A Survey of Undergraduates of Computer Science Performance In Programming and Non-Programming Courses

O.E. Oduntan (Ph.D) & O. J. Adegboye
Department of Computer Science
The Federal Polytechnic
Ilaro, Ogun State, Nigeria
E-mail: estherbest2000@yahoo.com, adegboyeoj@gmail.com

ABSTRACT

Development of a functional software have been a challenge in most developing countries, software industry have been importing application from on the shelf medium, which does not treat domestic problem. This study delves into investigating the cause of the inability of computer science graduates to fully deliver as a software developer in the nation. The causes were traced back into checking the performances of the students while undergoing their course of study. A descriptive research approach was employed in this study. Dependent and independent variables extracted from students results in programming and non-programming courses. The Dependent variable comprises of the Scores while the independent variable comprises of the nature of the course: programming or non-programming; Statistical tools such as mean, mode, regression analysis and Pearson correlation coefficients were adopted and analyzed using SPSS. Results show that students perform better in non-programming courses than programming courses. It was discovered that the ratio of programming courses to non-programming courses for computer science is 1:3. Finally, the correlation of the average score of students is 0.21475, which is a low positive correlation. It is recommended that computer science curriculum should entail more of programming courses than non-programming courses.

Keyword: Performance, programming, non-programming, scores, computer science

1. INTRODUCTION

Programming is an art and it requires the individual's ability to interpret challenges into solutions. Computer Science students are required to take several programming courses as structured in their four or five years program either in Polytechnic or University as the case may be. In their early years of studies they are required to study programming. The art of programming includes knowledge of programming tools and languages, problem-solving skills, and effective strategies for program design and implementation (*Shaun, 2014*). Computer programming is part and parcel of the computer science and its related programs in education. It is an essential skill that must be mastered by anyone who seeks to study computer science. Normally, computer science students will first be introduced to the concept of programming and data structure where they are taught on how to analyze problems, use specific techniques to represent the problem solution and validate the solution. Computer programming courses are a part of many tertiary institutions' curriculum and among the most important subjects for computer science students.

Statement of Problem

Students' inability to work as software developers after graduation is one of the major concerns that lead to this research. Many students graduating from schools where they have learnt various programming courses for 2 or 4years including a year of industrial attachment are unable to work as professional programmers. Instead, they drift completely to other fields that bear little or no relation to programming. This does not promote the course: Computer Science. This study focuses on investigating some of the factors responsible for the deviation of Computer science graduates from programming related job.

2. LITERATURE REVIEW:

Many researchers conducted detailed studies about the varying performance of students at different levels of study (Shoukat, Zubair, Fahad, Hamid, Awais, 2013). According to Minnesota (2007) "the higher education performance is depending upon the academic performance of graduate students. Durden and Ellis quoted Staffolani and Bratti, (2002) observed that "the measurement of students previous educational outcomes are the most important indicators of students future achievement, this implies that the performance of graduate student is a function of their performance while they are undergoing their course of study.

Detailed studies have been conducted in the area of students achievement in terms of programming and identified and analyzed the academic performance of the student at the tertiary level of education. Their findings identify that students who study computer science are very poor in terms of software development due to poor performance in different programming courses while in school. The usefulness of these findings lies in the need to undertake corrective measures that improve the performance of students in their programming based course, in other to see that they have learnt necessary programming skills needed to become a software developer. It is generally assumed that the students who had better or higher performance in programming in the starting classes of their studies, will progressively master and acquire some programming skill for future programming endeavors.

Since the last few years it has been noticed significantly that there is great addition in research literature and review material relating to students' achievement with much emphasis on the measures of programming performance as a determinant of future programming skill in software industry or programming innovative measures.

3. RESEARCH METHODOLOGY

This is a descriptive research that involves carrying out a random sampling of undergraduates of computer science course. Dependent and independent variables were used. The dependent variable is the Score (%) of the student, while the independent is the nature of the course (that is : programming and non-programming courses). Scores of students were extracted in programming and non-programming courses. The case study for the research is the 2016/2017 Academic Session, 1st and 2nd semester results of Higher National Diploma Students of the Department of Computer Science, The Federal Polytechnic, Ilaro, Ogun State. The sample size of data analysed is forty-two (42) in number, comprising of four (4) non-programming courses and four (4) programming courses. The analysis involved comparing Mean, Pearson Correlation on data collected, and Regression Graph was plotted. Tables and charts were used for the presentation of results. Data collected were compiled, sorted, edited, classified and coded into the coding sheet of SPSS 23.0 (version).

4. RESULTS AND DISCUSSION

Table 1: shows a total of forty–two (42) average scores of students' performance in programming and non-programming courses. In Table II, the **frequencies** of the students' score range was determined using **SPSS**, the mean for non-programming based courses came out the most; which is **70** whereas programming based courses is **40**.

Figures I and Figure II; present the analysis in table I and II using a bar chart for quick comprehension of the study.

Table I: Average Score of Students Performance

Programming	43 51 58 63 40 85 46 79 50 42 70 41 50 55 42 69 42 56 21 54 41 69 40 69 40 54 40 30 61 74 70 72 54 74 60 83 58 68 65 75 45 71
Non-Programming	69 76 86 82 52 81 62 86 69 73 80 70 70 70 56 64 58 63 46 72 60 60 50 66 64 59 64 71 52 70 56 63 40 70 54 70 54 79 48 43 62 40

Table II: Descriptive analysis of Students Performance in Programming Based and non-Programming based Courses using Frequencies in SPSS

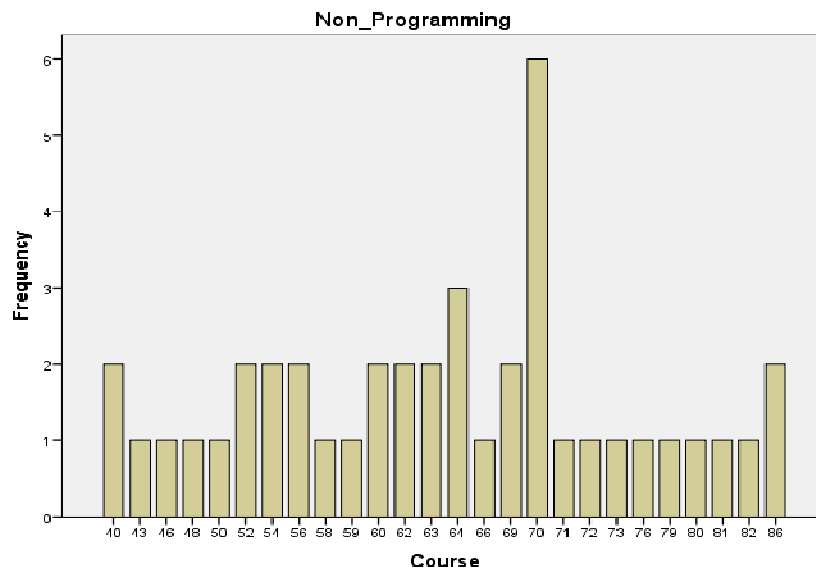| Course Category	Programming	Non_Programming
Sample Size	42	42
	0	0
Mode	40	70

Figure 1: A chart of an Average Score of Students in Non-Programming Courses
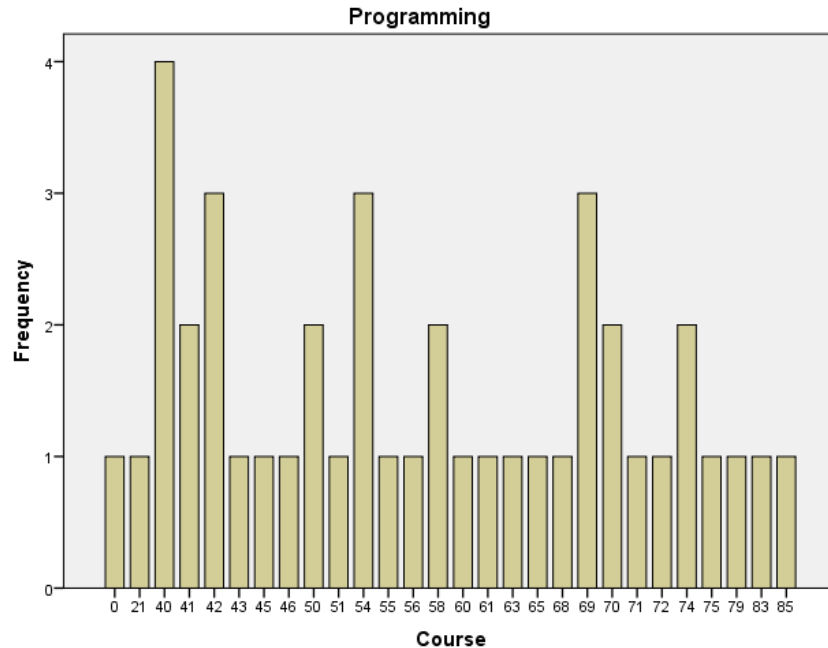
Figure 2: A chart of an Average Score of Students in Programming Courses

Pearson Correlation Result:

Pearson Product Moment Correlation (r) signifies the degree of relationship that exists between dependent variables and independent variables. With Pearson correlation coefficient, the valid result for r lies between -1 and +1. If the result lies between 0 and 1, it shows there is a positive correlation that is X increases as Y increases. If r = 1, it shows that the result is perfect positive. If r is between 0.5 and 1, it shows a high positive correlation, when r is between 0 and 0.49, it exhibits a low positive correlation. When r = -1, it shows a perfect negative correlation that is the rate at which the dependent variable increases is exactly equal to the rate at which the independent variable decreases. When r is between -0.5 and 0, it shows a weak negative correlation, when r is between -0.49 and -1, it exhibits a strong negative correlation. The formula for deriving the Pearson Correlation Coefficient as stated by Islam and Hogue (2012) is:

$$r = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{(\sum X^2 - \frac{(\sum X)^2}{N})(\sum Y^2 - \frac{(\sum Y)^2}{N})}} \qquad \text{eqn (1)}$$

In this study, the results of the investigation of students' performance in programming and non-programming courses in computer science, were presented making use of the Pearson correlation coefficient to determine if there is a relationship between the student performances in both courses. The dependent variable is the Score of the student while the independent variable is the nature of the course: which are: Programming and Non programming.

Result shows that (r) is 0.21475, which indicates that there is a low positive correlation between the performances in programming and non -programming courses.

Regression Analysis Result

Figure 3: is a **Regression Graph;** it could be observed that the students' performance in their non-programming courses is still higher (graduated to about **70** on the graph) than their performance in the programming courses (graduated to just about 56 on the graph), the difference glaring.

All the various data analysis approach used are unequivocal in their outcomes, we can then conclude that students show better performance when it comes to courses that are not programming based than their performances in programming based courses.
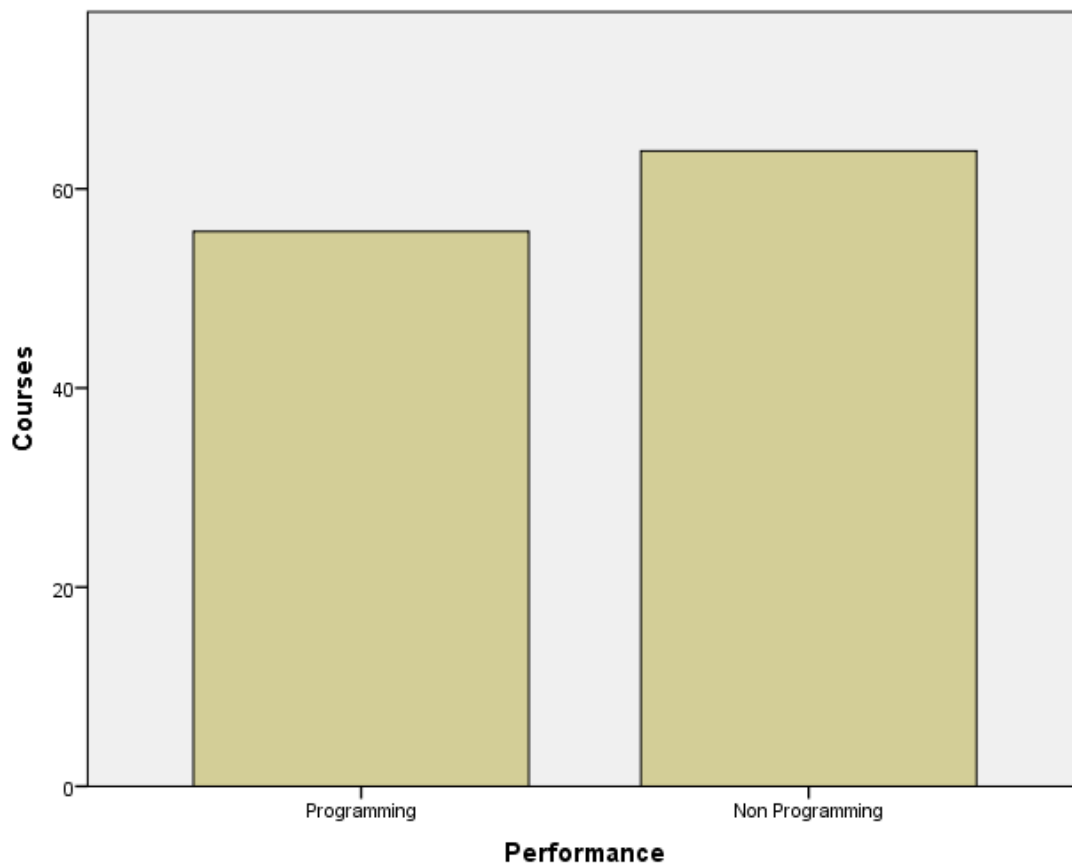

Figure 3: Regression Variable Plots –Graph of Both Programing and non-Programming based courses

5. CONCLUSION

This study endeavored to investigate into the number of programming courses and non-programming courses in computer science curriculum for Higher National Diploma students in Nigerian Polytechnics, it was discovered that the ratio of non-programming to programming courses is 1:3; even in institutions where local content have been added. Also, it has been discovered that undergraduates of computer science are more proficient in non-programming courses than the programming courses; which can be detriment to their career as software developers.

REFERENCES

1. Bratti, M. and Staffolani, S. (2002). 'Student Time Allocation and Educational Production Functions', University of Ancona Department of Economics Working Paper No. 170.
2. Durden, G. C. and Ellis, L. V. (1995). The effect of attendance on student learning in Principles of Economics. American Economic Review, 85, 343-346.
3. Islam, M.M., Hogue, A.S.M.L., (2012): "Automated Essay Scoring Using Generalized Latent Semantic Analysis" Journal of Computers vol 7 no 3 pp.616-626.
4. Minnesota Measures (2007) Report on higher education performance. Retrieved on May 24, 2018 from www.opencongress.org/bill/110.s/642/show-139k.
5. Shaun Bebbington (2014). "What is programming". Retrieved 2nd June, 2018
6. Shoukat et. al., (2013) Factors Contributing to the Students Academic Performance: A Case Study of Islamia University Sub-Campus American Journal of Educational Research. **2013**, 1(8), 283-289. DOI: 10.12691/education-1-8-3