EFFECTIVENESS OF PYTHON PROGRAMMING CLASSES USING PRE-EXISTING C PROGRAMMING KNOWLEDGE

N. Fujishima[*,a] and S. Ikeda[b]

[a] Department of Electrical Engineering and Computer Science, National Institute of Technology, Matsue College, Matsue, Japan
[b] Practical Education Support Center, National Institute of Technology, Matsue College, Matsue, Japan

*E-Mail fujishima@matsue-ct.ac.jp

Abstract

An essential skill for information engineers is learning a new programming language based on ones they have already learned. In this study, a learning method is presented by practicing classes in which students learn Python by comparing it with the C language. Moreover, students are experienced this effectiveness. The students learn C language 6 months earlier. First, C code is presented to review the basics of the C. Next, the same and different points between Python and C language are explained. Then, Python code is presented. This allows students to efficiently learn new knowledge while using their pre-existing knowledge. This method enables them to use Python in a short period. We teach arithmetic, input/output, control statements (conditional branching, repeated statements), collections, and functions in eight classes. In order to evaluate the effectiveness of the method, we have confirmed the questionnaires done on the final class in the 2021 and 2022 academic year. The results are the following. First, more than 80% of the students have found the content appropriate or easy. More than 65% have felt that the learning speed was appropriate, and less than 20% have done that it was fast. More than 72% said that the comparison made it easier to understand contents. These results indicate that the transition of learning based on existing knowledge is realized in the class and that the students are able to experience the benefits of this transition. More than 88% of the students have felt that they understood the differences between C and Python, and more than 67% have done that their knowledge of C was deepened. Several statements about the perceived benefits of the comparison are identified in the feedback. The results of the survey indicate the following. First, this method allows students to experience the benefits of learning from comparison between new programming knowledge and understood one. Secondly, this allows students to learn new knowledge quickly. Finally, many students feel positively about this.

Keywords: *teaching methodology, programming, comparison, python*

Introduction

Numerous programming languages run various devices and systems in society. Each programming language has its own suitability. Hence, some programming languages are used vary widely depending on each company. In addition, new programming languages are created to meet the required applications, and old languages that have outlived their usefulness fall into disuse. Some languages have been in use for a long time. However, new features, syntax, etc. are sometimes added, so it is still necessary to relearn them. Hence, engineering students who want to work in the information field need to develop the skills to learn programming languages quickly.

An essential skill for information engineers is learning a new programming language based on ones they have already learned. One of the reasons is that understood knowledge can help them to understand the new language. the grammar of the new language will be able to be understood relatively quickly if concepts such as variables and control statements are similar between the learned language and the new one. The second reason is to clearly recognize the differences between languages. Knowing the differences between languages will make it easier to formulate ideas about which language to use when developing something. However, these are basically self-study methods. There would be few examples of classes that use one language as a base to learn a new language. We have not found no studies that show implementation in school. They are basically separated by class even if you're in a class where you learn two or more languages.

We have gotten an opportunity to teach Python with comparison. Therefore, A new learning method is presented by practicing classes in which students learn Python by comparing it with the C language. Moreover, students are experienced this effectiveness. In this study, we report our findings from our classroom practice.

Class Environment

Matsue is the capital city in Shimane Prefecture in Japan. Shimane is a tranquil prefecture far from major cities. This area is adjacent to Hiroshima Prefecture across the mountains. People in the eastern region including Matsue, tend to be generous, conservative, patient and diligent. On the other hand, people in the western region tend to be cheerful, dynamic, and emotional.

National Institute of Technology, Matsue College called “Matsue Kosen” is a school that mainly aims at growing students up a mid-level engineer. Students interested in engineering and with relatively high academic ability in Shimane Prefecture are admitted. The number of departments is five, which are mechanical engineering, electrical engineering and computer science (DenkiJoho) where the first author belongs, control engineering, information engineering (Joho) and civil environmental engineering. Each department has approximately 40 students per grade level. The school has dormitories. Hence, several students from the western region, other prefectures, or other country are in each class. One class are held in 90 minutes. There are 15 classes per course. In addition to this, one final examination may be given.

Most students in DenkiJoho department enter the program with little experience in computer programming. Some are interested in information technology, but others are not. It is not uncommon for them to have difficulty with programming. Students interested in programming tend to belong to Joho department not this department.

The classes concerned with this study are conducted in the second semester of the third year. DenkiJoho students have experienced a variety of computer-related classes to this point. They learned elementary computer use, touch typing, and Microsoft office software, brief programming using a kit in the first semester of the first year. They wrote HTML, CSS and C programming code in the second semester. This class provides students with the experience of learning by themselves through a step-by-step active learning process (Fujishima, 2019). C language is experienced in three classes. The contents are input/output, variables, conditional branches, and repetitive statements. All codes are given. There are no computer-related classes in the second year. Drawing graphics using Processing in the first semester and microcomputer programming using the C language are taught in the second semester as part of the practical training courses. Starting in the third grade they study the C language for two years. C is the most important programming language in the DenkiJoho department.

During the first semester of the third year, the students learn types, standard input/output, expressions and operators, conditional branches, iterations, and arrays. This subject is taught by a teacher who is not the authors.

Figure 1 is a schematic of the room used for the class. Classrooms are designed for students to work alone. The teacher's computer screen is projected onto a screen at the front of the classroom and onto each sub-display between the students' personal displays.

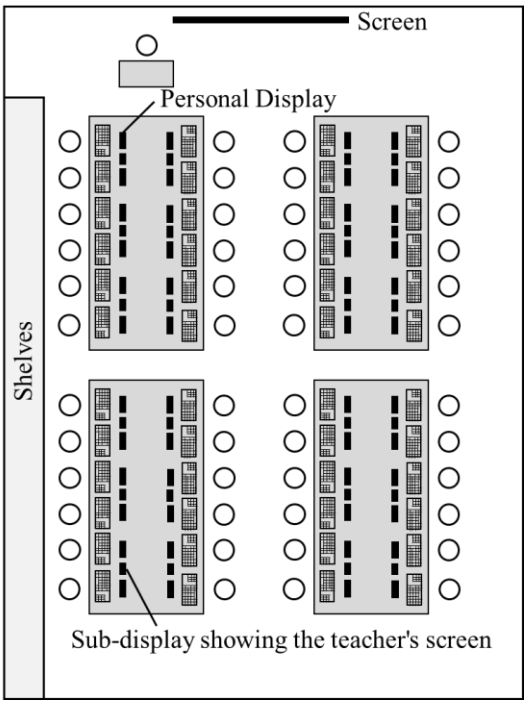

Figure 1. The structure of the teaching room

Teaching Methods

The proposed methodology implements from first to ninth. The 8th class is a mid-term examination. Therefore, the number of teaching classes adopt this study is eight. From the 10th to 15th Python-specific knowledge, scientific library usage, image processing programming and artificial intelligence programming is covered. Table 1 shows what is covered in each class. During the week of the 7th class, functions are treated in C class taught by another teacher, too. In other words, the content of this class catches up with the content of the C language in week 7.

The flow of the class is as follows. First, the teacher (the first author) provides the students with printed materials of the presentation slides. Second, the student logs into the Moodle system and opens the course page for this class. Sample code is placed inside the page. They download them to their computers. Then, the teacher proceeds with the class using presentation slides while students take time to run the sample code. After all explanation are finished, students work on their assignments. These are to rewrite C code into Python code or to write Python code that produces the desired result. They do this by asking questions of us or friends as appropriate. The students are allowed to move seats. In fact, most students do not move due to the structure of the teaching room and other factors.

The process of explanation by presentation slides is as follows. The teacher has the student run the sample code as needed. Sample code is provided through Moodle. First, the teacher explains to the students a relevant unit of the C. The symbol written "C Language" in Japanese in the upper left corner of the slide make it visually clear that this is an explanation of the C. At this time, the teacher explains them in a short time.

Table 1. Theme and Contents

Class number	Theme	Contents
1	Introduction	The term about programming, the kind of translation and the way to use Google Colaboratory.
2	Basic knowledge about the way to use variables	Variable, type, escape sequence, initialization, operator, type conversion, arithmetic, input and output.
3	control statement	Conditional branch, conditional operator and repeat statement using “while”.
4		Repeat statement using “for”, Statements changing the flow of processing (break / contnue).
5	Collection	List type, the way to use “for” statement to list variables, delete / addition, and slice.
6		Tuple/dictionary types, unpack and in/ not in operator.
7	Function (C)	Basic knowledge of functions.
8	Mid-term examination	Translation from C to Python, etc.
9	Function (Python)	The way to write functions, multi returns and default argument.

Next, the teacher teaches knowledge of python in the unit. Basically, he highlights the same and different aspects of python and C using slides with the symbol "Comparison" as Figure 2. Elements that do not exist in the C language are marked with the symbol "Python" to indicate that they are new knowledge as Figure 3.

Both sample codes are executed using web services: coding ground is used for C. Google Colaboratory is used for python. They do not require the configuration of an execution environment. The student copies the text code into the editor section and presses the execute button. The execution result is then displayed.

The Way to Research on Effects

We consider that this method is effective if the students express the following. First, the level of content should feel appropriate or easy. In addition, they perceive the learning rate to be appropriate or slow. In this class, students learn the contents that they spent more than 8 months learning in the C language class in about two months. In other words, the learning progress is very fast.

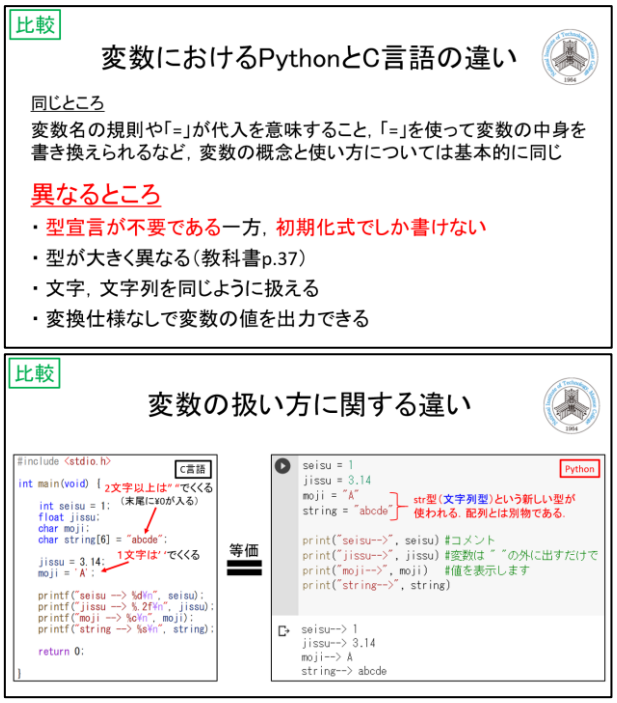

Figure 2. Slides about comparison.

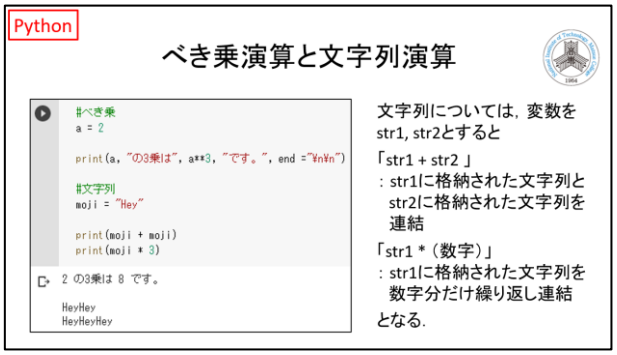

Figure 3 A slide with new knowledge about Python.

If the students think they can follow along normally or effortlessly, then the effectiveness of this approach is evident.

Second, they feel that the comparison promotes a better understanding of Python. Furthermore, as a secondary effect, we need to research whether they are experiencing a deeper knowledge of the C language. Finally, as aspects other than the effect of the class, it is important to check whether the students enjoyed the class and whether the simultaneous learning was not a burden.

Based on these, the effects have been researched using a questionnaire in the final class in 2021 and 2022 academic year. Questions are written in Japanese shown in Table 2. The teacher told the students that their responses never affect their grades and that they were to respond truthfully.

The questions related to this study are as follows. Where, questions and choices that were difficult to translate into English were converted into words with similar nuances.

Table 2. All questions written in Japanese.

・授業での Python 学習の楽しさについて，最も当てはまる選択肢をチェックしてください．(Q1 for discussion)
[1] つまらなかった [2] どちらかというと，つまらなかった [3] どちらともいえない [4] とちらかというと，楽しかった [5] 楽しかった

・C 言語で同時に学習している「文字出力～関数」までの部分について，難易度はどう感じましたか (Q2).
[1] 簡単だった [2] どちらかというと，簡単だった [3] 適切だった [4] どちらかというと，難しかった [5] 難しかった

・C 言語で同時に学習している「文字出力～関数」までの部分について，進度はどう感じましたか (Q3).
[1] 遅かった [2] どちらかというと，遅かった [3] 適切だった [4] どちらかというと，速かった [5] 速かった

・本授業で新しく習った，「クラス～深層学習」の部分について，難易度はどう感じましたか．

・本授業で新しく習った，「クラス～深層学習」の部分について，進度はどう感じましたか．

・C 言語と比較しながら Python を学んだことで，内容理解のしやすさはどうなりましたか．(Q4)
[1] 理解しにくくなった [2] どちらかというと，理解しにくくなった [3] どちらともいえない [4] どちらかというと，理解しやすくなった [5] 理解しやすくなった

・C 言語と比較しながら Python を学んだことで，C 言語と Python の違いや同じ点について理解が深まりましたか．(Q5)
[1] 理解が深まった [2] どちらかというと，理解が深まった [3] どちらともいえない [4] どちらかというと，理解が深まっていない [5] 理解が深まっていない

・C 言語と同時並行で Python を学習するのは負担が大きいですか．(Q6 for discussion, in 2022)
[1] 負担は大きい [2] どちらかというと，負担は大きい [3] どちらともいえない [4] どちらかというと，負担は少ない [5] 負担は少ない

・C 言語と比較しながら Python を学んだことで，「C 言語」の知識は深まりましたか．(Q7)
[1] 理解が深まっていない [2] どちらかというと，理解が深まっていない [3] どちらともいえない [4] どちらかというと，理解が深まった [5] 理解が深まった

・既知の言語を足掛かりに新しい言語を学んだことで感じたことについて書いてください．(Q8)

・AI プログラミングに触れたことで感じたことを書いてください．

・授業の感想を書いてください．(Q9 for discussion)

・本授業の方法は続けるべきと感じましたか．(Q10 for discussion, in 2022)
[1] そう思う [2] どちらかというと，そう思う [3] どちらともいえない [4]どちらかというと，そう思わない [5] そう思わない

Q1. Select the best describes about degree of your enjoyment of learning Python in class.
[1] boring [2] if anything, boring [3] neutral [4] if anything fun [5] fun
Q2. How did you feel about the difficulty level from character output to function which you are learning at the same time by C?
[1] easy [2] if anything, easy [3] appropriate [4] if anything difficult [5] difficult
Q3. How did you feel about the progress speed from character output to function which you are learning at the same time by C?
[1] slow [2] if anything, slow [3] appropriate [4] if anything, fast [5] fast
Q4. What did you find in understanding the content of python by comparing it to C?
[1] hard to understand [2] rather hard to understand [3] neutral [4] rather easy to understand [5] easy to understand
Q5. Did you understand the differences and similarities between them by comparing C and Python?
[1] no [2] if anything, no [3] neutral [4] if anything, yes [5] yes
Q6. Did you feel burden to learn Python concurrently with C? (This question was asked in 2022.)
[1] yes [2] if anything, yes [3] neutral [4] if anything, no [5] no
Q7. Did learning Python by comparing it with the C language deepen your knowledge of the "C"?
[1] yes [2] if anything, yes [3] neutral [4] if anything, no [5] no
Q8. Describe your feelings about learning a new language as a stepping stone to a known language.
Q9. Write your impressions of this class.
Q10. Did you feel that this teaching method should be continued?
[1] yes [2] if anything, yes [3] neutral [4] if anything, no [5] no

Q1 indicates the student's final motivation. Q2 and Q3 do whether the classes are being conducted appropriately within the scope of this study. Q4 and Q5 do whether knowledge was acquired in depth through comparison. Q6 does how the students perceive the burden created by the comparison. Q7 does whether they feel that they have experienced relearning or transfer effects for a known language. Q8 provides a clue as to how learning this method has worked for them. Q9 and indicate their impressions of the class as a whole. Q10 is the students' overall evaluation.

In order to reduce the psychological impact of the order of affirmation and negation on the selection results, the order of the display was intentionally changed in the choice-type questions. Results in the next chapter will present the data in a unified order to improve visibility.

The number of responding students in 2021 is thirty-four; in 2022, it is forty. For questions that used options, the percentage of selection for each item has been calculated.

In the next chapter, the results of Q2 - Q5, Q7, and Q8, which concern only the scope of this study, are discussed first. The discussion will then include Q1, Q9, and Q10, which relate to the overall impression of the class, and Q6 which confirms the burden of parallel learning.

Results and Discussion

The results clearly indicate the effectiveness of this method. We will confirm the items of difficulty and speed, effects by comparison, and transition effects.

Table 3 shows the responses regarding difficulty and speed. more than 80% of the students have found the content appropriate or easy. Moreover, more than 80% have felt that the learning speed was slow or appropriate. In other words, less than 20% answered that it was fast. Therefore, this method adapts the majority of the students to fast learning. Furthermore, they have realized it. Table 4 shows how the comparison affected Python learning. More than 72% students said that the comparison made it easier to understand the content. These results indicate that the transition of learning based on existing knowledge is realized in the class and that the students is able to experience the benefits of this transition. More than 88% of the students have felt that they understood the differences and similarities between C and Python. Fewer than 13% of students find the comparison difficult to understand. Thus, learning by comparison works for many students, and few students have problems.

Table 5 shows the effect of relearning C in comparison. More than 67% students answer that their

Table 3. Evaluation of teaching appropriately

Q2: contents level (%)		
Categorize	2021	2022
Easy	29.4	10.0
If anything, easy	32.4	27.5
Appropriate	32.4	42.5
If anything, difficult	2.94	17.5
Difficult	2.94	2.5
Q3: progress speed (%)		
Categorize	2021	2022
Slow	2.94	2.50
If anything, slow	11.8	12.5
Appropriate	79.4	65.0
If anything, fast	5.88	17.5
Fast	0.00	2.50

Table 4. Effect about understanding by comparison.

Q4: understanding by comparing python to C (%)		
Categorize	2021	2022
Easy to understand	29.4	17.5
Rather easy to understand	47.1	55.0
Neutral	17.6	15.0
Rather hard to understand	5.88	12.5
Hard to understand	0.00	0.00
Q5: Understanding differences and similarities (%)		
Categorize	2021	2022
Yes	23.5	37.5
If anything, yes	64.7	55.0
Neutral	0.00	5.00
If anything, no	11.8	2.50
No	0.00	0.00

Table 5. Relearning or transfer effects.

Q7: experienced relearning or transfer effects (%)		
Categorize	2021	2022
Yes	23.5	15.0
If anything, yes	47.1	52.5
Neutral	17.6	20.0
If anything, no	8.82	7.50
No	2.94	5.00

knowledge of C is deepened. The effect is less effective when compared to understanding Python. However, more than half of the students experienced a further understanding of the language they had already learned through relearning.

Finally, we review the qualitative data obtained in Q8. Table 6 shows the incidence of positive and negative factors in students' descriptions as learning effects. Irrelevant responses found in the responses are ignored. Each is counted if a response contains both positive and negative content. About 3/4 of the responses is positive. 61.0% of the positive comments were related to the expected benefits, such as "I understood it faster," "I understand C better," and "I understand Python better now that I know how it differs from C," and so on. Some

Table 6. Rate of impressions about comparative learning

Q8: impressions of comparative learning (%)		
Impression	2021	2022
Positive	73.7	77.5
Negative	26.3	22.5

comments include, for example, "I was not good at C, but learning Python helped me to deepen my understanding of C. In particular, it was easier to understand the function part through Python." and "I was surprised at the simplicity of Python because I could write many lines of code in Python that I had to write in C.

The most frequent negative response was "I was sometimes confused between C and python" which accounted for 9 cases (47.4%) of all negative comments. Three cases said that "the comparison made it difficult."

On the other hand, six of them also noted that the comparison, for example, made it easier to understand. The comments are as follows.

“While there were some similarities, there were also some completely different ways of writing the language, and sometimes it was mixed up with C and sometimes it became a jumble. However, there were many advantages, such as making the C language easier to understand.”

“It was easy to learn because I knew how it worked, but I often got the two languages mixed up.”

“Occasionally, I would confuse C with Python and make mistakes. However, the basic concepts were the same as in C. Therefore, it seemed easier than when I learned programming from scratch.”

“It was easy to understand how the program was executed. However, there were times when I could not tell which was which in the way the code was written in C and Python.”

“I think Python was easy to learn because it has many similarities with C. However, because of this, there were quite a few things that got mixed up and confused in my head. Another factor may have been that I was learning it at the same time as the C language.”

“I feel like it was easier to get into my head than learning all of Python from scratch because of the similarity of the languages. However, it was difficult to learn Python because it seemed to get mixed up with C.”

These results indicate that this method provides significant benefits to students. Moreover, they also explain that the students were able to experience the effects and the difficulties caused by the comparison. The problem is that learning at the same time causes confusion.

From here, we will begin our discussion of this methodology. The topics are as follows.

1. How was the burden on the students?
2. Which scenes can this method be applied?

First, we would like to consider the burden borne by the student. While this method has the advantage of allowing students to reflect on past knowledge, it has the disadvantage of steadily increasing the burden on students. Even if there is a learning effect, it becomes problematic if the burden on students is too large.

Therefore, let us consider the extent to which students consider classes to be a burden.

We have placed a question on the enjoyment of learning Python in Q1, a question asking if there is a burden in Q6, a feedback section in Q9 as a place to express opinions, and an evaluation of teaching methods in general in Q10. Where, the evaluation related solely to the scope of this study is Q6. The other questions include evaluations about classes which are out of study.

Table 7 shows the percentage of responses to Q1 and Q10. Results indicate that many students enjoyed learning Python and think that this style should be continued. These values are influenced by classes out of research target. However, they would not be as high if the students did not also think so within the scope of the study. Thus, it can be evaluated that the students do not evaluate negative about the comparison classes due to burden.

Based on the 2021 results, we surveyed the burden as question in 2022. Table 8 shows the results: 30.0% of students reported some burden, while 42.5% reported little burden. I think the percentage of students who feel burdened is at a level that cannot be ignored. However, we the that the percentage of students who feel less burdened and the aforementioned results taken as a whole do not indicate that this method is unreasonable. This method will be superior if improvements were made to reduce the burden.

Table 7. Evaluation of the enjoyment of learning Python

Q1: enjoyment of learning Python (%)		
Categorize	2021	2022
Fun	38.2	27.5
If anything, fun	50.0	57.5
Neutral	8.82	12.5
If anything, boring	2.94	2.50
Boring	0.00	0.00
Q10: continuation of teaching methods (%)		
Categorize	2021	2022
Yes	-	65.0
If anything, yes	-	30.0
Neutral	-	5.00
If anything, no	-	0.00
No	-	0.00

Figure 8. Burden of comparison work on students.

Q6: existence of burden by comparison (%)		
Categorize	2021	2022
No	-	25.0
If anything, no	-	17.5
Neutral	-	27.5
If anything, yes	-	25.0
Yes	-	5.0

Next, we will consider situations in which this method can be adapted. As mentioned in Q8, a non-negligible number of students reported to confuse the two languages in this method. Therefore, it is necessary to have time to carefully explain the new language and the differences between the two languages. On the other hand, the results of Q2 and Q3 suggest that the speed of instruction could be faster. We think that less confusing units can be taught faster. This method will be applicable if the number of classes is sufficient as a result of considering the balance.

We also think it is important that as many or more similarities between the two languages are exist compared with C and Python. As shown in Q4 and Q7, about 70% of the students felt that the comparison deepened their understanding of the two languages. This number is somewhat low. If many students are unfamiliar with programming, comparison learning may be difficult without many commonalities. On the other hand, several sources, such as Exploring Data (Ramiro, 2019), indicate that the similarity between C and Python is low. Thus, the scope of applicability is not considered narrow.

Finally, this method requires sufficient time in the first year; there are no books that explain the new language by comparing the two languages. Therefore, you must create those texts yourself. One of the determining factors in this method is the availability of time for the teacher in charge.

Conclusions

In this study, we investigated the impact of a class in which students who have already learned C half a year are taught Python in comparison to C. Teachers have students relearn the essentials of C, emphasize the differences between C and Python, and then teach Python code.

The effects by this method have been researched using the questionnaire in the final class in 2021 and 2022 academic year. As a result, the followings is confirmed.

1. this method allows students to experience the benefits of learning from comparison between new programming knowledge and understood one.
2. this allows students to learn new knowledge quickly.
3. many students feel positively about this.

Analysis of qualitative data confirms that the assignment is to reduce language confusion. In addition, about 1/3 of the students feel burdened. This method will be superior if improvements were made to reduce the burden.

We discussed how students feel burdensome and the conditions under which this study can be applied. As a result, we conclude that the burden is not problematic. The applicable conditions may be that there be equal or greater similarity between Python and C, and that sufficient time be allowed for preparation in the first year.

References

Fujishima N. (2019). Effects of step-by-step activation learning in a primary computer programming class. *Proceedings of ISATE*. Yamaguchi: Tokuyama

Ramiro G. (2019). Exploring Data. https://exploring-data.com/info/programming-languages-influence-network/ Reference date: 08-21-2023.