

A Concept of Agent-based Learning Support System for C Programming

Kazuhiko Nagao and Naohiro Ishii¹

*Information Science and Technology Department,
Yuge National College of Maritime Technology*

*Department of Electrical and Electronic Engineering, Nagoya Institute of Technology¹
nagao@info.yuge.ac.jp, ishii@egg.ics.nitech.ac.jp*

Abstract

The programming is one of the most important factors in the education of computer literacy. However, in the initial learning of the programming, many students often have faced a simple problem, like syntax-error by careless or miss typing. Therefore, they feel that the programming is too hard. The most important problem is the student can't take advantage of the error messages with displayed from the computer. These messages are difficult to understand for beginners because the message only indicates a syntactic error and where it occurred.

We propose the support system for the C language programming on the UNIX by using agent technology. This system monitors the error message from the compiler. It analyzes, rewrites the acquired message, and reports to the student and the teacher on the network. The teacher can improve the teaching way by finding category of error. The student customizes these messages, and they can use the suitable one at each level. Moreover, these messages send to other students by communication line of each agent. Consequently, many students are able to use these useful messages; they can select the most preferable message from the collection of the agent. We executed the preliminary experiment; the improvement of efficiency to the complication work was able to be confirmed.

1. Introduction

The study of the programming is very useful for the understanding of the computer science and training a logical idea. In our college, we are teaching about the programming by the C language on the UNIX. We require that student thinking about how to solve the problem and designing the algorithm and creating the program by C language. Students have to understand the syntax of this language, and they should be able to operate the several tools, such as a editor, compiler, debugger, and online manual.

Many students often have faced a simple problem, like syntax-error by careless or miss typing. Therefore, they

feel that the programming is too hard. The most important problem is the student can't make effective use of the error messages from the compiler. These messages are difficult to understand for beginners because the message only indicates a syntactic error and where it occurred.

To correspond to these problems, we arranged several Teaching Assistant (TA) in charge of the question from students [1]. TA holds the skill level of student and tutors the appropriate guidance. However, the work of TA is too large, so there is no spare to follow the each student.

On the other hand, the student who had some experience gives the hint for the problem solving to other student. Thus, the most preferable method of programming is a corroborative learning [2].

Generally, it is natural result that there is a difference of the skill between students. The skill of the tool and library of the language influences the progress of the programming. Especially, it is necessary to the experience for the debugging of the program by using of error messages. Unfortunately, error messages are written in English, so students of our college feel like difficult.

On the Windows OS, there are several environments for programming (IDE) based on Graphical User Interface [6-8]. These IDE provide the more easy method for the programming, like editor, online manual, and debugger. However, our programming environment on the UNIX OS is based on Character User Interface, and tools are individual. Major IDE does not exist yet.

To cope with these problems, we provide a new environment for C programming on the UNIX system. It aims a improvement of the environment for compiling and debugging. This system monitors the error message from the compiler. It analyzes, rewrites the acquired message, and reports to the student and the teacher on the network.

If this system encounter the new error message, it query to other system for other student. It supports for corroborative learning.

2. Problems on the programming

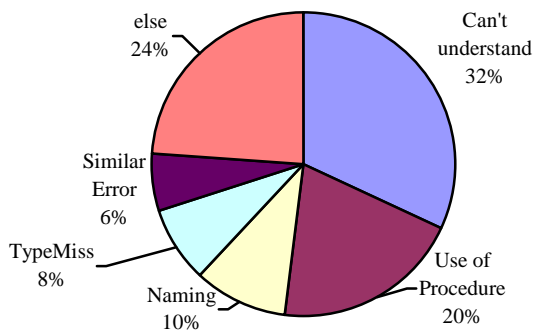


Figure.1 Problem which student feels on the programming

Skill of the programming only improves by training and experience. But, we feel that the exercises dose not registered so much. So, we research the questionnaire for the students of our college (the 3rd, 4th and 5th student of Information Science and Technology Dept.) The result are indicate as follows(shown in Fig.1):

- Difficult to understand about the error message
The Compiler inspects the whether the program is correct or not and reports the error messages. But some students could not understand about what is it meaning, so they could not use this information for debugging. They feel that it is hard to understand English, so the error message too.
- How to use procedures
Number of the standard library procedure that is used in the first class of programming is about 10-20, but student often mistake them. Student can use the online manual “man” at any time, but they do not use it. Student feels that the statement of man is too difficult, and they wants to indicate the example of these functions.
- Naming rule
It is necessary that name of variables or functions should be significant. But, if student has poor vocabulary, then it is difficult to naming to even if the student knows the importance of the name.
- Too much miss typing and miss-operating
The type mistake on the program causes many error messages from the compiler. Student feels the difficulty of the programming by amount of these messages. It should be work out with experiences on the debugging, but completion of the name of function is also useful.
- A similar error is often repeated
Student could not understand the error message what does it mean. Student should read these messages more carefully, and find the cause of errors.

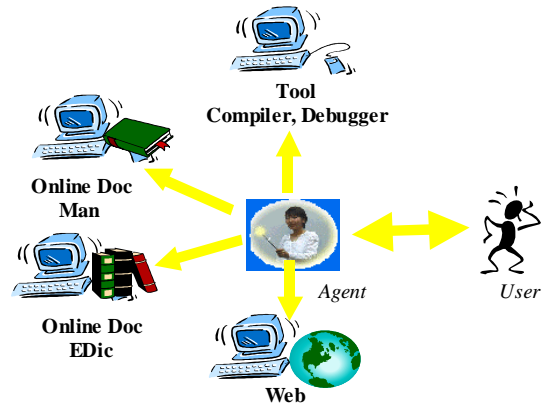


Figure.2. Concept of support system for C programming

From these analyses, student could not use the error message from the compiler. A lot of errors make the student thinks the programming is difficult and it decrease the desire of programming. English messages also complicate the problem. They encounters the problem at the before of the design of the algorithm.

3. Necessity of the IDE on the UNIX

The environment of UNIX has several tools, such as a compiler, a debugger and online documents, so UNIX is effective to the programming education. The C language was developed for programming on the UNIX system, the syntax is a simple, and developed code is very effective, so C language can be used on many operating systems. However, according to the GUI environment like the Windows system, UNIX style became old past.

A basic concept of the UNIX provides many small good quality tools. These tools are connected by IO redirect and pipe, so the UNIX is a good environment as well as Windows. If these tools can be combined, the usability of each tool is rapidly improved.

Several IDE systems with GUI on the UNIX, which is like on the Windows, are developed [4,5,6,9]. Students can develop the program without considering the compilation. The other related work, Takamoto provides the support system for programming by monitoring the compilation errors [3].

4. Support system for C programming by agent

4.1 Function of this system

In this research, we propose the agent-based support system for C programming. The agent indicates the computer program, which acts independently and supports the user. The agent collects the related information from

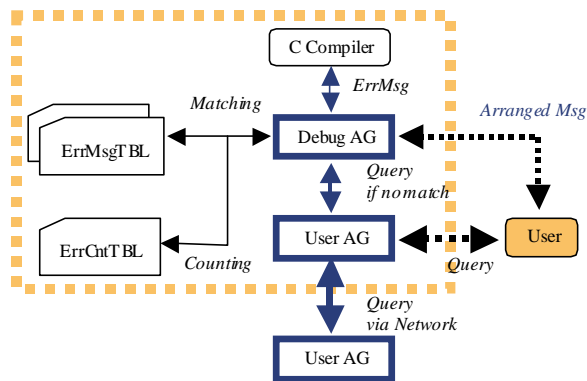


Figure.3 Processes of debug agent

the compiler, Man and WWW. It monitors the user's operations. If some errors are occurred, the agent puts out an appropriate suggestion. Each agent is cooperated by inquiring in the network. Agent connects several tools to the one integrated environment. Agent acts as a "Paste" between these tools.

The function of this agent are :

- Integrate the editor, compiler, debugger and online manual.
Most works are executed on the editor, so every tool are executable from the editor.
- Searching the document
It automatically searches the keyword from the Man and WWW documents.
- Monitoring the compiler
It monitors the message from the compiler, and creates the summary of these errors. It records the incidence of the errors for each student.
- Learning the unknown errors
If a unknown error is encountered, it reports to the student and wants to more information.
- Networked agent
On the learning, important things are noted by each student, so each notebook becomes different. The agent also grows up as well as this according to the use of each student.

4.2 Implementation of the system

Our proposed system is still now developing. This system are consisted from 3 modules:

- Edit agent module
It monitors a operation of editing, mistyping, completion the function name.
- Debug agent module
It monitors the message from compiler, and rewrites to Japanese and summarizes.
- User agent module
It provides the user interface. It manages the

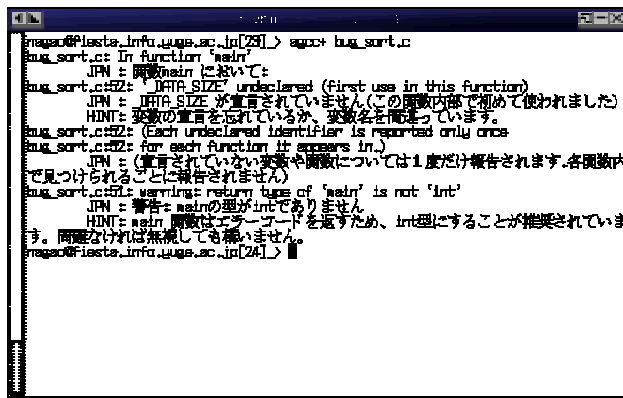


Figure.4 Arranged messages from the debug agent

history of operation and errors. It inquires to the other agents.

5. Implementation

We are now developing the system, which we proposed. Development of debug agent module is almost finished, and student can use it's function. It is easy to change the configuration of the agent for each student by hand. Prototype of debug agent was implemented by Perl. Several tools, such as GCC, Emacs, and man, which we often use were built in.

5.1 Debug agent

Debug agent monitors the output from the compiler. It makes the statistics of the errors and indicates the converted messages (to the Japanese, or summarized). The agent has the error statistics table and the message transfer table for each student individually. The message transfer table was created by past study [1], so some messages are not registered.

Debug agent calls the C compiler inside, and acquires the error messages. Each message are matched on the message transfer table, and transferred to the Japanese message or Japanese summary. These messages can be control by program options.

If there is no correspond message, the debug agent requests the collection of this information to the user agent. The user agent collects the some information from the users or other agents, and replies it.

The occurrences of these errors are recorded to the error statistics table and report to the teacher's agent.

6. Evaluations

- To evaluate the effectiveness of this system, we tested this system on the practical programming exercise. Student are separated into 3 groups:

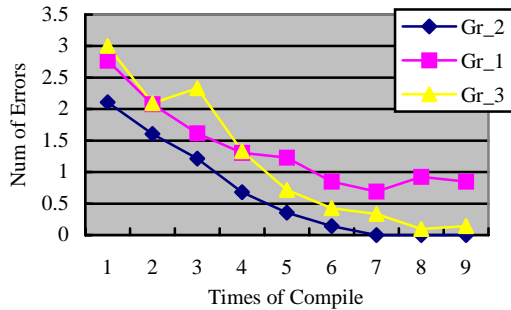


Figure.5 Evaluation of the efficiency of messages

- Gr-1(agcc-) uses the original compiler
- Gr-2(agcc) uses the compiler with agent which rewrite to Japanese message
- Gr-3(agcc+) uses the compiler with agent which rewrite to Japanese message and some hints

All students should complete five programs. All the errors that occur while developing are recorded. Fig.6.1 shows the number of compilation and errors. This indicates that the number of the errors of Gr-2 decreases earlier than other groups and completes programs. It was confirmed that the agent who rewrite to Japanese has an enough effect in the programming.

7. Conclusion

In this paper, we provide the programming environment on the UNIX system. Classical tools on the UNIX integrate into this environment by using of software agent. This system monitors the user's operation, error messages from the compiler, and collects new information from others. This system grows up separately according to each student.

By using of this system on the practical exercise, this system is useful for support the programming.

It is necessary to compare with the other environments based on GCC. The analysis algorithm of the error message is based on syntactic errors from the compiler, so it is not able to guess the exact causes of errors. It is very difficult problem. In the questionnaire about programming, student wants to indicate the similar example of programming. We have to provide a extra documentation.

8. References

[1] Nagao, Takagi, "A.B. Smith, C.D. Jones, and E.F. Roberts, "Effect of teaching assistant in the programming exercise in the college", Proc. of the computer education of college, No17,pp.127 1997.

[2] Takagi, "Effective method for programming education", Proc. of the computer education of college, No.18, pp.67, 1998

[3] Takamoto, Fujii, Izumi, Tanaka, "A Support Based on a Bug Factor for Programming and its Effectiveness", Transactions of JSISE, Vol.17, No.4, pp.533 2000

[4] "gIDE", <http://gide.pn.org/>

[5] "GNU Emacs", <http://www.xemacs.org/>

[6] Borland Inc., "J Builder", <http://www.borland.co.jp>

[7] Borland Inc., "C++Builder", <http://www.borland.co.jp>

[8] Borland Inc., "Delphi", <http://www.borland.co.jp>

[9] Borland Inc., "kylix", <http://www.borland.co.jp/kylix/>

Appendix.1 Questionnaire about Error Message

1. Did you use the error message?

Yes	8
Probably Yes	11
Probably No	6
No	2
2. Why did (didn't) you use the error message?

understand error message	5
understand position of error	12
can't understand	5
can't understand English	4
3. Is the message from agent useful?

Yes	6
Probably Yes	8
Probably No	2
No	0
4. Do you think the message from agent is necessary?

Yes	12
Probably Yes	4
Probably No	0
No	0