

Efficiency of Engaging Python in Teaching Discrete Mathematics: Case Study

Lamya A. Omer Alani

Department of Computer, College of Science, University of Duhok.

E-mail: lamyaomer@yahoo.com.

Abstract

This study describes how the subject of discrete mathematics can be integrated with programming techniques for the purpose of teaching. Achieving such integration which can be understood easily by the students reminds that this subject is usually taught to the first years of the academic study. Python is one of such programming language which can be easily understood and practiced compared to other language such as C++, java etc. for instance using Python does not require such effort in designing the structure of program. By providing students with a python language they could use to understand both discrete mathematics and programming effectively and efficiently in the most appropriate manner, this reduce difficulties of imagining the logical way of writing programs also give a brilliant way to rise programming skills and give a simple way to study mathematics practically not only theoretically. This paper represent a case study for teaching discrete mathematics using Python programming language to second year students at computer science department - faculty of science, University of Duhok were given practical lectures for inserting python with their theoretical discrete mathematic study.

Keywords: *Programming, Computer Science Concepts, Learning, Math Code, Implementation.*

Introduction

Discrete mathematics as other subjects in mathematic science taught theoretically not practically. Discrete mathematics topics include logic and proofs techniques; number theory; set theory; functions and relations on sets; graph theory; algorithms, their analysis, correctness; matrices; sequences est. Computer science students need a basic knowledge of many of these topics [1].

With the experience of more than 20 years as a lecturer teaching both mathematics and computer programming, it has been noticed that students were facing problems in understanding both of these two subjects. It sounds complicated to characterize of a mathematical problem and solve it, sometimes for some students, they didn't know even where and how to start.

It should be mentioned that the lectures are not responsible for this drawback, in fact the impose system of the university curriculums still adopt the classical methods of study thirty or more years ago, the use of computer in many branches of science was not so insensitive as these years, nowadays computers enters many fields as effective tools to improve different branch of science even

though those whom far a way of computer so its defiantly important to reconsider the techniques used in teaching these subjects, for instance discrete mathematics. The concepts of computer programming are logical and mathematical in nature.

The specialization of computer science has many fields which require a strong foundation of mathematics; a good computer science specialist must have good mathematical background knowledge. Learning how to program is generally considered hard, It has even been said, that it takes about 10 years for a beginner to become an expert programmer [2][3]. Students need to know how to solve problems before they can computerize it, by helping the students developing their problem solving skills, then learn how to use these skills to solve problems using computers. Usually, the math lecturer, starts with simple examples to clarify a new subject, and gradually complicate the examples to cover all sides of the subject, this attract the student and start to interact with the subject, however when they start facing the complicated questions they fail to solve them, a similar state happen when teaching a programming language, however there is an additional

problem with teaching programming language since knowing language concepts are not enough the important part is building the logical side for any student thinking, that will convert the student to be a good programmer based on the fact that the student will have this by repetitive and memorize methods and techniques of language. Although There are close links between mathematics and computer programming, in fact the computer science was a part of mathematics and afterwards gained independence as a sole discipline [4].

During these days students accustomed to the use of computers and deal with modern applications sometimes as if they were professional programmer, even though most of them face difficulty in writing simple program, so how can they become familiar in writing software programs especially main programs with complicated mathematical equations which are required in modern research, also the student's mathematical knowledge are always theoretical and the student never try to see its practical applications.

This paper depicts some examples of discrete mathematics problems while teaching computer science student of BS by using python software.

Teaching Discrete Mathematics in Computer Sciences

Computer science is one of the modern trends of sciences which rapidly evolving. The evolution of this scientific branch is accompanied by a great diversity in the related subjects. The mathematics of modern computer science is built almost entirely on discrete math. The students who study computer science will need a solid background in mathematic to get ready to understand the fundamental concept of computer science subjects. Then they will be able to consider the issues of the real world without complexity. For instance, the lecturer can get use of sets and relations concepts in the design of databases in a simple manner. Also the concepts of propositional logic assist the students to understand the installation of computer's hardware and software. Thus, lecturers can apply other mathematic concepts for the rest of relevant principles and applications of Computer science.

The core of Computer Science is to transform real world problems into computational ones and solve the computational problems [5]. Considering the style used in the teaching of mathematics in high school, we find that the student suffers from resolved smoothly and the reason for that is relying on one or more of three beliefs, "The book says that, my teacher says that, everybody knows that".

With old versions of compilers and machine user interface the fresh student of computer science department were facing difficulties in memorizing the instructions of a given programming language. In addition, they were focusing on typing these instructions without having any syntax errors. Recently with the development of smart devices, nowadays students have curiosity to look for more modern applications that, satisfying the same traditional objectives.

In order to manage the above options, a lecturer may encourage his/her students to implement their requirements using a simple programming language such as Python instead of using more complex languages. Students should learn that solving math problems is similar to write a programming function in many aspects. The Iraqi high schools typically focus on segmentation techniques in learning how to solve highly stereotyped problems in math classes. This is the way that professional mathematicians solve real problems that can arise from the everyday life, or from science [5]. This clearly approved during the process of writing programs by undergraduate students who suffered from unpleasant experience using high level programming languages. Therefore this article suggests to solve problems by combining both math and programming concepts by using Python.

What is Python?

Discrete mathematic defined as foundational material for computer science that is required by many other areas in the syllabus [5]

Python is an example of a high-level language; Python is a modern, object-oriented programming language [6].

It combines remarkable power with very clean, simple, and compact syntax. Python is

easy to learn and very well suited for an introduction to computer programming [3].

Python is a popular general purpose programming language which is well suited to a wide range of problems [5].

Recent developments have extended Python's range of applicability to econometrics, statistics and general numerical analysis [7].

Although there are a large number of Western studies on the use of Python as a useful language in improving the understanding of discrete math and enhance the performance of students techniques in programming, still the studies in Iraq are few, or we can say do not have such studies. In this paper we describe python labs which merge between these two subjects. As known teaching them is one of the difficult tasks faced by both teachers and students.

Methodology

Discrete Mathematics is one of the most thematically rich lessons and it has a good advantage of not using the complicated mathematics and good understanding for its concept lead to good construction of different basis of all aspects of Computer Science. A sample of 52 students in second year at computer science department at Duhok University were given "with their theoretical discrete math class at computer lab" an overview of Python inaugurated and explain matters athletic style demanding them to write programs in a Python, trying to simplify the concepts as much as possible.

In this course the students were given a series of examples to develop their ability in both python and mathematics incrementally. Here are some of the examples that were given to focus on the basics of the language.

Example1:

[This example is intended to introduce students to the rules of the Python as programming language (defining inputs process these inputs to get outputs also focus on the correct formula)].

Find out why the following program does not work:

```
>>>z = x + y
x = 3
y = 2
```

```
print z
```

it must describe that we define first the variables then apply the formula as written below

```
>>> z= x + y
Traceback (most recent call last):
File "<pysshell#0>", line 1, in <module>
z=x+y
Name Error: name 'a' is not defined,
While the correct syntax must be
>>> x = 3
>>> y = 5
>>> z = x + y
>>> print z
8
```

Example 2:

The second example give a comparison between using the concept of list and array in python with their different properties which are related with computer science branches

```
>>>[1,2,3,4] # to write a simple list
[1, 2, 3, 4] # The result
>>>[5,2,4,5,7,2] # reputation and unordered
elements allowed
[5, 2, 4, 5, 7, 2]
>>>['Ali','Haval','Perween']
# the elements of array can be string
['Ali', 'Haval', 'Perween']
>>> a= [1,2, 3, 4] # assign a as array
>>> names=['Ali', 'Haval', 'Perween' ]
>>> a[0] # individual element of array
1
>>> names [2]
'Perween'
>>> a.append (8)
>>> a
[1, 2, 3, 4, 8]
>>> a[4]
8
>>> a[2] =10 # changing the value of element
>>> a
[1, 2, 10, 4, 8]
```

and so on define different functions built-in such as range(function to use order a list), len() also using A mathematical operations with list.

Writing this with simple syntax

```
>>> [x*2 for x in range (1,11)]
[2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
>>> [x for x in range (2,21) if x% 2 == 0]
[2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
```

```
>>>evens=[x for x in range (2,21) if x% 2=0]
>>> evens
[2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
So we can create any mathematical expression
like adding, subtracting, multiply dividing also
find odd even prime greater than less than etc.
>>> [z for z in a z>= 8]
[8,10]
To teach the student to define his own function
such as Fibonacci series called fib. As explain
next.
```

Example 3:

Sequences represent ordered lists of elements. The sum of the terms of a sequence is called a series.

```
>>> # Fibonacci series:
>>> # the sum of two elements defines the
next
>>> a, b = 0, 1
>>> while b < 10:
print b
a, b = b, a+b

1
1
2
3
5
8
```

A trailing comma avoids the newline after the output:

```
>>> a, b = 0, 1
>>> while b < 1000:
print b,
a, b = b, a+b
1 1 2 3 5 8 13 21 34 55 89 144 233 377 610
987
```

We can create a function that writes the Fibonacci series to an arbitrary boundary:

```
>>> def fib(n): # write Fibonacci series up to n
>>> """Print a Fibonacci series up to n."""
'Print a Fibonacci series up to n.'
>>> a, b = 0, 1
>>>while a < n:
>>> print a,
>>> a, b = b, a+b
>>> # Now call the function we just defined:
>>> fib(2000)
```

```
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610
987 1597
```

(Teaching Algorithm concept and changing it to written program)

After giving a good description about algorithm and its characteristics, we try to applied the bubble sort as an example of understanding algorithm.

Example 4:

Here is the algorithm for the bubble sort:

```
Repeat
X ← Start of Array
Flag ← False
Repeat
If Number(X) > Number (X+ 1) Then
Temp ← Number(X)
Number (X) ← Number (X+ 1)
Number(X+1) ← Temp
Flag ← True
End If
X ← X+1
Until End of Array
Until Flag = False
As a written program
my_list = [12, 5, 13, 8, 9, 65]
```

```
def bubble(bad_list):
length = len(bad_list) - 1
sorted = False

while not sorted:
sorted = True
for i in range(length):
if bad_list[i] > bad_list[i+1]:
sorted = False
bad_list[i],bad_list[i+1]=bad_list[i+1],bad_list[i]
```

```
bubble(my_list)
print my_list
=====
```

Then the output will be

```
>>>
[5, 8, 9, 12, 13, 23, 31, 65]
```

Some notes about python labs

- When the instructor starts to explain how to convert mathematical problem to software program, the first step (after explain the problem idea) start with comparing how to solve this problem by python and by the language used before such as C++ by this way the programming ideas will be

repeated and this helps the student to understand and conservation.

- Discrete mathematical concepts become easier to understand such as logical statements, algorithms, GCD, LCM, Prime number, etc.

Questionnaires

Sample of 52 students were selected to complete a questionnaire after they had completed their 6 trial labs to give their opinions about python language and its concept, importance and their level of acceptances. They were asked to indicate a series of statements about Python whether by selecting answers as (true /false) or (maybe it is true or false).

Two kinds of set of questions were given to students the first set about python as a high level programming language which they were revolve around the use of Python as a programming assistance for the development of student skills in both mathematics and programming both (Appendix A). While in the second category were the questions revolve around the importance of math programming Is it possible to be actually help in the development of student skills in programming and increased grasping math either (Appendix B).

The number of students who took part in the trials was not bad and the level of interest was much higher than had been expected. Fifty-two students completed the course (from a year group of around 85). They were middle-programming capabilities or less, their abilities were generally representative. They Self-assessed programming ability was close to the theoretical average (6 compared to 10). More able students seemed to be taking part out of curiosity. The large number of relatively inexperienced programmers probably took part because of the offer of practical computerize problem in a new programming language.

Questionnaire Results

Number of student applying questionnaire rose steadily over in dealing with a python as interesting programming language (71.06%) more than the other high level language like C++, while about (17.9%) don't care dealing with python (8.4%) only refuse it. Fig.(1).

More than (65.38%) of students believe that changing the Mathematic problems to programs is very important to develop the idea of software programs in different language while 2 in ten refuse this idea and about (9.4%) agree with both choices, While most students didn't find that python improve their understanding discrete math, (61.5%) (sometime clarify some fundamental mathematic concept33.12%) Fig.(2). (54.8%) of students prefer to study each subject separately.

Students hadn't a clear idea for taking class of python in which stage (first or second stage)(67.3 %) of them with maybe (53.3%) of students agree with statement "python is an easy, clear, readable, and can simply memorized programming language" while (26%) refuse it.

Most student (74.4%) suffering from the style of writing in python (sentences arrangement, spaces est.).

Although it was easy to correct any mistake occur (67.13%).

Student had trouble with the limits of the range () function (61.09%) and has confuse with list () function.

Not all python concepts were clear (53.3%).

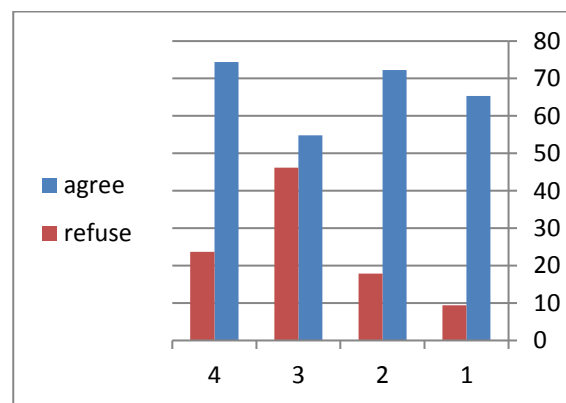


Fig.(1): Python as a high level programming language with discrete mathematic.

Python as interesting programming language find that changing the Mathematic problems to programs is very important.

Students prefer to study each subject separately. Students hadn't a clear idea for taking class of python in which stage.

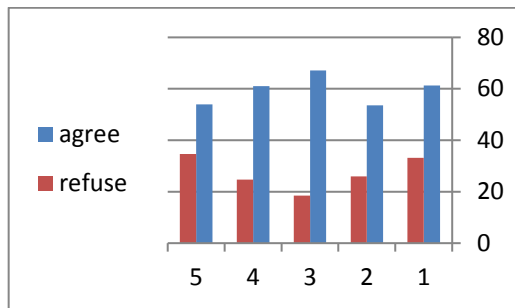


Fig.(2): Computerizing the mathematical problem.

Most students didn't find that python improve their understanding discrete math but they find that python is an easy, clear, readable, and can simply memorized programming language". Although sometimes suffering from the style of writing in python (sentences arrangement, spaces est.) it was easy to correct any mistake occur.

Student had trouble with the limits of the python concepts.

Conclusion

Considering Python as teaching language in computer science department because of its feasibility and the design and implementation of python preferable to other languages, such as C++. Students could write (and understand) basic programs quickly. by providing them with a good feedback on their progress and enhance their interest in computer programming by depending on the simple fundamental of discrete mathematic which lead the student to deal with more real world complicated problems.

it's appropriate to be teach both of these subjects at one year (for Duhok University students 2nd class) .Focus on some of the topics in Discrete Mathematics student helped in understanding programmed well.

Appendix A

The set of questions

1. Converting mathematical problems to the running programs give you a good understanding for mathematic.
2. Is python a good language to change real time problem to understand program.
3. I think that C++ is better than python as a high level language.

4. I think that study python at first stage in computer science department better than second stage.
5. I think it is better to study discrete math and python separately better than studying them Simultaneously.
6. Joining studying discrete math with programming help you in understanding both of them.
7. You think that python is an efficient and easy language to handle programmatically.
8. It is better to take classes in python in the secondary school to create a good base in programming.
9. It's not clear how to change mathematical problem to a software program.
10. Not all mathematical problem can be change to software program.

Appendix B.

1. Although Python as a high level Programming language is an easy, clear, readable, and can simply memorized programming language but it is boring.
2. You easily follow any occurred errors that you discovered during program running.
3. You can easy correct the occurred errors, unlike other programming languages like C++.
4. Object-oriented programming can be found clearly in Python.
5. You find the error message sufficiently helpful such that when they arose you could quickly see the problem.
6. You often made syntax errors like missing colons and brackets.

You understand the explanation of arrays, classes, functions ..., rest as fundamental programming concepts.

1. Python can describe any mathematical problem better than C++.
2. You can say that python is continues for programming with other high level language.
3. It's better to start understanding programming concept with python.
4. There was some confusing between the range () function and list.
5. Class, Module, dictionary are fundament programming concept became clear using python.

References

- [1] T. Vandrunen, "Discrete Mathematics and Functional Programming", p(31) June 23, 2006.
- [2] J. D. La Fuente, J. Santiago, A. Román, C. Dumitrache, and D. Casasanto, "A Primer on Scientific Programming with Python", vol (25), 2nd Edition no. (9) 2014.
- [3] G. Gopalakrishnan, B. Bolick, and T. Tuttle, "Discrete Structures via Circuits, BDD, SAT, SMT, and Functional Programming in Python", 2013.
- [4] E. Aydin, "The Use of Computers in Mathematics Education: a Paradigm Shift From 'Computer Assisted Instruction' Towards 'Student Programming'", *Turkish Online J. Educ. Technol. – TOJET*, vol (4), no. (2), p (8), 2005.
- [5] T. Whelan, S. Bergin, and J. F. Power, "Teaching Discrete Structures :A systematic review of the literature", *Hum. Factors*, pp. 275–280, 2011.
- [6] M. Williams, "Teaching scientific programming using Python", *Science p(11)*, 2002.
- [7] K. Sheppard, "Introduction to Python for Econometrics, Statistics and Data Analysis", *Report*, p (363), December 3, 2002.

الخلاصة

توضح هذه الدراسة كيف يمكن الدمج بين موضوعي الرياضيات المتقطعة وتقنيات البرمجة من خلال اثراء الخلفيه العلمية لطالب علوم الحاسبات وفهم كلتا المادتين بشكل يساعد على كسب مهارات جيدة وقابليه لتطوير التفكير العلمي والمنطقي للطالب وهذا يحد من الصعوبات التي تواجه الطالب المبتدئ في تخيل كيفية كتابة البرامج ويساعد في إعطاء وسيلة رائعة لزيادة مهاراته كما يهيئ طريقة بسيطة لدراسة الرياضيات عمليا لا نظريا حيث اعتاد الطالب دراسة الرياضيات بأسلوب نظري فقط. علما ان كلتا المادتين تدرس في السنوات الأولى من الدراسة الأكاديمية لقسم علوم الحاسبات. وقد اجريت تجربة تدريس الرياضيات الممتقطعة باستخدام لغة البرمجة بايثون وهي احدى لغات البرمجة العليا الحديثة والتي تمتاز بسهولة ايعازاتها وتعتمد على اسلوب البرمجة الشبثيه لطالب السنة الثانية في قسم علوم الحاسوب/ كلية العلوم بجامعة دهوك أعطيت المحاضرات العملية على عينة من الطلاب وخلال فصل دراسي متكامل تم فيه أدرج امثلة من مواضيع الرياضيات المتقطعة وكتابتها بلغة البايثون.