Primal Strategies to Enhance Performance of Computer Science Students

Zarana M. Kanani[1] Krishna C. Pate[2]
[1,2]Teaching Assistant
[1,2]Uka Trsadia University

Abstract— Programming is an essential step for those who are beginners in computer science field. Now a day it becomes necessary to provide effective learning environment to students for improving their logical skills. In this paper we have proposed some strategies which can help to undergraduate students to build their programming competences and also to create basic programs.

Key words: Information Technology, Academic Performance, Computer Science, Programming Skill

I. INTRODUCTION

To learn computer programming, student must have some prerequisites knowledge such as solving mathematical equations, interest in field of programming and problem solving ideas[3][7].

It is fact that numbers of students are jobless because they were practically very weak in field of computing and IT. To survive in IT field for long period of time; it becomes compulsory to be updated and capture knowledge about new trends in programming. Thus, to improve the ability to learn programming languages of novice students; effective strategies are needed.

During the study, we found Digital Computing Game, Programming Tool, Algorithm, Debugging Tool and Error Detection Tool that support students to boost their logical thinking ability[1][4][6]. Apart from it, continuous internal evaluation (Quizzes, Open Book Test, Unit Test, Practical Exams and Self Creation Parameter) also play a major role to enhance result in programming. There are various approaches applied by many researchers to improve programming and logical skills in students. In this paper we mainly focuses on how to improve students' academic performance in the field of computer science using average practical marks analysis of first year students which is describe in methodology part of the paper. Also some recommendations are suggested in this paper.

A. Objective of the study

The main objective of the study is to recommend some strategies which can help to improve programming skill as well as academic performance of under graduate students of computer science.

II. LITERATURE REVIEW

In Paper [2], author focuses on study of intervention strategies which can be used in teaching Data Structure course in IT. They have analysed student's academic performances based on semester wise students' grade point. Analysis includes grade points distribution of students in Data Structure and averages of all grade points are studied. Author have recommended following intervention strategies- Tutoring, Student Support Program, Li fe Skill Program, Warm-up activities using small group and emphasize real-life application. As a result they have concluded that intervention strategies are excellent to improve academic performance of students in Data Structure course.

In paper [1], authors mainly focus on impact of Digital Competition Game to improve the abilities on sequencing, defined iteration and nesting in computer science students. The paper also described an aspect of interest that learners are most likely to become intellectually engaged when they are working on task is most interesting and relevant to them. In their research work they developed ProBot game with the goal of to boost ability to development of basic algorithm thinking in students. The game was designed with important point such as attractive interface, error detection, difficulty levels, scoring and validation. At conclude with the result that digital game in educational purposes helping students to reinforce analogous skills, progressive levels, error findings and build alternatives algorithms and its logic for the specific problem.

In paper [6] describe the teaching methodology for analysis of student's mistakes in both compiler and logical errors. The research method includes compiler error detection that produce "Syntax Errors" and "Lexical Errors", a pattern in compiler error by collecting all the compiler errors while debugging, a characteristics of good debugger vs. weak debugger on basis of their performance.
At last it discovered that good debugging skill is most important to be a good programmer by reading and understand other people's code to enhance competence of novice programmer.

In paper [8] authors, focuses on various ways or methodology that helps to students in learning introductory programming courses. Authors suggest some strategies like build graphical models, programming assignments, and developing machine learning techniques and applied it in assignment and how students learn to program by an investigation of debugging in their own code.

They developed machine learning techniques and applied to instrument IDE that provides a progress of student's performances graphically. A research work include log analysis of student`s program, bug, future performance prediction, data analysis by student`s growth through an assignment which represent distance metric evaluation. Research work also include some data mining part for early identification of student`s difficulty in computing. A research work conclude with analysing student development path on an assignment play vital role over simplify analyse their final grade.

In[3] authors focuses on the approach to the development of algorithmic thinking of beginners within the subject algorithm and data structure and graph theory and combinational thinking in students, author had apply puzzle and logical games, algorithm structures and dealing with graph algorithm in their teaching activities. In subject algorithm and data structure they have used Czech meta-language for writing algorithm for developing logical thinking in subject Graph Theory and Combinatorial

Optimization. They have introduce two puzzle are Riddle and Crossword puzzle dealing with isomorphism graph theory and two logical games namely Sprouts which can be solved by two or three students and HYCLE is designed for one player. In subject Graph Theory and Combinatorial Optimization they have used one tool namely GrAlg (Graph Algorithm) used by students as well as teachers.

In [7], the main objective of this paper is to identify the behaviours of students in computer programming course. To identify the academic performance of student, some recommended improvement strategies were specified. Semester wise grade distribution for each students were used to determine the progress of students in programming course and result analysis is done based on the distribution. Also some recommendations are given for improvement in teaching and technical skills.

III. METHODOLOGY

For analysis of student's performance in programming course (Bachelor in Computer Application), the study was conducted using average of practical marks obtained by students for first and second semester in the subject Fundamentals of Programming and Object Oriented Programming respectively.

Average marks of all CIE(Continuous Internal Evaluation) parameters i.e Unit test-1, Unit test-2, Internal Examination and external Examination marks is used for study as follows.

CIE Parameters	Average Marks
Unit Test-1(20 Marks)	10
Unit Test-2(20 Marks)	7
Internal(30 Marks)	14
External(30 Marks)	17
Over All(50 Marks)	27

Table 1: Average marks of first semester students in the subject fundamentals of programming.

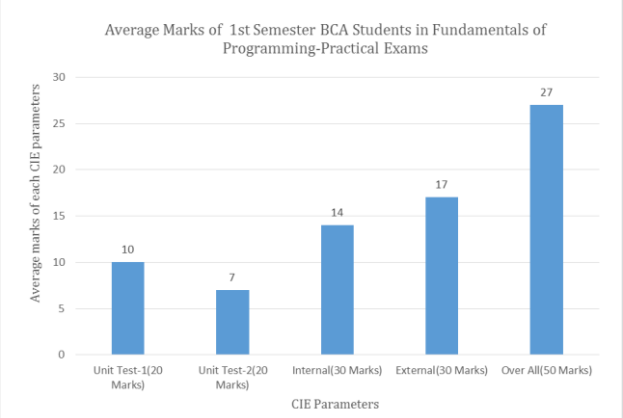

Fig. 1: Average Marks of 1st Semester BCA Students in Fundamentals of Programming-Practical Exams

CIE Parameters	Average Marks
Unit Test-1(20 Marks)	10
Unit Test-2(20 Marks)	12
Internal(30 Marks)	20
External(30 Marks)	18
Over All(50 Marks)	32

Table 2: Average marks of second semester students in the subject Object Oriented Programming

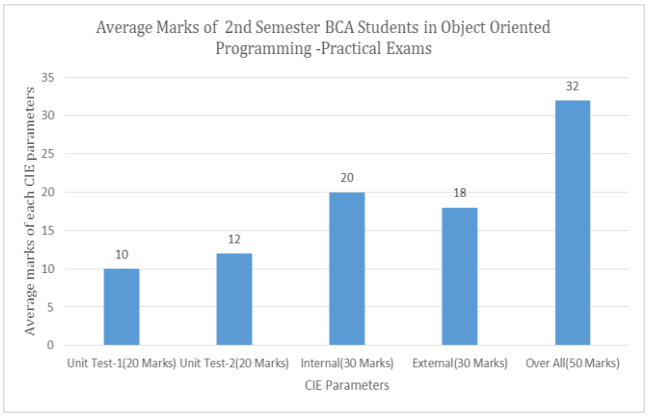

Fig. 2: Average Marks of 2ndSemester BCA Students in Object Oriented Programming-Practical Exams

CIE Parameters	1st Semester Average Marks	2nd Semester Average Marks
Unit Test-1(20 Marks)	10	10
Unit Test-2(20 Marks)	7	12
Internal(30 Marks)	14	20
External(30 Marks)	17	18
Over All(50 Marks)	27	32

Table 3: Comparison of average marks of first and second semester students.

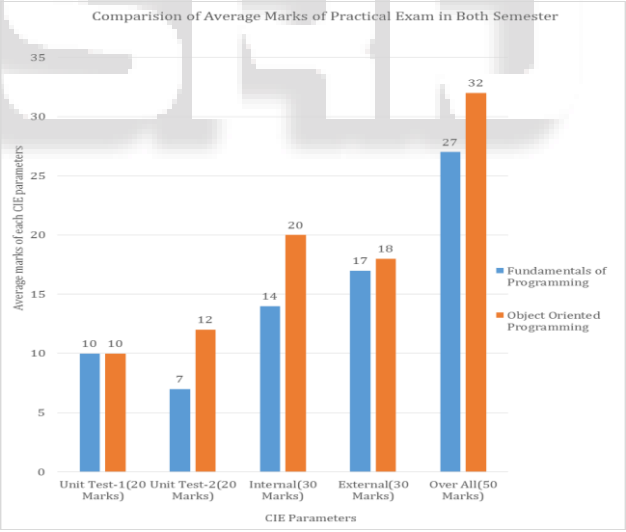

Fig. 3: Comparison of average marks of first and second semester students.

A. *Analysis of Results*

During the first semester and second semester (figure-3) the results showed that there is no change in the performance of the students in unit test-1, after that in unit test-2 students performance has increased by 5(25%), in internal exam also performance has increased by 6(20%), where as in external exam student's performance has increased by 1(3%). Furthermore, overall results showed that there is slight improvement of 5(10%) compared to first semester overall result.

From above analysis of result it is clearly noticed that there is a need of proper attention in programming subjects of first year computer science students. For that appropriate teaching strategies should be adopt by lecturers to improve student's academic performance specially in practical sections [7]. For planning teaching strategies, some recommendations are given in below section.

IV. RECOMMENDATIONS

A. Lesson Plans

Pre-planning should be made by the lecturers about lesson plan and their methodology to explain particular topics to students in class room. This lesson plan should include course objective, course outcomes, units, sub-units, number of lectures to complete the unit, references per unit, teaching methodology and evaluation parameters.

B. Group Study

Working in small groups helps especially to low grading students to improve their academic performance by discussing with each other about their problems. They can ask questions to other group mates and get solutions from them.

C. An Error Detection and Correction

Many students have a good knowledge of programming but don't have the skill to debug program intelligently and this is major issue in productivity of their working program code. Debugging and error solving skills are stepping up confidence of programmer. For that first of all collect all the students' compiler/logical error messages while program exercise during the term then find most common error among all students. In second phase discuss and explain previously found common errors with students so that misunderstanding of a concept can be removed. Furthermore, third phase involves an examination may conduct, in that a question will design with some errors including both compiler and logical errors intentionally. The given errors will similar to those most commonly raised by students, which will help students to find out the reason, mistakes and solution for those errors efficiently.

D. Assignments and Practical List

Practical problems along with error and output exercises for each topic should be given to students for their practice in programming as well as for concept clearance assignment for each unit should be given and discuss in class about it. Class exercises helps students a lot to understand and grab the concepts clearly.

E. Logic Building Exercise

Puzzle solving exercise, reasoning question solving exercise, mathematical equations solving and logical games can helps students to improve there thinking ability.

V. CONCLUSION

In this study, the researches explored some recommendations to enhance ability of students in programming field. Apart from their regular lectures, additional activities like logic cultivation process, good evaluation skill for their own designed code, lesson plan and right evaluation scheme, discussion, pseudo code designing exercises and practical list plays a great role at initial stage of computing field career. The impact of above listed parameters are few strategies that helps in logic building but also helps to boost students' confidence and make the right attitude and provides the motivation for innovations in Information Technology and Computer Science domain.

REFERENCES

[1] J. Moreno, "Digital Competition Game to Improve Programming Skills," Educational Technology & Society, vol. 15, no. 3, pp. 288-297, 2011.
[2] R. A. G. a. L. A. Al-Safadi, "Intervention Strategies for the Improvement of Students' Academic Performance in Data Structure Course," International Journal of Information and Education Technology, vol. 4, no. 5, 2014.
[3] H. EVA MILKOVÁ, "Algorithmic and Logical Thinking Development: base of programming," WSEAS TRANSACTIONS on COMPUTERS.
[4] M. Cristiana Areias, "A tool to help students to develop programming skills," International Conference on Computer Systems and Technologies - CompSysTech'07, 2007.
[5] Y. G. Filiz KALELIOĞLU, "The Effects of Teaching Programming via Scratch on Problem Solving Skills: A Discussion from Learners' Perspective," Informatics in Education, vol. 13, no. 1, pp. 33-50, 2014.
[6] D. E. a. C. H. Marzieh Ahmadzadeh, "An Analysis of Patterns of Debugging Among Novice Computer Science Students".
[7] O. G. O. O. A. Y. Akinola Kayode Emmanuel, "Improvement Strategies for Computer Science Students' Academic Performance in Programming," American Journal of Computer Science and Information Engineering, vol. Vol. 2, no. No. 5, pp. 45-50, 2015.
[8] M. S. D. K. S. C. P. B. Chris Piech, "Modeling How Students Learn to Program".
[9] M. Rinard, "Using Programming Language Concepts to Teach General Thinking Skills".

