Strategies for Improving Academic Performance of Information Technology Department Students' in Computer Programming Skills : The Case of Wachemo University

Wubetu Barud Demilie

Department of Information Technology, Wachemo University, Hossana, Ethiopia

ABSTRACT

Computer programming skill is an essential skill for information technology and for some other natural science department students. However, teaching and learning computer programming ideas and skills has been recognized as a great challenge to both teachers and students. Therefore, the development of efficient learning strategies and environments for computer programming courses have become an important concept. This study is intended to recommend strategies for improving academic performance which can be used in teaching computer programming course effectively to undergraduate students in information technology related departments or informatics departments (including information Technology, computer science, and software engineering and information systems This concluded that there is a need for strategies for improving academic performance which requires planning and implementation to strengthen the motivation of the students to study, boost their academic performance, and uplift the quality of education in information technology departments. Finally, relevant and important workable recommendations have been suggested by the researcher.

Keywords : Academic Performance; Computer Programming; Informatics; Information Technology; Academic Performance

I. INTRODUCTION

As we know, the students' academic performance is the outcome of several graded points such as the final examinations, quizzes, assignments, attendance and other graded points related to the course. The primary mission of each higher institution in department of information technology is to offer high quality and relevant education in order to produce skillful and competent graduates. To accomplish this, a number of practical and instructional strategies were designed to improve the students' academic performance. Computer programming courses are the basic and/are the requirements to obtain a degree in all information technology departments. It is generally belief that computer programming courses are the most difficult courses in higher institutions for information technology department students. Because it is too abstract and requires higher analytical and problem solving skills. Thus, to improve the performance of struggling students in computer programming courses, improvement strategies are needed. Apart from the fact that, teaming number of graduates are unemployable and jobless because they were ill equipped in practical and employable skills in their courses.

Thus, there is a need to assess and suggest strategies for improving academic performance for information technology department students' in computer programming courses [1]. The purpose of determining strategies for improving academic performance in

higher education is to identify all factors that may advance students' performance academically. Lecturers have usually found ways for classroom strategies that will reach out and help the students to boost their performance in all computer programming courses [2][3].

1. Statement of the Problem

Learning computer programming is a complex task since it requires new idea in thinking and creative skills in problem solving. The average of students' academic performance in computer programming courses for over the years in undergraduate program of information technology departments in many universities can be concluded as poor compared to other courses in the field. However, the percentage of students with poor academic performance every semester is unquestionable and do not support the mission of many academic programs. This affects the quality of education the academic programs are trying to keep up. Improving the academic performance of students is part of the continuous effort to raise the standard of education in the area of information technology.

2. Objectives of the Study
2.1. General Objective

The general objective of the study is to identify strategies for improving academic performance of information technology department students' in computer programming skill.

2.2. Specific Objectives

To achieve the general objective, the following points are listed as specific objective for the study.

- ✓ To identify the behaviors of students in computer programming course
- ✓ To identify the academic performance of students in the courses

- ✓ To recommended improvement strategies to advance the students' performance in information technology departments.

3. What Does Mean Computer Programming?

A computer programming language is a systematic notation by which students describe computational processes to others [4][5]. Computational process in present context means a set of steps or algorithms that a machine can perform for solving different kinds of problem accordingly. However, computers, unfortunately, do what we tell them to do so, not necessarily what we want them to do! There must be no ambiguity in the instructions that we give to a computer in our computer programmers, i.e., no possibility of alternative interpretations. Since the computer always takes some course of action, great care must be taken to ensure that there is only one possible course of action so that we get the desired results. Computer programming can be hard. However, we can do a great deal to make it easier. First, it is important that we separate the problem solving phase of the task from implementation phase [6][7][7].
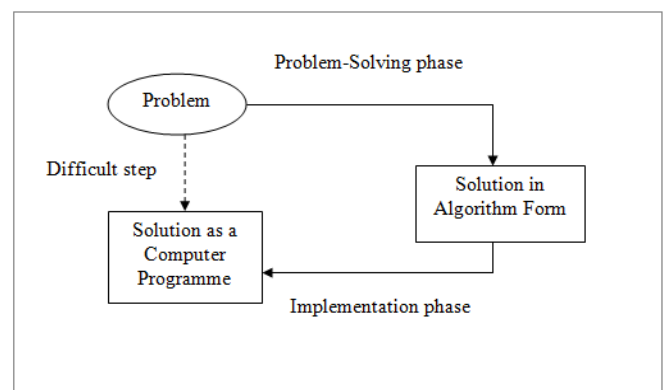

Figure 1 : Problem solving steps with computer programming

4. Why Computer Programming?

Information technology department students need to keep in mind the following basic points if they want to be a successful computer programmer[2] [6][8][9][10].

4.1. Focus on developing your practical computer programming skills

What do you learn during information technology departments' degrees? Most of the courses are not all theory compared to courses from other types of degrees, like Management, accounting, business administration etc. Even if information technology department degrees include theoretical classes, their approach is far more practical. For instance, right after a theoretical class, you will have to apply that knowledge through practical exercises in a computer laboratory. Sure, you may have to work on essays, but they will always have to include practical work or projects both successfully combined, with logical explanations for each step and chapter from your work.

4.2. Studying the night before a deadline will not work

Try to get your assignments done well in advance of deadlines, ideally have them ready a week in advance. Otherwise, you'll just get tired, your code won't work, and you'll fail the assignment. Information technology department is one of those areas where it is absolutely necessary to keep working throughout the semester because sometimes you'll need to take a break to get some clarity as to why something isn't working. So do yourself a favor and work all the way through the semester. That doesn't mean you should spend every minute studying; it means you'll have to be constant with your learning progress throughout the year. You'll do far better this way.

4.3. Don't expect to learn everything during courses

All information technology department courses are rock. However, they won't cover all the skills you might need in a job. For example, if you're planning to go into software development, you might learn a considerable amount of computer programming languages, but you may need to go beyond that for work. You may have to understand and get the hang of new computer programming languages on your own, without anyone to teach you. It might be a good idea to look for introductions to online computer programming languages, some of which are even offered for free.

4.4. Don't be a lone wolf

Being able to work on your own and independently motivate yourself is great, and you'll need that. But don't forget that you're not alone, and its' highly unlikely you'll be alone in the workplace as well. You're likely to be part of a team. And team work takes some skills you'll need to brush up on and work constantly. Once you'll find a good study group, you'll surely enjoy to meet with your peers, share ideas and work on interesting projects. After all, that's just the classical way innovation starts: with a few genius minds that come together and work on a new invention until it becomes trademark.

4.5. Remember that you still have life

It may seem pretty daunting as you'll need to learn more, study more, and do more. But don't forget to live your life. Have fun, meet up with friends, go clubbing or travelling, get involved in some student activities.

5. Significance of the Study

The study aims to assess the academic performance of students in all computer programming courses.

The beneficiaries of the study are:

- ✓ Students, to boost their academic performance in computer programming skills and to help them recognize the value of learning each course of their chosen field.

- ✓ Lecturers, to assist them in the area that their students needed, helping them to make the course appealing to their students.
- ✓ Curriculum development committee, to review the course as a whole, focus on the current computer programming software, hardware and area that is needed by the students, recommend necessary actions to assist the department in maintaining the quality of education.
- ✓ Information technology academic departments and institutions, to help them maintain the standard of the quality of education that they are providing to the community.
- ✓ Future researchers, to study on how to assist students' of information technology in improving their academic performance.

6. Methodology

The research was conducted in an academic institute which offers degree program in information technology department. Descriptive and qualitative method was employed by identifying the characteristics of the students in all computer programming skills. The grade points for each student per semester were utilized to determine the progress of students in computer programming course. In addition to students' grade points, informal interviews with some selected faculty members were conducted to gather additional information that is useful for this study. Since most computer programming courses are given for information technology department students during second and above semesters (mainly from second year firsts semester and above for information technology department students). Due to this, I have used data of computer programming courses from second and above batch students. For experimental purpose, I have used information technology department students of one batch only (from year 2017-2019) for year 2nd – 4th with all computer programming courses of the curriculum for three consecutive years.

Table 1: Semester breakdown of computer programming courses

No	Course Name	Year	Semester
1.	Fundamentals of programming II	2nd	1st
2.	Data Structure and Algorithms	2nd	2nd
3.	Fundamentals of Internet programing		
4.	Object Oriented Programming		
5.	Event Driven Programming	3rd	1st
6.	Advanced Programming		
7.	Advanced Internet Programming		
8.	Information Storage and Retrieval	3rd	2nd
9.	Introduction to Distributed Systems		
10	Integrative Programming and Technologies	4th	1st
11	Computer Graphics		
12	Artificial Intelligence		
13	System Simulation and Modeling (the remaining are networking and system administration related courses)	4th	2nd

7. Presentation of Data and Discussion

Grading Scale and Letter Grade System of Wachemo University students

Table 2 : Grading scale and letter grade system of Wachemo University students

Raw Mark Interval [100 %]	Corresponding fixed Number grades	Corresponding Letter grade	Status Description	Class Description
[90, 100]	4.0	A+	Excellent	First class with Great Distinction
[85, 90)	4.0	A		
[80, 85)	3.75	A-		
[75, 80)	3.5	B+	Very Good	First Class with Distinction
[70, 75)	3.0	B		
[65, 70)	2.75	B-	Good	First Class
[60, 65)	2.5	C+		Second Class
[50, 60)	2.0	C	Satisfactory	
[45, 50)	1.75	C-	Unsatisfactory	Lower Class
[40, 45)	1.0	D	Very Poor	
[30, 40)	0	Fx	Fail	
[0, 30)	0	F		

(Source: Harmonized Wachemo University Senate Legislation [11])

The following sections discuss the results of the students' academic performance in computer programming courses for first and second semester (year two and above).
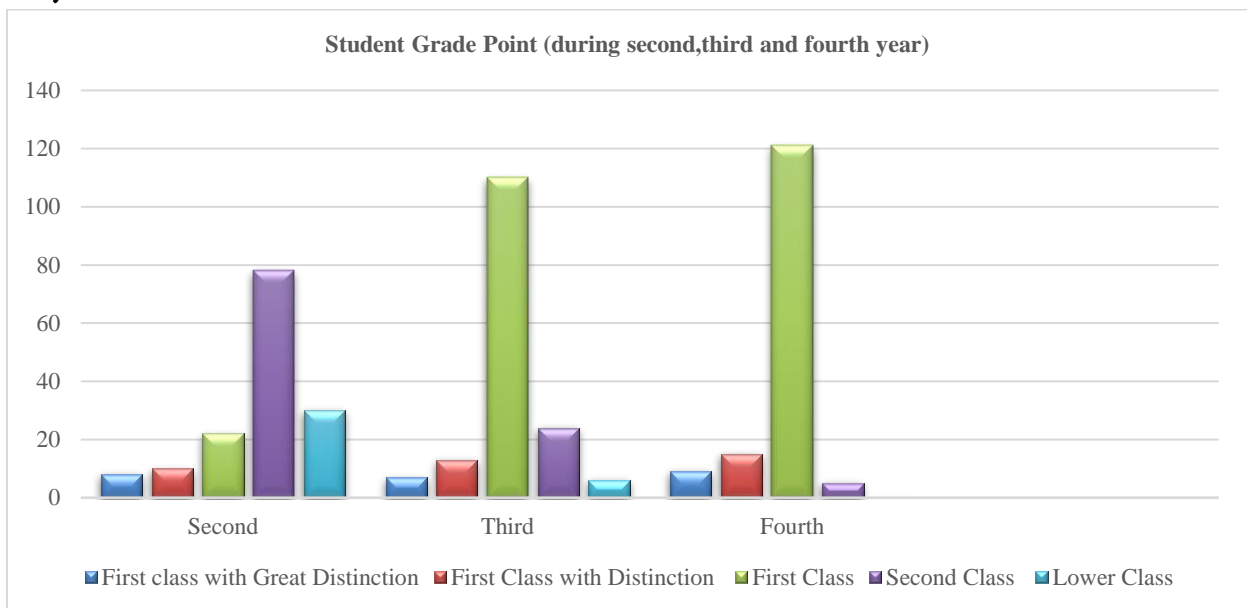
8. Analysis of the Results

Figure 2 : Percentage distribution of the student grade for the second semester

During second year students revealed that 8(35%) class description of "first class with great destniction",10(32%) class description of "first class with distinction", 22(14%) class description of "first class", 78(33%) class description of "second class" and 30(33%) class description of "first class".

During third year students revealed that 7(35%) class description of "first class with great destniction",13(32%) class description of "first class with distinction", 110(14%) class description of "first class", 24(33%) class description of "second class" and 6(33%) class description of "first class".

During fourth year students revealed that 9(35%) class description of "first class with great destniction",15(32%) class description of "first class with distinction", 121(14%) class description of "first class", 5(33%) class description of "second class" and 0(33%) class description of "first class".

The above analysis clearly indicates poor academic performance of the student as only few of them can attain the level of upper credit and distinction. In addition, the students' academic performance has increased from semester to semester and from year to year due to several reasons. As the result indicates, students' academic performance can be improved by adopting and answering what computer programming and by consecutive practice of computer programming problems with appropriate problem solving techniques and skills. Finally, students' practical activity from semester to semester and from year to year will increase the performance and it leads information technology department students to be successful in computer programming skills [4][12].

9. Conclusion and Recommendation

9.1. Conclusion

It cannot be fairly conclusive that the background of student pursuing computer programming course is a major influence on performance although there is a significant correlation since majority of students who enroll in information technology programs in which computer programming courses are part it. Learning strategies that students adapt to the study of computer programming course are of greater importance with regards to their performance. In most cases, learning approach which requires technical/ practical and collaboration as well as persuading the learner to think outside the box and solve problem is much effective. It is therefore, worth stating that implementation of multiple strategies to learn computer programming courses will enhance high performance.

9.2. Recommendation

1. Informatics department courses should be taught by exemplary lecturers/instructors who have the requisite
   knowledge to teach the curriculum and who continue to upgrade their technical/practical and teaching skills throughout their careers because of rapid changes and advancement in the world of technology.

2. There is need to implement a national information technology curriculum for high schools (Senior Secondary School and Junior Secondary School). This curriculum must be principle based, must address core content and key skills, and must incorporate appropriate strategies to reach and teach our students.

3. There is need to support the implementation of this new curriculum with a plan that includes a realistic timeframe and the provision of the resources required to achieve it.

4. There should be program to mentor, counsel, advise, guide, and tutor student. There should be academic
   support through extra lessons, remedial lessons which should be facilitated or supervised by a qualified   and
   dedicated person who supports the department's vision. Also, in order for the tutor to be effective,

the tutoring style should match with the students' learning style. Auditory learning style work best, when they are able to hear it. While kinesthetic learning style work best, when they are doing it or physically involved with it.

5. Instructor's Capacity Building. Instructors' should be given opportunity to develop and improve their teaching skills by attending seminars, conferences etc. and should be allowed to teach other instructors as well.

6. Using Small Groups. Having students work in small groups or in student pairs is an effective instructional
strategy especially for struggling students. In small groups, struggling students are more likely to ask questions and benefit from explanations from fellow students.

This issue requires a national vision, supportive action, and commitment at all levels of the political and educational systems as schools alone cannot achieve these outcomes.

REFERENCES

[1]. H. Xiaohui and B. B. Zhou, "Improving teaching in Computer Programming by adopting student-centred learning strategies," no. November, 2006.

[2]. W. B. Demilie, "Causes of Failure of University Students in Computer Programming Courses : The Case of Wachemo University," vol. 5, no. 5, pp. 123–132, 2019.

[3]. H. Shaari and N. Ahmed, "Improving Performance and Progression of Novice Programmers : Factors Considerations," vol. 8, no. 1, pp. 7–10, 2018.

[4]. P. Y. Owusu, J. K. Arthur, and C. Adjetey, "Performance of Students in Computer Programming : Background , Field of Study and Learning Approach Paradigm," vol. 77, no. 12, pp. 17–21, 2013.

[5]. M. N. Ismail, N. A. Ngah, and I. N. Umar, "INSTRUCTIONAL STRATEGY IN THE TEACHING OF COMPUTER PROGRAMMING : A NEED ASSESSMENT ANALYSES," vol. 9, no. 2, pp. 125–131, 2010.

[6]. D. Nandigam and H. Bathula, "Competing Dichotomies in Teaching Computer Programming to Beginner-Students," Am. J. Educ. Res., vol. 1, no. 8, pp. 307–312, 2013.

[7]. A. K. Emmanuel, O. G. Oluwadamilare, and O. A. Yomi, "Improvement Strategies for Computer Science Students ' Academic Performance in Programming Skill," vol. 2, no. 5, pp. 45–50, 2015.

[8]. S. Wiedenbeck and D. Labelle, "Factors Affecting Course Outcomes in Introductory Programming," no. April, pp. 97–110, 2004.

[9]. S. Mohorovičić, "An Overview of Computer Programming Teaching Methods."

[10]. I. Miliszewska and G. Tan, "Befriending Computer Programming : A Proposed Approach to Teaching Introductory Programming Difficulties Encountered by First Year Programming Students Impact of Failure / Poor Performance on Students," vol. 4, 2007.

[11]. "Harmonized WCU Senate Legislation." .

[12]. S. E. E. Profile, "Academic Achievement in Computer Programming Instruction and Effects of the Use of Visualization Tools ; at the Elementary School Level Academic Achievement in Computer Programming Instruction and Effects of the Use of Visualization Tools ; at the Elementary School Level," no. November, 2015.