Jurnal Inovasi Teknologi Pendidikan
Volume 10, No. 1, March 2023 (43-54)
Online: http://journal.uny.ac.id/index.php/jitp

Effectiveness of blended learning implementation for algorithm and programming course

Yuricha[1]*, **Irwan Kurnia Phan**[2]
[1] Institut Teknologi dan Bisnis Sabda Setia, Indonesia
[2] Universitas Widya Dharma Pontianak, Indonesia
* Corresponding Author. E-mail: yuricha@itbss.ac.id

ARTICLE INFO

Article History
Received:
19 December 2022;
Revised:
2 February 2023;
Accepted:
28 February 2023
Available online:
09 March 2023.

Keywords
Algorithm and programming; blended learning; classroom action research

ABSTRACT

Algorithm and programming is a prerequisite course at the beginning of the semester, which not only requires an understanding of the basic concepts but also how these concepts can be implemented using a programming language. However, 72% of students have never studied programming at all. This is a challenge for teachers to be able to achieve learning objectives. This study aims to describe the learning design using the Blended Learning model so that its effectiveness can be measured in increasing student understanding in algorithms and programming courses. This research is Classroom Action Research (CAR) with a one-group pre-test post-test design for seven cycles implementing Blended Learning. Through the Wilcoxon Signed Rank test, it was found that there is an average of 92% increase in student understanding from the pre-test to the post-test. Therefore, the application of Blended Learning needs to be done and provides mature readiness for students during face-to-face meetings and teachers can evaluate learning outcomes more quickly and determine the direction of further learning actions.

How to cite:
Yuricha, Phan, I. K. (2023). Effectiveness of blended lerning implementation for algorithm and programming course. *Jurnal Inovasi Teknologi Pendidikan*, 10(1), 43-54. doi: https://doi.org/10.21831/jitp.v10i1.54707

INTRODUCTION

Algorithm and Programming is one of the prerequisite courses in universities. This course has many names, but all have roughly the same content: introducing basic programming. Students are challenged to understand fundamental theories and concepts while simultaneously being able to apply them using a programming language (Lahtinen et al., 2005). The understanding and programming skills gained from this course become the foundation for advanced programming courses in the following semesters.

A good understanding of the Algorithm and Programming course can be a solid foundation in completing around 20-30 percent of the courses related to programming (Dicoding Internal, 2020). Conversely, a lousy understanding can cause a prolonged domino effect on students. One possible result of this domino effect is the delay in student graduation (Agwil et al., 2020). This happens because the prerequisite courses did not reach the target, so students could not take the next course and needed another year to retake the course. The timeliness of graduation is an indicator of assessment and accreditation in tertiary institutions (Widarto, 2017).

Many fundamental theories and concepts in programming are associated with problem-solving skills and an understanding of the mathematical logic (Derus & Ali, 2012). Found that students tend to understand written theories and concepts faster but slower when these theories and ideas have to be implemented into a programming language (Demaidi et al., 2019). So this is not only a challenge for students but also for the lecturers of these courses (Barroso et al., 2018). In the era of student-centered education, the way lecturers orchestrate learning is the key to student success in achieving learning goals which are also contained in learning designs (Shohib, 2018). One approach that can be applied to learning is Blended Learning (Cronje, 2020).

The application of Blended Learning allows learning through face-to-face meetings and the internet as a media (Budiningsih et al., 2019). Several advantages are obtained from this application, such as ease of accessing learning materials, improvement of the quality of learning, and saving on knowledge costs (Stein & Graham, 2014). With a combination of education using the internet as a media, students can access materials and instructions from lecturers from anywhere and anytime (Beaver et al., 2015). Lecturers can also take advantage of this application by enriching students' knowledge and experience in basic programming.

Several previous studies using the Blended Learning approach in Algorithm and Programming courses, such as those carried out by (Bibi & Jati, 2015), showed that the average increase in understanding of students taking these courses was 30.288. This study also showed an increase in student learning motivation by an average of 11.705. The research focuses on comparing conventional learning and learning by applying Blended Learning. The research needs to focus on the learning design and the Blended Learning model used in teaching. Implementing Blended Learning has an essential role in learning because it combines face-to-face and online learning and several learning techniques, learning media, technology, and modes of delivery offline and online (Tang, 2013).

The Blended Learning approach is also applied (Jusuf, 2017) in the Algorithm and Programming course by utilizing games on the www.code.org website. 46% of respondents who take the approach consider learning more varied and preferred. In addition, this model is also liked by as many as 35% of respondents, which is directly proportional to the increase in students' motivation to learn to program. This research does not describe the division between online and face-to-face learning structures and only focuses on how online activities are carried out. A good and balanced learning structure between face-to-face and online learning and how to divide portions between theoretical material and practical material (Demaidi et al., 2019) are essential for lecturers to pay attention to in programming courses.

Described how the learning design implements Blended Learning by dividing online and face-to-face learning structures in programming courses (Zhang & Cui, 2021). The research only focuses on the K-12 level, namely the elementary to high school / vocational school, not the undergraduate level. The learning design also uses scratch at www.code.org, focusing on training computational thinking skills. This will be different if it is applied to undergraduate students. Students are not only required to think computationally (Computational Thinking). Still, they can also apply it using a programming language, which can later form simple programs that solve problems (Hawa et al., 2022).

At the undergraduate level, students must also play an active role in learning independently (Fitriasari et al., 2018; Hendrik et al., 2021). Blended learning is proven to form student independence in learning as long as it is mixed in such a way that students are accustomed to independent learning (Diana et al., 2020). This discipline can also train students from the beginning of the semester as a provision for learning in the following semester's (Bati et al., 2015).

One of the Blended Learning approach models that can be applied is the Flipped Classroom (Muzyka & Luker, 2016a). Implemented Flipped Classroom in programming subjects and showed an increase in students' average scores from 58.8 to 82.75 after this implementation (Zakhia & Dermawan, 2021). Likewise, with research conducted by (Muzyka & Luker, 2016b), the application of Flipped Classroom allows students to access material before and outside class so that during face-to-face meetings, students can discuss and get a more detailed explanation of the theories and concepts involved. To be conveyed by the lecturer.

Learning design also needs to consider students' backgrounds and programming experience (Aleksić & Ivanović, 2013). Early-semester students taking Algorithm and Programming courses often experience difficulties in terms of familiarity with computers, using compilers, and writing specific program code (Demaidi et al., 2019). These three things can also factor in student failure in achieving the objectives of the Algorithm and Programming course. Appropriate learning orchestration can minimize student failure, especially in subjects that require practice (Krpan et al., 2014; Mukhidin et al., 2019).

Based on the description above, this study focuses on learning design by implementing Blended Learning to improve the understanding of undergraduate-level students in Algorithms and Programming courses and calculate its effectiveness.

METHOD

The author conducts Classroom Action Research which is quantitative, by calculating the effectiveness of the application of Blended Learning. Classroom action research is a model that forms a cycle of planning, action, observation, and reflection (Adelman, 1993). This research raises the problem of improving student understanding in Algorithm and Programming courses by applying Blended Learning to learning designs. This research is a pre-experimental rearch involving only one group, namely the experimental group using the One Group Pre-test and Post-test Design to determine the effect of applying blended learning on learning design and measure its effectiveness. This classroom action research focuses on learning design using Blended Learning which calculates the effectiveness at the end of all class action cycles. One cycle is carried out for one week, where the action is carried out at one meeting. This study collected data for seven weeks, thus producing 7 seven cycles from 22 July 2022 to 6 October 2022, as shown in Figure 1.

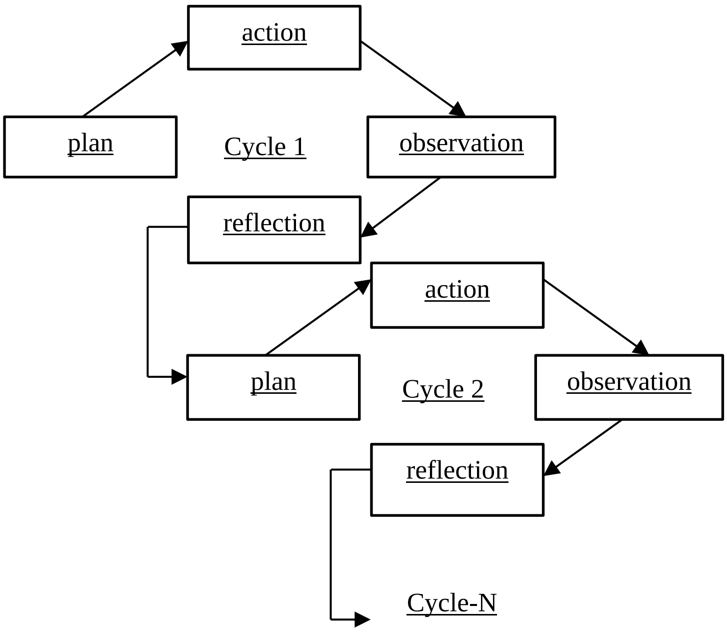


Figure 1. Classroom Action Research Model (Adelman, 1993)

Figure 1 shows that the cycle begins with the planning stage, where the author prepares a lesson plan appropriate for the material, including the application of Blended Learning. After that, proceed with the stages of action. Action stages are carried out through Google Classroom and during face-to-face learning. Then proceed with the observation stage after conducting online and face-to-face activities and end with the reflection stage. The reflection stage will determine the plan's steps in the next cycle up to the seventh cycle.

Respondent

The respondents of this study were active semester one student at the Sabda Setia Institute of Technology and Business. The number of respondents in this research was 76 students who took the Algorithm and Programming course as a prerequisite course. The courses consist of three Semester Credit Units, divided into two credits for face-to-face meetings and one credit for online sessions via Google Classroom.

Data Collection

Data collection was carried out by giving pre-tests and post-tests to respondents in each cycle. The pre-test is carried out at the beginning of the action section, while the post-test is carried out in the reflection section. The pre-test and post-test were given in the form of a survey form with a 5-point Likert scale filled in by students before (pre-test) and after face-to-face meetings (post-test). The survey form provided is a one-time filling that also collects the personal email of the respondent. All answers are analyzed per cycle, and their effectiveness is calculated.

The pre-test describes students' abilities before offline learning while the post-test describes students' abilities after offline learning with the following formula.

$$O1 \times O2 \tag{1}$$

Explanation:

O1	: pre-test value
X	: blended learning application
O2	: post-test value

Learning Design

The Blended Learning approach moderates the learning session using Google Classroom (GC). Every student has a campus email account to access Google Classroom via their mobile or desktop computer. Each student has also joined the same class according to the division of classes on campus so that moderation is carried out per class. The mapping of the Algorithm and Programming material taught for seven meetings/cycles is shown in Table 1.

Table 1. Algorithm and Programming Material Mapping

Meeting	Material
1	Algorithm Basic Definitions and Concepts
2	Data Types, Variables, Constants, and Data Values
3	C# Language Scope – Part 1
4	C# Language Scope – Part 2
5	The Basic Structure of the Branching Algorithms
6	The Basic Structure of Looping Algorithms
7	The Function Paradigm

All these meetings used the C# programming language as a supporting tool in applying programming theories and concepts. Students can choose any compiler to run the C# programming language, such as Visual C#, Visual Studio Code, and online compilers on the internet.

In Table 1, the material for meetings one to three contains more theoretical material, while the rest requires practice. At meeting three, fundamental theories and concepts regarding the scope of C# were thoroughly discussed so that the direct approach could be carried out at meeting four. At meeting 4 (four), students were introduced to compilers and allowed to try directly using the C# programming language code.

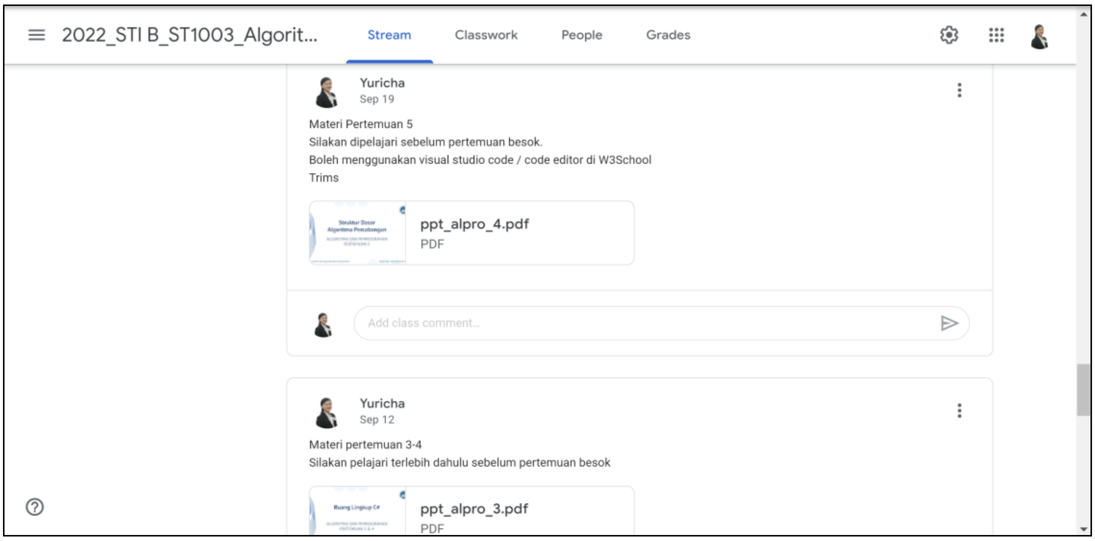
The application of Blended Learning carried out in this study starts from the first to the seventh cycle. There are several Blended Learning models; one of then this study is the Flipped Classroom. Flipped Classroom is a learning model that first provides material for students to study before face-to-face learning is given (Hung et al., 2020). Flipped Classroom has three stages: before learning (pre-learning), during, and after learning (post). These three stages are connected with the settings classroom activities. The Flipped Classroom model is carried out in D-1 meetings and requires students to access these materials. The Flipped Classroom model is connected to the class action, as shown in Table 2.

Table 2. The Relationship between Flipped Classroom and Classroom Action

Flipped Classroom Stage	Classroom Action Stage	Activity
Before learning	Action	Provision of Materials/Modules through GC
During Learning	Action	Apperception
During Learning	Action	Pretest
During Learning	Action	Review
During Learning	Observation	Code Practice
During Learning	Observation	Quiz (if it exist)
During Learning	Observation	Learning Conclusion
After Learning	Reflection	Posttest
After Learning	Reflection	Task (if it exist)

The class action stage begins with the planning stage of learning. In the first cycle, this stage begins with making a lesson plan and asking students to fill out a survey form regarding their educational background and programming experience. In the second cycle and so on, the stages of the plan are prepared based on the reflection results in the previous process.

At this stage, the action begins by providing material on the D-1 meeting through Google Classroom (Figure 2) in the form of a pdf module from the author, which contains (1) Material Title, (2) Course Learning Achievement, (3) learning indicators to be achieved, (4) elaboration of material points, (5) video/online course links as supporting materials, (6) programming code exercises, (7) conclusions, (8) references.



Yuricha
Sep 19

Materi Pertemuan 5
Silakan dipelajari sebelum pertemuan besok.
Boleh menggunakan visual studio code / code editor di W3School
Trims

ppt_alpro_4.pdf
PDF

Add class comment...

Yuricha
Sep 12

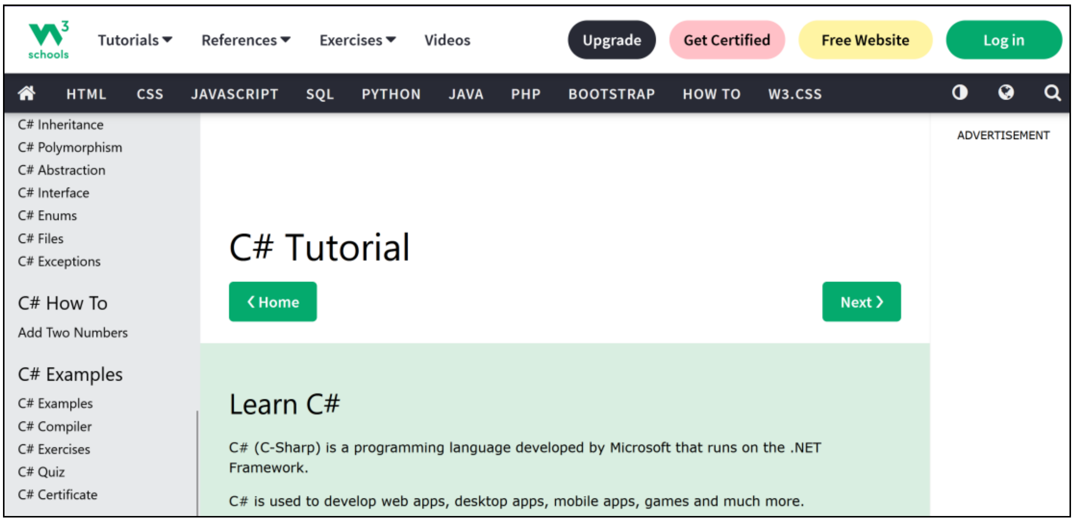
Materi pertemuan 3-4
Silakan pelajari terlebih dahulu sebelum pertemuan besok

ppt_alpro_3.pdf
PDF

Figure 2. Google Classroom Post Thread Page View

Students must have accessed the pdf module before lectures to implement Blended Learning. The video/online course link as supporting material is adapted to the material by mapping the material. The online course link provided is a course that contains material (C# Tutorial), examples of coding/program code (C# Examples), online compilers (C# Compiler), exercises (C# Exercises), and quizzes (C# Quiz) at www.w3schools.com such as in Figure 3.

On the day of the meeting, students are given several activities in the action stages, such as (1) apperception, (2) pretest, (3) review of the given module and focus on discussion on material that students find difficult, (4) question and answer and (5) lecture. Then proceed with the observation stage by providing (1) programming code exercises by providing examples other than those in the module as part of the observation stage, (2) giving quizzes (if any), (3) providing learning conclusions, and finally doing (1) posttest and (2) assignment (if any) as part of the reflection stage.



C# Tutorial

Learn C#

C# (C-Sharp) is a programming language developed by Microsoft that runs on the .NET Framework.

C# is used to develop web apps, desktop apps, mobile apps, games and much more.

Figure 3. C# Tutorial On www.w3schools.com

Data analysis was carried out using the Wilcoxon Signed Rank test on pre-test (O1) and post-test (O2) data in each cycle, with the output interpretation which was divided into three, namely Negative Ranks (to see if there was a decrease in value from pre-test to post-test), Positive Ranks (to see the number of respondents who have increased in value along with the average increase) and Ties (to see if there are respondents who have not changed in value). Through the Wilcoxon test, if the statistical test results with Asymp.Sig. (2-tailed) have a value of more than 0.05, it can be concluded that the application of blended learning in learning design has an influence.

RESULT AND DISCUSSION

Result

In the first cycle, the authors collected some data, such as the respondents' educational background or high school origin, as shown in Figure 4. This data is needed in learning design, especially when emphasizing technical matters in learning.

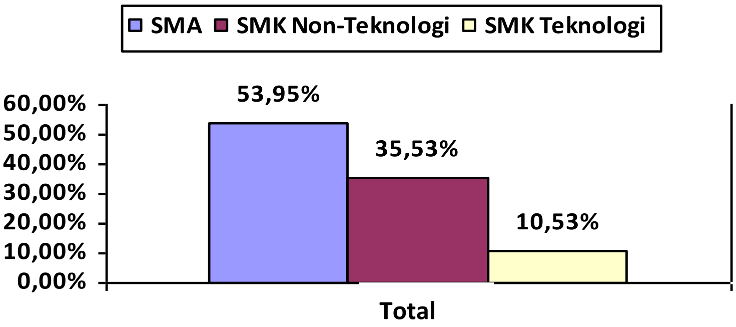


Figure 4. Educational Background

Figure 4 shows that most respondents came from Non-Technology Senior High Schools (SMA) and Vocational High Schools (such as accounting and marketing) that did not study programming in depth. Meanwhile, Technology Vocational Schools (such as Computer and Network Engineering, Software Engineering, and Multimedia) have at least one subject that discusses programming. From these data, it can be concluded that only 10.53% are familiar with programming.

The author also collects data about respondents' experience learning programming before this lecture or before the first meeting material is given in Figure 5.

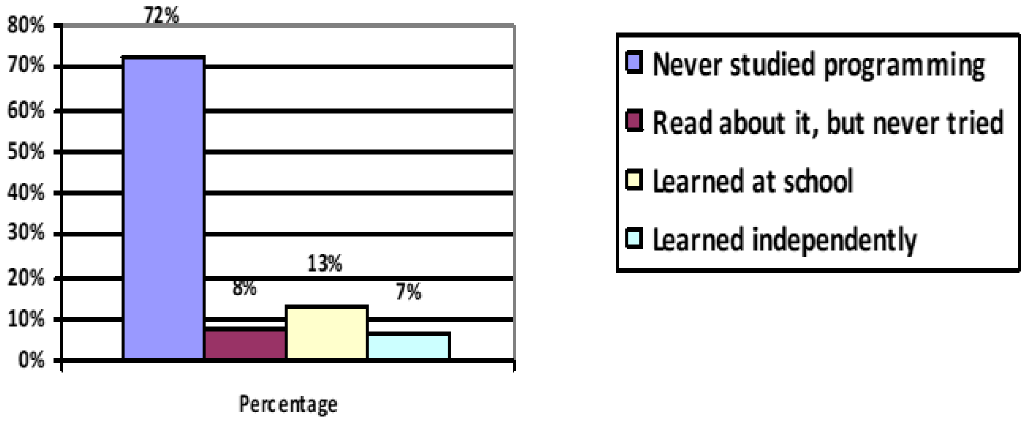


Figure 5. Experience in Learning Programming Before Lecture

Figure 5 shows the next exciting thing: out of the 76 respondents, 72% admitted that they had never studied programming, and 8% had only read about programming but never put it into practice. 80% of new respondents will practice programming for the first time. This fact is a challenge for the author, how learning design should be able to provide understanding as well as proper practice. 13% of respondents studied in schools (both through one subject and from extracurriculars), and 7% of respondents learned programming independently (self-blend).

Furthermore, data were collected from respondents who had studied programming before lectures at school and on their own, regarding the percentage of data from respondents who had studied the C# programming language, as shown in Figure 6. 60% of them had studied C# programming. In contrast, the others had never (learned another programming language besides C#).

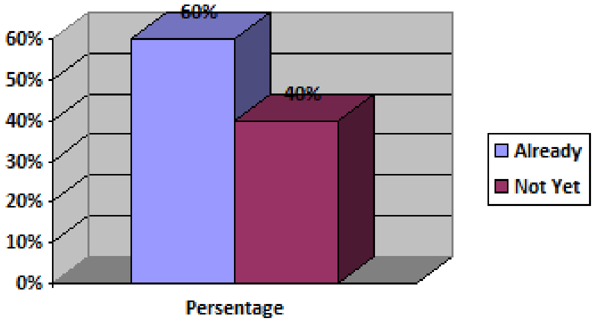


Figure 6. Percentage of Students Who Have Learned Programming in C#

Figure 6 shows that in all the data collected above, the authors realize that this is one of the challenges that need to be addressed and how the existing learning design can provide a good understanding of algorithms and programming, such as emphasizing programming logic and computational thinking. Students are given lectures and examples of cases that involve problems. In addition, the supporting material provided is accessible for students to try, namely, using an online compiler.

In the learning design using Blended Learning, moderation is needed to be able to provide instructions to students. The author also collects data regarding how respondents access video materials and learning links. These data show that the respondents are divided into two: access via a smartphone and a laptop or desktop computer, as shown in Figure 7.

The module provided is a pdf, and the video material is a YouTube link to online courses with a responsive display, allowing students to access the material using a smartphone. 73.68% of respondents admitted that accessing these materials using a laptop or desktop is better. 26.32% of them admitted that they did not have a laptop or desktop computer or saw the ease of accessing material using only a smartphone.

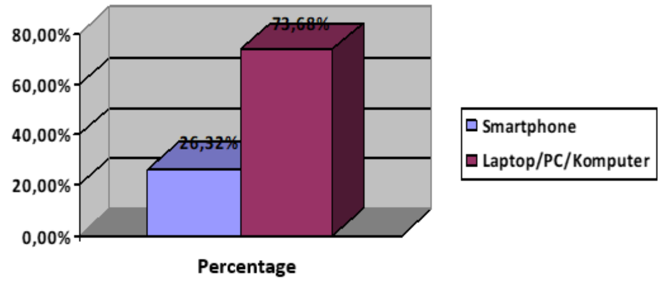


Figure 7. Graph of Devices Used to Access the Learning Material

Then the author also took data from respondents who had accessed video material and online course links and obtained data on what respondents did after getting the link through Google Classroom, which can be seen in Figure 8.

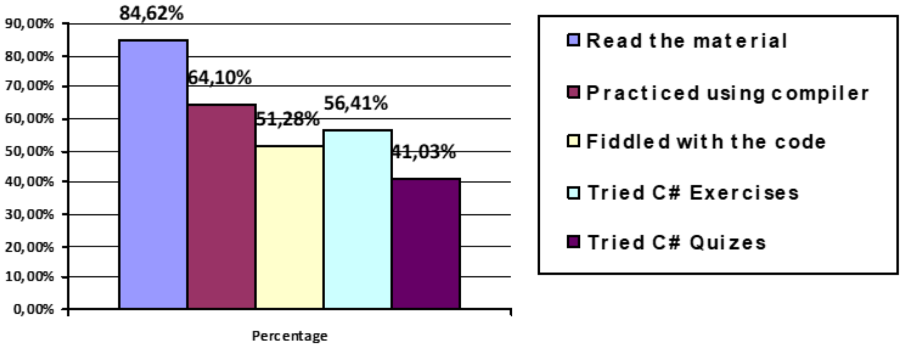


Figure 8. Student Activities in Online Courses

From the data in Figure 8, only 84.62% read the material provided, while 15.38% did not. As many as 64.10% had practiced the material using the compiler, meaning that the rest did not try to practice programming material directly using the existing compiler, so they did not also see the output produced. Furthermore, 51.28% fiddled with the coding examples to get different outcomes so that students could study other inputs for different results and see which writing structures could be changed. Practical material is contained within the compiler section, so if the students did not try it, they were considered to have not practiced the material provided before face-to-face learning. The statistical data on the pre-test and post-test results can be seen in Table 3.

Table 3. Pre-test and post-test statistical data for 76 respondents

Label	Mean	Std. Deviation	Minimum	Maximum
Pre-test 1	.475	1.0557	.0	4.3
Post-test 1	2.862	.7519	1.0	5.0
Pre-test 2	2.554	.4706	1.5	3.5
Post-test 2	4.021	.3492	3.0	4.8
Pre-test 3	2.591	.6567	1.0	4.0
Post-test 3	3.674	.5546	2.0	5.0
Pre-test 4	2.713	.7267	1.0	4.5
Post-test 4	3.838	.6046	2.0	5.0
Pre-test 5	2.905	.5041	1.3	4.3
Post-test 5	3.524	.5501	2.3	4.5
Pre-test 6	2.487	.5772	1.3	3.8
Post-test 6	3.407	.6367	2.0	5.0
Pre-test 7	3.111	.5977	2.0	4.5
Post-test 7	4.011	.5505	3.0	5.0

Discussion

On each cycle at each weekly meeting, data of pre-test and post-test tested using Wilcoxon is collected as part of the reflection stage to evaluate whether the cycle needs to be repeated or not.

The Wilcoxon Signed Rank test uses SPSS v22. The results of the comparison using Wilcoxon shows that there was an increase from the pre-test results to the post-test results for each cycle as shown in Table 4.

Table 4. Results of Pre-test and Post-test using Wilcoxon in 7 Cycles

Cycle	Negative Rank			Positive Rank			Ties
	N	Mean	Sum	N	Mean	Sum	
1	0	.00	.00	75	38.00	2850.00	1
2	0	.00	.00	76	38.50	2926.00	0
3	1	8.50	8.50	72	37.42	2694.50	3
4	1	.00	.00	72	37.40	2692.50	3
5	0	.00	.00	66	33.50	2211.00	10
6	0	.00	.00	68	34.50	2346.00	8
7	0	.00	.00	67	34.00	2278.00	9

The test results shown in Table 4 show that of the seven cycles the lowest number of positive ranks was 66 respondents (in cycles 3 and 4) meaning 10% of the total participants. All cycles show an average of 92% positive rank. The positive rank value indicates that there has been an increase from the pre-test to the post-test. The positive rank values showed a decrease in the 5th to 7th cycles compared to the positive rank values at meetings 1-4. The fifth to seventh cycles were focused on practical/coding material. This is directly proportional to what was said by (Demaidi et al., 2019) that students would be slower to understand when theories and concepts need to be implemented in the form of a programming/coding language. This becomes homework for teachers to be able to convert the 10% to be in a positive rank.

While the negative rank of all cycles shows that only twice has the value of 1, meaning that there is only one respondent that has got a decrease in value. On the 5th to 7th meeting, less than 10% had not increased or decreased in pre-test and post-test values. It can be seen from the statistical test using the Wilcoxon test in each cycle, as shown in Table 5.

Table 5. Results of Pre-test and Post-test Statistics using Wilcoxon in 7 Cycles

Cycle	Z	Asymp. Sig. (2-tailed)
1	-7.552	.000
2	-7.587	.000
3	-7.402	.000
4	-7.389	.000
5	-7.081	.000
6	-7.179	.000
7	-7.128	.000

Table 4 shows that the Asymp. Sig. (2-tailed) which is less than 0.05, indicates an influence of applying blended learning in the learning design. The Z-score averages -7, indicating that the average value is below the pre-test's mean. The author also collects data on how students assess the application of blended learning to help them understand Algorithm and Programming course material, as shown in Figure 9.

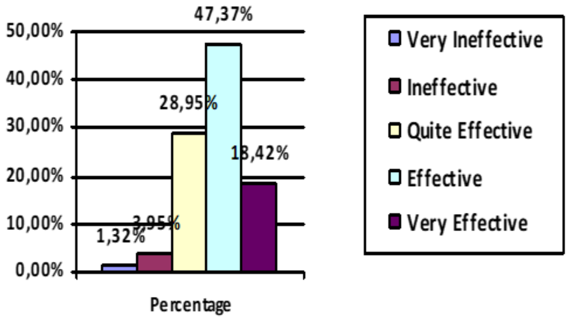


Figure 9. Student Responses to the Application of Blended Learning

Figure 9 shows that 28.95% responded that Blended Learning was quite adequate for them, 47.37% answered that it was effective, and 18.42% responded that the learning was very effective. When combined, the application of Blended Learning tends to give positive results.

CONCLUSION

From the collection and discussion results, it can be concluded that implementing Blended Learning in the Algorithm and Programming course can improve student understanding with an average positive rank of 92%. These results are considered good when viewed from the educational background and experience of learning programming owned by students and are calculated from the duration of learning, which lasts for approximately seven weeks. Further research needs to be tested more precisely, and control variables can be used to see the value of its effectiveness in greater depth. The author found that when implementing Blended Learning, which requires programming practice, students tend to avoid doing it, which impacts the effectiveness of practical/coding material. Students must be encouraged to practice the material and be open to more than just trying coding during face-to-face learning. Applying Blended Learning in Algorithms and Programming courses must be done from the results and conclusions. By combining Blended Learning and CAR, students will be more prepared to learn during face-to-face meetings and be able to explore further the material for the next meeting. Through this application, lecturers can evaluate learning outcomes more quickly and determine the direction of further learning actions.

REFERENCES

Adelman, C. (1993). Kurt Lewin and the Origins of Action Research. *Educational Action Research*, *1*(1), 7–24. https://doi.org/10.1080/0965079930010102

Agwil, W., Fransiska, H., & Hidayati, N. (2020). Analisis ketepatan waktu lulus mahasiswa dengan menggunakan bagging cart. *FIBONACCI : Jurnal Pendidikan Matematika Dan Matematika*, *6*(2), 155–166. https://dx.doi.org/10.24853/fbc.6.2.155-166

Aleksić, V., & Ivanović, M. (2013). Blended learning in tertiary education: A case study. *CEUR Workshop Proceedings*, *1036*(March), 96–103. https://ceur-ws.org/Vol-1036/p96-Aleksic.pdf

Barroso, R., Castro, A., & Rocha, A. (2018). Computer programming as a tool to improve mathematic skills in basic education. *Iberian Conference on Information Systems and Technologies, CISTI*, *1*(1), 1–4. https://doi.org/10.23919/CISTI.2018.8399192

Bati, T. B., Gelderblom, H., & Bilijon, J. Van. (2015). Blended learning of programming in large classes: A reflection of students' experience from an Ethiopian University. *Transform*, 1–13. http://transform2015.net/live/Resources/Papers/Blended learning.pdf

Beaver, J. K., Hallar, B., Westmaas, L., & Englander, K. (2015). Blended learning: Lessons from best practice sites and the Philadelphia context. In *Perc* (Issue 1). http://www.researchforaction.org/wp-content/uploads/2015/11/RFA-PERC-Blended-Learning-Phase-II-Report-September-2015.pdf

Bibi, S., & Jati, H. (2015). Efektivitas model blended learning terhadap motivasi dan tingkat pemahaman mahasiswa mata kuliah algoritma dan pemrograman. *Jurnal Pendidikan Vokasi*, *5*(1), 74–86. https://doi.org/10.21831/jpv.v5i1.6074

Budiningsih, C. A., Haryanto, H., & Rahmadona, S. (2019). The development of blended learning theories of learning course for educational technology students in FIP UNY. *KnE Social Sciences*, *2019*, 170–182. https://doi.org/10.18502/kss.v3i17.4637

Cronje, J. C. (2020). Towards a new definition of blended learning. *Electronic Journal of E-Learning*, *18*(2), 114–121. https://doi.org/10.34190/EJEL.20.18.2.001

Demaidi, M. N., Qamhieh, M., & Afeefi, A. (2019). Applying blended learning in programming courses. *IEEE Access*, *7*, 156824–156833. https://doi.org/10.1109/ACCESS.2019.2949927

Derus, S. R. M. D., & Ali, A. Z. M. (2012). Difficulties in learning programming: Views of students. *1st International Conference on Current Issues in Education (ICCIE2012)*, *1*, 74–78. https://doi.org/10.13140/2.1.1055.7441

Diana, P. Z., Wirawati, D., & Rosalia, S. (2020). blended learning dalam pembentukan kemandirian belajar. *Alinea: Jurnal Bahasa, Sastra, Dan Pengajaran*, *9*(1), 16–22. https://doi.org/10.35194/alinea.v9i1.763

Dicoding Internal. (2020). *5 hal yang kamu harus pertimbangankan ketika mengambil jurusan teknik informatika*. Dicoding. https://www.dicoding.com/blog/jurusan-teknik-informatika/

Fitriasari, P., Tanzimah, T., & Sari, N. (2018). Kemandirian belajar mahasiswa melalui blended learning pada mata kuliah metode numerik. *Jurnal Elemen*, *4*(1), 1–8. https://doi.org/10.29408/jel.v4i1.439

Hawa, D. M., Ghoniem, E., & Saad, A. M. (2022). Integrating problem based learning into blended learning to enhance students' programming skills. *Journal of Positive School Psychology*, *6*(8), 4479–4497. https://www.journalppw.com/index.php/jpsp/article/view/10621

Hendrik, B., Masril, M., & Firdaus, F. (2021). Meningkatkan kemandirian belajar mahasiswa melalui penerapan Blended Learning pada mata kuliah algoritma dan pemrograman 1. *Edukatif : Jurnal Ilmu Pendidikan*, *3*(4), 2192–2198. https://doi.org/10.31004/edukatif.v3i4.1156

Hung, H. C., Liu, I. F., Liang, C. T., & Su, Y. S. (2020). Applying educational data mining to explore students' learning patterns in the flipped learning approach for coding education. *Symmetry*, *12*(213), 1–14. https://doi.org/10.3390/sym12020213

Jusuf, H. (2017). Model blended learning berbasis teknologi informasi dalam pembelajaran algoritma dan pemrograman. *Jutisi*, *6*(2), 1449–1588. https://doi.org/10.35889/jutisi.v6i2.242

Krpan, D., Rosić, M., & Mladenović, S. (2014). Teaching basic programming skills to undergraduate students. *Proceedings of CIET 2014*, *1*(1), 147–158. https://www.researchgate.net/publication/275271925_Teaching_Basic_Programming_Skills_to_Undergraduate_Students

Lahtinen, E., Ala-Mutka, K., & Järvinen, H. M. (2005). A study of the difficulties of novice programmers. *Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, *1*, 14–18. https://doi.org/10.1145/1067445.1067453

Meilisa, R., & Pernanda, D. (2020). Model pembelajaran flipped classroom pada mata kuliah algoritma dan struktur data. *Jurnal Ilmiah Pendidikan Dan Pembelajaran*, *4*(3), 571–577. https://doi.org/10.23887/jipp.v4i3.26928

Mukhidin, M., Mahdan, D., Hasan, B., Hakim, D. L., & Somantri, Y. (2019). Implementation of blended learning methods to improve the ability and learning student results in basic programming subject. *5th UPI International Conference on Technical and Vocational Education and Training (ICTVET 2018) Implementation*, *299*(1), 453–457. https://doi.org/10.2991/ictvet-18.2019.104

Muzyka, J. L., & Luker, C. S. (2016a). The flipped classroom volume 1: Background and challenges. In *The Flipped Classroom* (Vol. 1, Issue 1, pp. 1–3). ACS Books Departement. https://pubs.acs.org/isbn/9780841231627

Muzyka, J. L., & Luker, C. S. (2016b). The flipped classroom volume 2 : Results from practice. In *The Flipped Classroom* (Vol. 1). ACS Books Departement. https://pubs.acs.org/isbn/9780841231627

Shohib, M. (2018). Solusi atau masalah di era revolusi industri 4.0. *Hukum Dan Keadilan*, *1*(1), 87. https://stihpainan.e-journal.id/HK/article/view/4

Stein, J., & Graham, C. R. (2014). Essentials for blended learning: A Standards Based Guide. In *Routledge*. Routledge. https://www.routledge.com/Essentials-for-Blended-Learning-2nd-Edition-A-Standards-Based-Guide/Stein-Graham/p/book/9781138486324

Tang, C. M. (2013). Readiness for blended learning: Understanding attitude of university students. *International Journal of Cyber Society and Education*, *6*(2), 79–100. https://doi.org/10.7903/ijcse.1086

Widarto, W. (2017). Faktor penghambat studi mahasiswa yang tidak lulus tepat waktu di jurusan pendidikan teknik mesin FT UNY. *Jurnal Dinamika Vokasional Teknik Mesin*, *2*(2), 127–138. https://doi.org/10.21831/dinamika.v2i2.16001

Zakhia, F. Y., & Dermawan, D. A. (2021). Penerapan flipped classroom pada mata pelajaran pemrograman berorientasi objek sebagai solusi pembelajaran daring dimasa pandemi di SMK N 2 Surabaya. *Jurnal IT-EDU*, *5*(2), 677–284. https://ejournal.unesa.ac.id/index.php/it-edu/article/view/41555

Zhang, S., & Cui, C. (2021). Implementing blended learning in K-12 programming course: Lesson design and student feedback. *2021 11th IEEE Integrated STEM Education Conference, ISEC 2021*, *1*, 38–44. https://doi.org/10.1109/ISEC52395.2021.9764091