(?question $?allowed-values)
       (printout t ?question)
       (bind ?answer (read))
```

Figure 3: part1 of code.



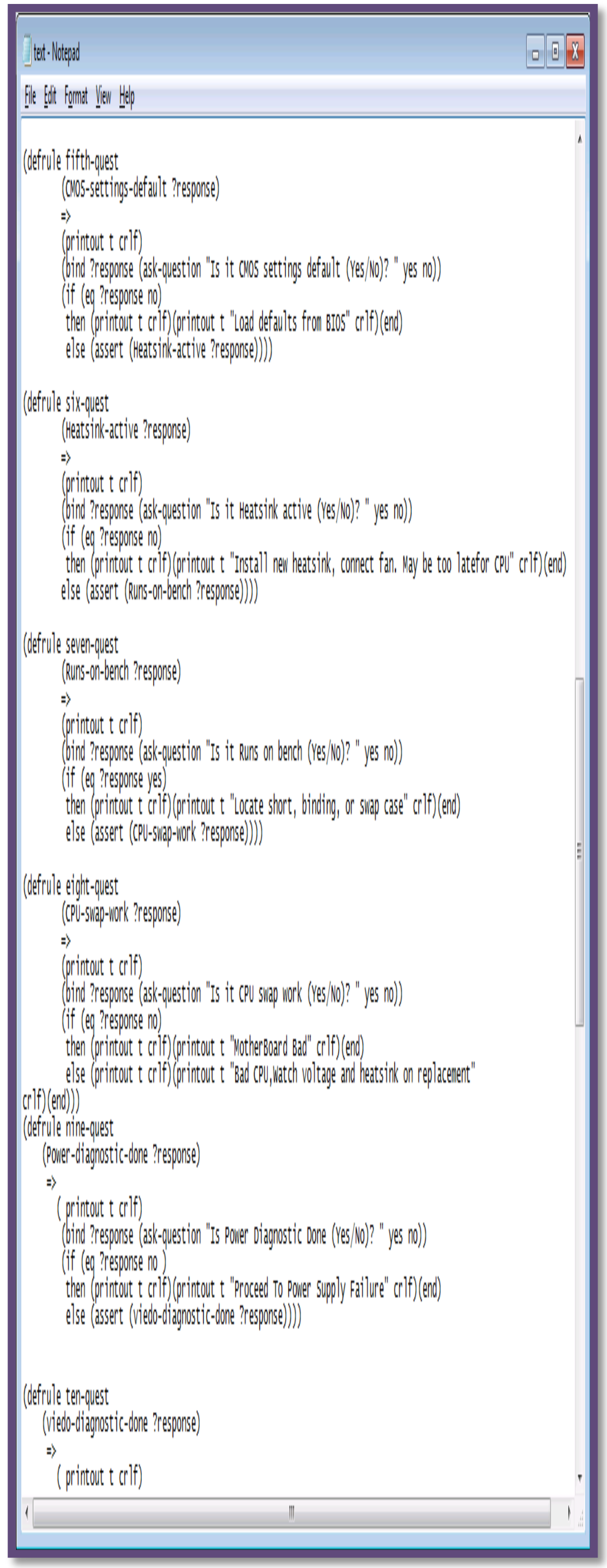
```
(defrule FORTEEN-quest
   (HEAR-BEEPS ?response)
    =>
     ( printout t crlf)
      (bind ?response (ask-question "Are you Hear Beeps(Yes/No)? " yes no))
      (if (eq ?response yes)
       then (printout t crlf)(printout t "Replace RAM With Type Listed In MotherBoard Documentation  " crlf)(end)
       else (assert (motherboard-setting?response))))

(defrule Fifteen-quest
  (motherboard-setting?response)
    =>
     ( printout t crlf)
      (bind ?response (ask-question "IS it Motherboard Setting Default (Yes/No)? " yes no))
      (if (eq ?response no)
       then (printout t crlf)(printout t "Restore default Motherboard Setting With Jumber or Switches  " crlf)(end)
       else (assert (Runs-on-bench ?response))))



;;;========================================


(defrule system-banner ""
  (declare (salience 10))
  =>
  (printout t crlf crlf)
  (printout t "Hardware Problem Diagnosis Expert System")
  (printout t crlf crlf))
```

Figure 4: part2 of code.



```
text - Notepad
File Edit Format View Help
     (if (lexemep ?answer)
      then (bind ?answer (lowcase ?answer)))
     (while (not (member ?answer ?allowed-values))do
     (printout t ?question)
     (bind ?answer (read))
     (if (lexemep ?answer)
     then (bind ?answer (lowcase ?answer))))
     ?answer)

(deffunction end()
     (printout t crlf)
     (bind ?response (ask-question "Do you want to make a new query (Yes/No)? " yes no))
     (if (eq ?response yes)
      then (reset)(run)
      else (printout t crlf)(printout t "Thanks Goodbye" crlf)(halt)))

(defrule first-quest
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it Live Screen (Yes/No)? " yes no))
     (if (eq ?response no)
      then (assert (Power-diagnostic-done ?response) )
      else (assert (Freezes-on-boot-screen ?response))))

(defrule sec-quest
     (Freezes-on-boot-screen ?response)
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it freeze on Boot screen (Yes/No)? " yes no))
     (if (eq ?response no)
      then (printout t crlf)(printout t "This Proceed to MotherBoard Performance chart" crlf)(end)
      else (assert (Freezes-bare-bones ?response))))

(defrule third-quest
     (Freezes-bare-bones ?response)
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it freeze bare bones (Yes/No)? " yes no))
     (if (eq ?response no)
      then (printout t crlf)(printout t "This Proceed to conflict resolution chart" crlf)(end)
     else (assert (Freezes-on-swapped-Ram ?response))))

(defrule four-quest
     (Freezes-on-swapped-Ram ?response)
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it Freezes-on-swapped-Ram (Yes/No)? " yes no))
     (if (eq ?response no)
      then (printout t crlf)(printout t "Replace Ram with type listed in MotherBoard documentation" crlf)(end)
      else (assert (CMOS-settings-default ?response))))
```

Fig 5 . Part 3 of code



```
text - Notepad
File Edit Format View Help

(defrule fifth-quest
     (CMOS-settings-default ?response)
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it CMOS settings default (Yes/No)? " yes no))
     (if (eq ?response no)
      then (printout t crlf)(printout t "Load defaults from BIOS" crlf)(end)
      else (assert (Heatsink-active ?response))))

(defrule six-quest
     (Heatsink-active ?response)
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it Heatsink active (Yes/No)? " yes no))
     (if (eq ?response no)
      then (printout t crlf)(printout t "Install new heatsink, connect fan. May be too latefor CPU" crlf)(end)
     else (assert (Runs-on-bench ?response))))

(defrule seven-quest
     (Runs-on-bench ?response)
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it Runs on bench (Yes/No)? " yes no))
     (if (eq ?response yes)
      then (printout t crlf)(printout t "Locate short, binding, or swap case" crlf)(end)
      else (assert (CPU-swap-work ?response))))

(defrule eight-quest
     (CPU-swap-work ?response)
     =>
     (printout t crlf)
     (bind ?response (ask-question "Is it CPU swap work (Yes/No)? " yes no))
     (if (eq ?response no)
      then (printout t crlf)(printout t "MotherBoard Bad" crlf)(end)
      else (printout t crlf)(printout t "Bad CPU,Watch voltage and heatsink on replacement"
crlf)(end)))
(defrule nine-quest
   (Power-diagnostic-done ?response)
   =>
    ( printout t crlf)
     (bind ?response (ask-question "Is Power Diagnostic Done (Yes/No)? " yes no))
     (if (eq ?response no )
      then (printout t crlf)(printout t "Proceed To Power Supply Failure" crlf)(end)
      else (assert (viedo-diagnostic-done ?response))))


(defrule ten-quest
   (viedo-diagnostic-done ?response)
   =>
    ( printout t crlf)
```

Fig 6 Part 4 of code

7. Future work:

Today, computers have become an integral part of everyday life, used for a variety of reasons at home, in workplace and at schools. Of course, almost every computer user encounters a problem occasionally, whether it is the disaster of a crashing hard drive or the annoyance of a forgotten password. The explosive use of computers has created a high demand for specialists to provide advice to users, as well as, for day-to-day administration, maintenance and the support of computer systems and networks. For that we have built our simple expert system as a troubleshooter that interpret problems and provide technical support for most hardware problem.

Mainly, building such a system is required for current needs but it also recommended to enhance the system for future use. It is advised to create friendly interfaces and connecting various expert system codes together. In addition, it is also recommended to make the system solve more hardware features, or even expand it to solve software problems and show the relationship between the failure that had occurred in both hardware and software. Another thing that can be done in that future

is deploying such a program online to rise the usability of the system and provide facilities for users.

8. Conclusion

Our simple expert system provides troubleshooting procedures on how to diagnose hardware problems CPu, Motherboard and RAM. We acquire knowledge and collect the information through electronics books, websites and PDFs. After collecting required information we have applied the rules on the previous knowledge to ensure that the results were correct. Clearly, we have built our expert system by CLIPS programming language which is the best language for building a rule based expert system. Obviously, our goal from building this simple expert system is to help computer users to fix some basic hardware issues or even to perform more extensive troubleshooting before they contacthttp://www.cisco.com/en/US/support/tsd_cisco_worldwide_contacts.html with help desk or technicians. Moreover, the system saves valuable time in searching for solutions and takes the computer to maintain centers. Finally, We gave some advices that will be added to the hardware failure expert system for better future use.

References

[1] http://www.pcguide.com/care/bu/risksHardware-c.html

[2] Samy S. Abu Naser, 1Abu Zaiter A. Ola, 2005-2008, AN EXPERT SYSTEM FOR DIAGNOSING EYE DISEASES USING CLIPS, Journal of Theoretical and Applied Information Technology.

[3] Mohd Daud Bin Isa, Othman Bin Sidek, 2000, PC Diagnosis And Troubleshooting Expert System Based On Computer Response During Power On Self Test (Post)-PCDIASHOOT,0-7803-6355-8/00/$10.0002000 IEEE.

[4] Su Myat Marlar Soe and May Paing Paing Zaw, 2008, Design and Implementation of Rule-based Expert System for Fault Management, World Academy of Science, Engineering and Technology 48 2008.

[5] http://www.pcmag.com/encyclopedia_term/0,2542,t=hardware+failure&i=44106,00.asp

[6] http://www.sigops.org/sosp/sosp09/papers/kadav-sosp09.pdf

[7] Yingying Wang, Li Li, Ming Chang, Hongwei Chen, Xiaoming Dong, Yueou Ren, Qiuju Lia,Dan Liu, 2009, Fault diagnosis expert system based on integration of fault-tree and neural network, 978-1-4244-4507-3/09/$25.00 ©2009 IEEE.

[8] Gennady Krivoulya, Zoya Dudar, Dariya Kucherenko,Mehana Sami, 2009, Fuzzy Expert System for Diagnosis of Computer Failures, CADSM'2009, 24-28 February, 2009, Polyana-Svalyava (Zakarpattya), UKRAINE.

[9] Guo-Zhong Zhou, 1993, A Neural Network Approach To Fault Diagnosis For Power System, IEEE TENCON'93 / Bei.Pg.

[10] http://en.wikipedia.org/wiki/Expert_system

Abdullah Saad AlMalaise Al Ghamdi is the chairman of Information System, Faculty of computing and Information systems department, King Abdul Aziz university, Jeddah, KSA. He has published many research papers in International and National journals.

Syed Mutiullah Hussaini- Senior Lecturer & Research Scholar in Computer Science & Information Systems, College of Computer Science & Information Systems, Umm Al-Qura University, Makkah, Kingdom of Saudi Arabia.

Abdul Khadar Jilani is presently working as a faculty in Majmaah University ,Al Majmaah(Formerly under King Saud University). Before joining KSU , he worked with various universities in Sultanate of Oman. He has published papers in International journals and conference proceedings.