THE AUTOMATED ARMY ROTC QUESTIONNAIRE (ARQ)

David L. H. Young

U.S. ARMY
HQ TRADOC BLDG 5 #G103
ATTN: ATRM-K
FT. MONROE VA 23561

Abstract. The Reserve Officer Training Corps Cadet Command (ROTCCC) takes applications for its officer training program from college students and Army enlisted personnel worldwide. Each applicant is required to complete a set of application forms prior to acceptance into the ROTC program . These forms are covered by several regulations that govern the eligibility of potential applicants and guide the applicant through the filling out forms . These forms and regulations are maintained by personnel at Army Education Centers, college ROTC departments and career placement offices. Because these individuals are not normally involved in ROTC admissions completion process, they are not thoroughly familiar with the filling out of forms or the eligibility criteria established by the regulations. Eligibility criteria changes as Army regulations are periodically revised. The distribution of revised regulations is a costly venture for ROTCCC. It results in missing or outdated regulations being maintained at the colleges and education centers. Outdated information results in a loss of applications attributable to frustration and error.

ROTCCC asked the Artificial Intelligence Center at Fort Monroe, for an inexpensive and reliable way of automating their application process. After reviewing the process, the Center determined that an expert system with good end-user interface capabilities could be used to solve a large part of the problem. The system captures the knowledge contained within the regulations, enables the quick distribution and implementation of eligibility criteria changes, and distributes the expertise of the admissions personnel at Cadet Command to the education centers and colleges worldwide.

The expert system uses a modified version of CLIPS that was streamlined to make the most efficient use of its capabilities. A user interface with windowing capabilities provides the applicant with simple and effective way to input his/her personal data.

DESIGN METHODOLOGY

The process of encoding the ROTC application form questions into CLIPS code was a straight-forward task. The true complexity occurred in trying to streamline CLIPS, the ROTC CLIPS code, and designing a compatible windowing interface. The design is based on an iteration process handling data based on the concept of a LIFO queue. The data is collected and analyzed with the appropriate results being displayed and the data on a potential cadet is saved.

The design steps are :

(a) stage one : understanding ROTC policy with the help of a domain expert;
(b) stage two : what was ROTC looking for in an automated program;
(c) stage three : designing prototype knowledge bases that demonstrate stage one and two are mastered;
(d) stage four : embedding the ROTC policy for GREEN TO GOLD, High School, and College Students into CLIPS code and testing with help of a domain expert;
(e) stage five : designing prototype C functions to handle problems that CLIPS cannot;
(f) stage six : using C functions to store collected information into a data file;
(g) stage seven : streamline C prototype functions and design a generic end-user interface for CLIPS;
(h) stage eight : modify ROTC CLIPS code to take advantage of CLIPS hieratical lookup table, the improved C functions, and the end-user interface commands;

The ROTC process is a vast and complex system of paper work. The ARQ was created to reduce this process. The ARQ does not cover all possibilities at this time. As the ARQ expands, it becomes more and more thorough.

This program is not meant to replace the current ROTC system already in place but to aid and reduce paperwork. None of the benefits listed by this program are a guarantee, as ROTC reserves the right of all final decisions.

ARQ SPECIFICATIONS

The ARQ was required to do the following :

(a) operate in a 640 k ram environment on a 286 or 386 IBM / Compatible machine.
(b) use a language easy to learn, update, and maintain a program.
(c) be efficient and thorough.
(d) save the information collected on a potential candidate.
(e) for the program to fit on one or two disk.

The design and structuring was the author 's.

DESIGN STRUCTURE OF ARQ

The design of the AUTOMATED ARMY ROTC QUESTIONNAIRE (ARQ) takes advantage of what CLIPS has to offer. Each feature of CLIPS was used where it was most beneficial to the program or saved memory. All C code written for the ARQ does functions that CLIPS does not provide.

The Rules are structured to assert new facts or carry out specific commands. Ninety percent of all printout statements have been removed from the rules. This allowed for smaller rules and more rules could be added and processed.

With a few exceptions, all statements, questions, and instructions are kept in a text file accessed by the hieratical lookup table. Exceptions are :

(a) statements found in the code.
(b) one line statements printed in color to attract the user's attention.

Information collected on a user is stored as facts. Select facts shall be used to "fire" other rules. Each program iteration stores the user's information to a DBASE file for future reference by ARMY ROTC.

Program templates provide quick and direct checking or modifying a stored information. One template retains the scores for select data comparison. The programmer can conveniently modify the scores list when test values change without changing the program or a particular rule. Some lists are not templates, and are stored as facts to save on memory. They are only traversed for particular value.

The Facts architectural design is modeled after the OOD (OBJECT ORIENTED DESIGN) principle to allow facts to integrate with future object oriented versions of CLIPS. Some facts do not follow this principle. They consist of one or two straight-forward words and control switching between processing phases.

The end-user interface is a popup windowing package written in Microsoft C 5.1 (a TURBO C compatible version is under development.) The windows for ARQ are basic and limited. (More advance window functions are under development.) The popup windows and menus displaying information to the screen allow for a friendly interface between the user and the computer. Window functions are called from CLIPS and do not consume much memory. Due to the time limit and all the possible combinations of facts, rules, and window functions, the actual number of windows open at one time is currently unknown. The program has ninety-two rules, the facts vary in number, and a maximum of four windows are opened at once.

There are other C functions for interfacing and are needed to run the ARQ.

The ARQ screens three different ROTC candidate profiles:

(a) The GREEN TO GOLD candidate (active duty enlisted soldier.)
(b) The On-Campus College candidate.
(c) The college bound High School candidate.

There are general questions which apply to all the prospects which are followed by three different sets of questions which were tailored to address the unique characteristics of a given profile. There is a rule for each profile which asserts a group of facts spontaneously. There is a question protocol. The facts channel relevant and related rules into proper firing sequence.

ARQ operates in a modified CLIPS 4.3. ARQ can be upgraded to run with CLIPS 5.0 assuming no memory limitations occur. All C code will have to be modified. The CLIPS Ver. 4.3 is the version of choice. Version 5.0 in a 640k ram environment requires further testing.

THE ARQ OPERATION

The ARQ asks the user questions by using popup windows and menus to obtain information. The questions asked, pertain only to the users. The number of questions vary depending on the user's responses. Each response is used for firing new rules or is stored as a fact. As the user answers the questions, informative statements will appear in a popup window. When the user finishes the questionnaire, input (data) is gathered into one template. The template is then written as a record in a DBASE III file. ARQ then prompts the user to run again.

If for any reason the program is exited before completion all data collected is destroyed upon exiting. No information will be saved to the DBASE III file.

BENEFITS OF USING THE ARQ

The ARQ allows users to quickly determine eligibility for the ROTC program. This makes the process less confusing. It provides the ARMY with a reliable upto date record of all individuals who are eligible and interested in ROTC. ARMY ROTC can follow up and refine the leads provided by ARQ.

The simplicity of ARQ means that anybody can use it without instructions.

ARQ serves as a tutorial to teach, inform, and update ROTC education and guidance counselors.

Common "What if" scenarios now have solutions. Estimating or second guessing is not necessary.

FUTURE GOALS

The next version should contain additional profiles thus making ARQ more thorough. The rules should be divided into single or like candidate files and a main rule file used for running the ARQ. The main file would give a menu selection of all the candidate types, select, and load in the corresponding candidate rule file and run.

A more advance windowing package built for CLIPS with graphics pictures should tempt more users to try the ARQ. The graphics should enhance the program and make the ARQ more palatable. Advanced window functions will make CLIPS more user friendly.

The ability to provide multiple drive access is needed for writing output to disk storage or for files located in different drives and directories.

The incorporation of the ROTC LOGO would give the ARQ a more authentic feel to the user.

Long term goals are to send and maintain the ARQ electronically world wide and to upload and down-load the ARQ on a main-frame.

The long term goals will involve a substantial investment in technical and

manpower resources.

SUMMARY

ARQ is a versatile and portable tool. It has the flexibility to keep expanding and still be extremely efficient and reliable.

The CLIPS language is easy to learn and makes it easy to update the ARQ.

Since the ARQ reduces paperwork, the ROTC application process can execute faster and smoother.