An Approach to Spaced Repetition Methodology for Teaching Markup and Scripting Programming Languages

Afonso Serafim Jacinto
Information Technology Graduate Program
Instituto Federal da Paraíba
João Pessoa, Brazil
0009-0009-1392-787X

Francisco Petrônio Alencar de Medeiros
Information Technology Graduate Program
Instituto Federal da Paraíba
João Pessoa, Brazil
0000-0003-2955-6785

Matheus Pereira de Sousa
Information Technology Graduate Program
Instituto Federal da Paraíba
João Pessoa, Brazil
0009-0003-0791-3013

***Abstract—Contribution*: This research full paper investigates the use of Spaced Repetition (SR) methodology in teaching markup and scripting programming languages to assess its impact on academic performance, knowledge retention and student perception of the learning process.**

***Background*: There is a quest for strategies capable of supporting students and helping them achieve better learning outcomes. The SR strategy, rooted in Ebbinghaus' Forgetting Curve, is widely utilized across multiple domains. This methodology has shown promising results in foreign language and medicine education, although it is not yet extensively explored in computer science education. Since its efficacy has been demonstrated notably in foreign language education, investigating its application within Computer and Engineering education disciplines could be beneficial, as programming languages share learning characteristics common to natural language.**

***Research Questions*: What is the impact of using SR methodology in teaching markup and scripting languages in relation to academic performance, knowledge retention and students' perception of learning?**

***Methodology*: The methodological journey of this research included four phases: (1) Systematic literature review, (2) Development of an SR methodology approach for teaching markup and script languages, (3) Semi-structured interviews, and (4) Planning, execution and analysis of the field experiment. The statistical models used were the difference between means for measurements of the two samples and the chi-square test with two criteria.**

***Findings*: The results indicate significant differences between class grades, validating the influence of Interleaved Study with Spaced Repetition on improving participants' academic performance and content retention, compared to Concentrated Study with Massed Repetition. However, the results do not indicate that the teaching method has an influence on the students' perception of learning.**

Keywords—spaced repetition, interleaved study, computer education, statistical analysis.

I. Introduction

One of the biggest challenges students face in Computer Science courses is overcoming the learning curve in computer programming subjects. Such subjects require significant effort and dedication on the part of the student, with the need for daily and regular study, coding, and solving various exercises. According to Carvalho et al. [1], constant and continuous study is a determining factor that favors the excellent performance of students in learning programming, in addition to the total amount of time dedicated to studying.

Introductory subjects involving programming languages play a fundamental role in the professional training of students in Undergraduate Computing courses. However, despite such importance, these subjects have high failure and dropout rates [2]. Bosse and Gerosa [2] add that students need help understanding basic concepts, commands, and programming logic.

According to Rabêlo Júnior et al. [3], some of the demotivating factors in the learning process of these subjects are difficulties in developing logical reasoning, difficulty understanding abstraction, and the pedagogical approach used. Almeida et al. [4] list other challenges faced in teaching programming: problems of understanding and assigning variables, tracking small pieces of code, and understanding how students plan to solve exercises.

According to Hermans [5], learning a programming language can be compared to learning a natural language because both share similar significant characteristics involving lexical, syntactic, and semantic issues in language acquisition and processing. In this way, Hermans [5] argues about the possibility of standardizing educational techniques and methods to teach natural language for programming education.

A widely used methodology in teaching vocabulary and natural language is Spaced Repetition (SR), also referred to as Spaced Review. This approach is based on the forgetting curve proposed by Hermann Ebbinghaus at the end of the 19th century [6]. Ebbinghaus was one of the first psychologists to study memory and the effect of forgetting and claimed that repetitive study reduces the amount of information we tend to forget over time.

SR is a learning approach that aims to improve and increase long-term retention of information [7]. This technique focuses on reviewing content repeatedly, at optimized time intervals, to ensure that the content the student has the most difficulty with appears more than the content the student already has some mastery of [8].

Although memorizing instructional content to reproduce the information verbatim from memory is not the ultimate goal of education, acquiring fundamental knowledge and quickly accessing the most relevant information from memory are often prerequisites for more effective learning and practical application of knowledge [9].

II. Motivation, Objectives, and Hypotheses

Morin et al. [10] state that Spaced Repetition and Test Effect significantly strengthen long-term memory. SR learning, considered more effective and efficient than the traditional teaching method [11], is widely adopted by high performing students [12]. This combination of SR and

Retrieval Practice, which involves trying to extract information from memory through flashcards, quizzes, or questions, is often associated with success for these students.

Robbes et al. [13] and Lungu [14] investigated the application of SR in the programming study, highlighting the similarities between learning programming languages and developing vocabulary in foreign languages. This relationship arises because both areas have token memorization processes during language learning. In both cases, tokens are elements of an atomic nature and need to be understood in their form and their interconnections with other elements. In natural language, tokens are vocabulary words, whereas in programming languages, tokens include classes, methods, codes, commands, tags, and lexical and syntactic structures, among others [14].

In this context, although the SR methodology is widely studied and applied in teaching foreign languages and medicine, among other areas, its application in teaching computer science subjects still needs to be explored. This opens up broad research opportunities in this area, from developing software to support the methodology and incorporating gamification elements to carrying out experiments with different SR configurations.

In this way, the proposal arose to investigate and propose an approach to the SR methodology for teaching markup and script languages to improve students' academic performance, increase knowledge retention in the medium and long term, and evaluate students' perception of learning about the teaching method adopted.

The choice of this area is justified by the fact that introductory web development teaching does not require students to have prior knowledge of computer programming [15] and can cover a broader range of students with potential interest in the topic. Furthermore, considering the similarity of the approach to learning programming languages with the process of learning natural languages, the positive results observed in the application of SR in teaching foreign languages can be replicated in the educational context of computing.

This work tried to validate three hypotheses through an empirical experiment:

Hypothesis H1: Using the Spaced Repetition methodology in a Distance Education (DE) course on Introduction to Markup and Script Languages increases students' academic performance compared to the Massed Repetition methodology.

Hypothesis H2: Using the Spaced Repetition methodology in a DE course on Introduction to Markup and Scripting Languages increases students' knowledge retention compared to the Massed Repetition methodology.

Hypothesis H3: Using the Spaced Repetition methodology in a DE course on Introduction to Markup and Script Languages increases students' perception of learning about the teaching method compared to the Massed Repetition methodology.

III. Related Works

The main objective of the analyzed works related to using the Spaced Repetition methodology in computer science education is the development of tools that support the learning and memorization process through SR.

Hedy, developed by Hermans [5], presents a new way for beginners to study the syntax of a programming language. It is based on difficulty levels that change slowly and gradually until the user can code. Schimanke et al. [16] developed a mobile tool based on SR that helps in learning complex subjects, and YeckehZaare et al. [17], in turn, present an SR tool, experimenting with variations in its use and correlating this factor with student performance.

Another objective is integrating the SR methodology into introductory programming courses, irrespective of utilizing automated tools. Bothe et al. [18] validate integrating a learning environment based on Flashcard-type Spaced Repetition to existing self-tests in Massive Open Online Courses (MOOC). The goal is to take advantage of existing tests and provide a renewed learning experience. On the other hand, Robbes et al. [13] utilize the SR to introduce a novel learning approach termed API Fluency, aimed at enhancing proficiency in Application Programming Interfaces (APIs).

In computer science disciplines where the SR methodology has been applied, researchers observed that the majority, approximately 67%, concentrate on teaching computer programming. Among the six studies addressing programming teaching, three utilize the methodology in introductory Python courses, one center on the C language, and two explore API programming.

YeckehZaare et al. [17] applied the practical retrieval tool developed in their research to an introductory Python programming course taught over a semester to 193 undergraduate students. The course was designed for students from all knowledge areas, not limited to Computing courses. Hermans [5], who also applied SR in a Python programming course, explains that because it is a more straightforward language with few syntactic elements, it presents advantages in teaching compared to complex languages, such as Java or C++.

Campbell et al. [19], in turn, report on their experiences in introductory Computer Science courses (CS1) taught in Python, offered at a North American university since 2013. In them, concepts are presented through instructional videos and practiced through online exercises. Finally, Denny et al. [20] experimented with an introductory programming course for engineers, this time in C language, which supported the creation of questions and self-testing.

Two studies were found regarding the use of SR methodology in API programming. First, Robbes et al. [13] argue that developers can acquire fluency in APIs using source code repository mining tools and memorization techniques, such as SR methodology. In the second, Lungu [14] reflects on the similarities in the vocabulary learning process of a new natural language and a new API.

The SR methodology has also been applied to other specific computing disciplines, including Relational Databases [16], Computer Architecture [21], Computer Networks, Process Engineering, and Data Science [18].

IV. Experiment Setup

The results of the systematic literature review, which examined the use of the SR methodology in computing teaching [22], combined with the qualitative analysis of semi structured interviews with web development teachers, were used to create a specific Spaced Repetition setup, inspired in the Leitner System, for teaching markup and script languages.

The Leitner System is a simple review scheduling algorithm that prioritizes the repetition of content in which the student has the most difficulty [23]. The configuration proposed in this research is an adaptation of the Leitner System, with the review scheduling scheme occurring weekly instead of daily.

The experiment was carried out in a DE course on Introduction to Markup and Scripting Languages. It was conducted with two samples of students, a control group and an experimental group. Separating students into two samples was essential to enable the application and comparison of different pedagogical methodologies: (i) Concentrated Study with Massed Repetitions and (ii) Interleaved Study with Spaced Repetitions. This way, participants were enrolled in the Virtual Learning Environment (VLE) in two classes, one for each sample. The VLE used was Google Classroom.

The Control Group, identified in the VLE as Class 1, used the Concentrated Study with Massed Repetitions methodology. This approach stands out for the grouping of items to be studied and the continuous presentation of these materials. A distinctive aspect is holding a single review session, which occurs immediately after the completion of each topic covered.

The Experimental Group, labeled Class 2 in the VLE, actively embraced the Interleaved Study with Spaced Repetitions methodology. This method distinguishes itself by distributing study items for interspersed presentation across future study sessions. A crucial aspect of this approach involves conducting multiple review sessions, wherein items correctly recalled during review sessions receive longer intervals between repetitions, while those not remembered accurately undergo shorter intervals.

Although it was necessary to divide the sample into two different classes, it is essential to highlight that both had the same teacher and monitor and used the same VLE, Google Classroom. Furthermore, all students had access to the same teaching materials, synchronous classes, exercises, and review resources. The difference between the classes resided in the methodology used during the course, which influenced the selection and distribution of questions in the weekly activity lists and the sending of review materials corresponding to each situation.

A. Control Group - Concentrated Study with Massed Repetitions

The pedagogical methodology in the control group class was a Concentrated Study with Massed Repetition. This approach stands out for the grouping of items to be studied and the continuous presentation of these materials. In other words, at the end of each theoretical module, students received a list of activities that included all the questions from the respective module, exclusively addressing the most recent content.

A critical aspect of this methodology involves promptly conducting a singular review session after finishing each module. This implies that corrections will occur after students submit a list of activities. Subsequently, students will receive the answer sheet and review materials related to the content corresponding to incorrectly answered questions. These materials and the answer sheet were dispatched via the emails provided during registration and confirmed upon activity submission.

Table I presents the number of questions per module, associating these numbers with the course week in which the questions were assigned to the class. This configuration is standard for the class. Therefore, all group participants received the activities following the criteria established in this table. It is essential to highlight that the lists of activities distributed over the first ten weeks of the course were used to evaluate the participants' performance.

On the other hand, the experiment utilized the Project and the Mock Exam to gauge knowledge retention. We set the deadline for preparing the Project three weeks after completing the modules. The Mock Exam became accessible during the course's last week, two weeks after the theoretical modules concluded.

TABLE I. EXPERIMENT SETUP FOR CONCENTRATED STUDY WITH MASSED REPETITIONS

Activity	**Week**	**Theoretical Modules**										**Mock Exam**	**Total**
		01	**02**	**03**	**04**	**05**	**06**	**07**	**08**	**09**	**10**		
Class	**Week 1**	22	-	-	-	-	-	-	-	-	-	-	22
Class	**Week 2**	-	22	-	-	-	-	-	-	-	-	-	22
Class	**Week 3**	-	-	22	-	-	-	-	-	-	-	-	22
Class	**Week 4**	-	-	-	22	-	-	-	-	-	-	-	22
Class	**Week 5**	-	-	-	-	22	-	-	-	-	-	-	22
Class	**Week 6**	-	-	-	-	-	22	-	-	-	-	-	22
Class	**Week 7**	-	-	-	-	-	-	22	-	-	-	-	22
Class	**Week 8**	-	-	-	-	-	-	-	21	-	-	-	21
Class	**Week 9**	-	-	-	-	-	-	-	-	18	-	-	18
Class	**Week 10**	-	-	-	-	-	-	-	-	-	12	-	12
Project	**Week 11**	-	-	-	-	-	-	-	-	-	-	-	0
Project	**Week 12**	-	-	-	-	-	-	-	-	-	-	-	0
Project	**Week 13**	-	-	-	-	-	-	-	-	-	-	30	30
Total		22	22	22	22	22	22	22	21	18	12	30	235

Throughout the initial seven weeks of the course, we assigned 22 questions after each module. By the eighth module, this number decreased to 21, followed by a reduction to 18 in the ninth module and 12 in the tenth module, summing up to 205 questions. The objective behind diminishing the number of questions in the latter modules aimed to equalize the total questions assigned to both the control and experimental groups within the first ten weeks of the course. Further details regarding the configuration of the methodology applied to the experimental group will be expounded upon in Section IV-B.

Based on investigations conducted by Morin et al. [10], they observed that students prefer shorter quizzes and activity lists containing fewer than 30 questions. Considering this evidence, researchers set a maximum limit of 22 new questions in the study sessions, aiming to prevent activity overload while still allowing for the inclusion of review questions in the activity lists. It is crucial to note that we did not establish a limit for the number of review questions.

B. Experimental Group - Interleaved Study with Spaced Repetitions

The pedagogical methodology in the experimental group class was Interleaved Study with Spaced Repetitions. This approach stands out for the distribution of items to be studied for an interspersed presentation of these materials in future study sessions. Questions relating to a module are divided and presented in different study sessions.

In other words, at the end of each teaching module, students receive a list of activities containing questions from the current module and from previously studied modules, interspersed.

Table II details the number of questions per module, associating these numbers with the course week they were assigned to the class. This configuration is the default for the experimental group, ensuring that all group participants have received the activities according to the criteria established in the table. When analyzing Table II, it is clear that the questions of a module are distributed across four study sessions over four consecutive weeks, starting from the week in which the theoretical module is studied. For the defined number of questions (22), it was decided to divide the questions as follows: 12 in the first session, 6 in the second, 3 in the third, and 1 in the fourth. With this configuration, each week presents a maximum of 22 new questions to be studied, maintaining the same number used in the control group class.

TABLE II. EXPERIMENT SETUP FOR INTERLEAVED STUDY WITH SPACED REPETITIONS

Activity	Week	Theoretical Modules										Mock Exam	Total
		01	02	03	04	05	06	07	08	09	10		
Class	Week 1	12	-	-	-	-	-	-	-	-	-	-	12
Class	Week 2	6	12	-	-	-	-	-	-	-	-	-	18
Class	Week 3	3	6	12	-	-	-	-	-	-	-	-	21
Class	Week 4	1	3	6	12	-	-	-	-	-	-	-	22
Class	Week 5	-	1	3	6	12	-	-	-	-	-	-	22
Class	Week 6	-	-	1	3	6	12	-	-	-	-	-	22
Class	Week 7	-	-	-	1	3	6	12	-	-	-	-	22
Class	Week 8	-	-	-	-	1	3	6	12	-	-	-	22
Class	Week 9	-	-	-	-	-	1	3	6	12	-	-	22
Class	Week 10	-	-	-	-	-	-	1	3	6	12	-	22
Project	Week 11	-	-	-	-	-	-	-	-	-	-	-	0
Project	Week 12	-	-	-	-	-	-	-	-	-	-	-	0
Project	Week 13	-	-	-	-	-	-	-	-	-	-	30	30
Total		22	22	22	22	22	22	22	21	18	12	30	235

It is crucial to emphasize that, similar to the control group, we utilized the activity lists distributed over the initial ten weeks of the course to evaluate the participants' performance. Conversely, the experiment employed the Project and the Mock Exam to assess knowledge retention—the deadline for preparing the Project three weeks after completing the modules. The Mock Exam became available in the course's final week, two weeks after the theoretical modules concluded.

Regarding review sessions, this approach, integrating SR, distinguishes itself by conducting multiple review sessions for the same item. Throughout these sessions, items correctly recalled undergo longer repetition intervals, while those recalled incorrectly experience shorter intervals. In essence, each time a question receives an incorrect answer, it is scheduled for future SR sessions. The scheduling follows the pattern illustrated in Fig. 1, commencing with intervals of one week after the error, followed by two, four, eight weeks, and so forth.

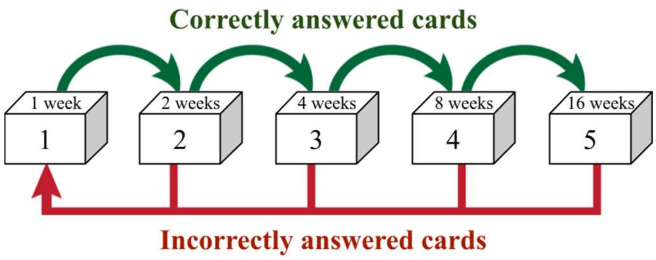

Fig. 1 - Leitner System algorithm adapted for the specific requirements of the experiment.

Therefore, the question will be revisited at progressively longer intervals whenever the answer is correct during review sessions. However, if the answer is incorrect, whether on the first exposure or during a review session, a question review will be scheduled according to the standard schedule, restarting the count from one week after the error. In short, whenever a question is answered incorrectly, it will be reviewed at shorter intervals.

Table III illustrates an example of question tracking and review scheduling for one of the experiment participants. In this case, the table refers exclusively to module 1 of the course, covering 22 questions from 01-01 to 01-22. Weekly, after correcting the activity lists, each participant's worksheets were updated, using green to indicate correct answers and red for incorrect answers. An electronic spreadsheet was created for each student, consisting of 10 pages dedicated to one of the ten study modules.

Question 01-11 serves as an illustration, in which the student assigned an incorrect answer to the question in week 1. For this question, reviews were scheduled in weeks 2, 4, and 8, and in this case, the participant answered correctly in all subsequent review sessions. Question 01-18 is an example of a question that received successive incorrect answers. In this case, the scheduling occurred in weeks 3, 4, 5, and 6, with the participant responding correctly in the latter, allowing the revisions to continue in increasing periods.

TABLE III. EXAMPLE SPREADSHEET FOR CONTROLLING AND SCHEDULING SPACED REPETITIONS FOR THE EXPERIMENTAL GROUP

Question	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10
01-01	x									
01-02	x									
01-03	x									
01-04	x									
01-05	x									
01-06	x									
01-07	x									
01-08	x									
01-09	x									
01-10	x									
01-11	x	1		2				4		
01-12	x									
01-13		x	1		2				4	
01-14		x								
01-15		x								
01-16		x								
01-17		x								
01-18		x	1	1	1	1		2		
01-19			x							
01-20			x							
01-21			x							
01-22				x						

V. MATERIAL AND METHODS

A. *Sample and Procedure*

The experiment was carried out in a DE course on Introduction to Markup and Scripting Languages, involving students from higher education courses in Electrical Engineering, Graphic Design, Control and Automation Engineering, and Computer Networks, among others coming from different higher education institutions in Paraíba and Ceará. The course had its inaugural class on October 23, 2023, and a total duration of 13 weeks, ending on January 22, 2024. The experiment began simultaneously with the course's inaugural class and lasted the same period.

The course received 129 registrations, with 90 students confirming their enrollment upon request. Subsequently, we enrolled and inserted these students into the course's VLE, with 45 students assigned to each class. To select participants from each class, the researcher ordered the list by age in ascending order and alternately made choices. This approach ensured that both classes included students from different age groups, thereby contributing to the quality and validity of the results.

The mentioned course was taught by a teacher, Bachelor in Computing, accompanied by a monitor, a student of the Internet Systems course at IFPB, Campus João Pessoa. Both have extensive experience in web development and in-depth knowledge of HTML, CSS, and JavaScript languages.

During each theoretical module, the course lecturer administered lists of weekly activities to assess the knowledge acquired at each stage and to structure the review materials that each student would receive in the subsequent weeks. Questions were outlined in various formats: true or false, multiple choice, selection, and discursive. It is noteworthy that questions in the discursive modality always necessitated the writing of small snippets of code.

The last three weeks were dedicated to developing a practical Project covering the main topics in each study module. The delivery of the Project was an essential requirement for students to obtain certification of course completion, along with solving a Mock Exam applied in the last week of the experiment.

A description of the minimum requirements to be considered in the evaluation was provided for implementing the Projects, including structuring the HTML document, styling with CSS, and inserting lists, images, and tables. The Project was individually corrected manually by the teacher since, based on research carried out and consultation with interviewed teachers, no automatic correction tool for markup language codes was found.

The Mock Exam of the course was the last assessment activity applied in the classes. The mock exam was administered two weeks after the completion of the theoretical modules, during the last week of the experiment. It included true or false, multiple choice, and discursive questions. Initially, using a tool that blocked the browser tab was considered, intending to prevent students from using artificial intelligence applications or carrying out direct searches in the main web indexers. However, the decision was to maintain the same tool used in weekly activities due to the student's prior familiarity with it and to prevent possible problems or dropouts.

B. *Instruments and Measures*

To test hypothesis H1: "The use of the Spaced Repetition methodology in a DE course on Introduction to Markup and Script Languages contributes to increasing students' academic performance compared to the Massed Repetition methodology," the statistical test used was the difference between means to compare two distinct samples: students who did not receive support from the SR methodology (control group) and students who used the SR methodology during the study period (experimental group). This technique was chosen due to the characteristics of the experiment, including the level of measurement, size, and normality of the sample, test requirements, and the need to compare the performance of two samples after completing the ten modules of the course, minimizing possible variables that could interfere in the experiment.

The difference between means test was carried out at the end of the last teaching module, and the means obtained by the two classes in each of the ten topics were considered. The experiment considered only students whose grades had completed at least 40% of the weekly assessment activities. All individual performances of students and classes were recorded through electronic spreadsheets using the Google Sheets tool. Statistical calculations and tests were performed using Google Sheets and Microsoft Excel to produce consistent results and increase confidence in the accuracy of the conclusions. The hypothesis was tested with a significance level of 0.05, representing a sampling error of 5%.

The evaluation of hypothesis H2: "The use of the Spaced Repetition methodology in a DE course on Introduction to Markup and Scripting Languages contributes to increasing students' knowledge retention compared to the Massed Repetition methodology," was carried out by comparing the class averages after delivery of the final course Project and resolution of the Mock Exam, applied after completion of the ten theoretical modules. The standard deviation of the means was also calculated to observe the dispersion or variability of the data.

To verify hypothesis H3: "The use of the Spaced Repetition methodology in a DE course on Introduction to Markup and Script Languages contributes to increasing students' perception of learning about the teaching method when compared to the Massed Repetition methodology" was used a questionnaire adapted from the Teaching Method Assessment Questionnaire concerning Motivation and Perception of Learning proposed by Moura et al. [24]. The statistical test used to test hypothesis H3 was the chi-square test, which offers a robust approach to analyzing the association between categorical variables due to the characteristics of the measurement level, the type of instrument used in the test, and the nature of the hypothesis tested.

VI. Results and Discussion

The method used to test hypothesis H1 was comparing means for unpaired data. The T-test for the difference between means for different samples assumes that the analyzed samples are independent; that is, there is no direct correspondence between the observations of the two samples. The observations in one sample are not affected or influenced by observations in the other sample. The T-test was applied to determine whether there was a significant difference between the means of the two samples. For this, the averages of the two classes were calculated separately: the average per topic and the general average for the class. Performance by class can be seen in the graph in Fig. 2.

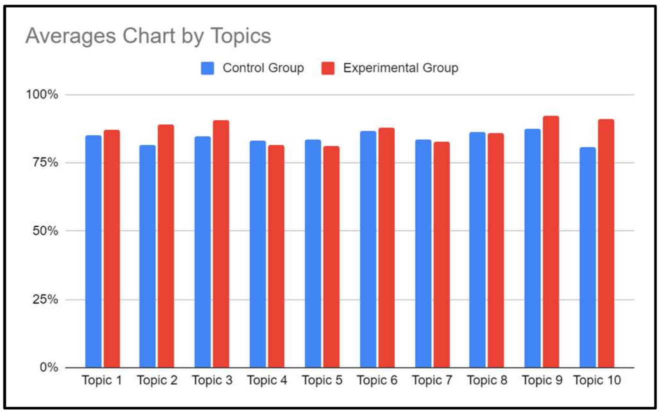

Fig. 2 - Class averages across the course's ten topics

The test of hypothesis H1 sought to verify whether the use of the SR methodology in a DE course on Introduction to Markup and Script Languages contributes to increasing students' academic performance. This means the comparison test was carried out based on the null hypothesis of no significant difference related to the use of the SR methodology against the alternative hypothesis of a significant difference occurring.

To reject the null hypothesis, the researchers utilized a significance level of 0.05, adhering to the convention where T-Tests for two unpaired samples deem a p-value below 0.05 indicative of rejecting the null hypothesis (H0). The p-value, representing the probability of obtaining the sample data under the assumption that the null hypothesis is true, is a crucial metric in hypothesis testing.

Initially, a variance comparison test of the means of the two samples was conducted using the F-test. This test assesses the likelihood that the two samples originate from populations with similar variances. The resulting F-Test value was 0.0984286738, surpassing the significance level of 0.05, which was also utilized in this preliminary examination. This outcome does not warrant the rejection of the null hypothesis (H0), indicating a supposed equivalence in variances between the samples.

Therefore, the T-test was performed to assess whether there is a statistically significant difference between the final means of the two samples, assuming equivalent variances.

Table IV below presents this test's results, which was generated using the Excel tool.

The statistical analysis revealed a significant difference between the overall performance averages of the classes, validating Hypothesis H1. This calculation observed the averages of 10 modules corresponding to the two classes. With a degree of freedom of df = 18, the probability of obtaining the observed effect was p-value = 0.048205507, thus remaining below the pre-established significance value of 0.05. This results in the rejection of the null hypothesis. The control group obtained an average of 84.35, while the experimental group obtained an average of 86.89, indicating a superior performance within the 95% confidence interval.

TABLE IV. T-TEST: TWO SAMPLES ASSUMING EQUIVALENT VARIANCES

	Control Group	Experimental Group
Mean	0.84355	0.86895
Variance	0.000500025	0.001596623
Observations	10	10
Pooled Variance	0.001048324	
Hypothesized Mean Difference	0	
df	18	
t Stat	-1.754167165	
P(T<=t) one-tail	0.048205507	
t Critical one-tail	1.734063607	
P(T<=t) two-tail	0.10	
t Critical two-tail	2.10092204	

Fig. 3 displays a line graph that illustrates the performance of the classes over ten weeks of study, making it easier to visualize variations in performance over time. The blue line represents the average performance of the 25 Class 1 participants who followed the Concentrated Study with Mass Reviews methodology. Meanwhile, the red line represents the average performance of the 24 Class 2 participants who adopted the Interleaved Study with Spaced Repetition methodology. Although the graph does not provide a definitive conclusion to test the hypothesis already verified by the statistical test, it contributes to the observation of the superior performance of the experimental group concerning the control group, especially in the first and last weeks of the study.

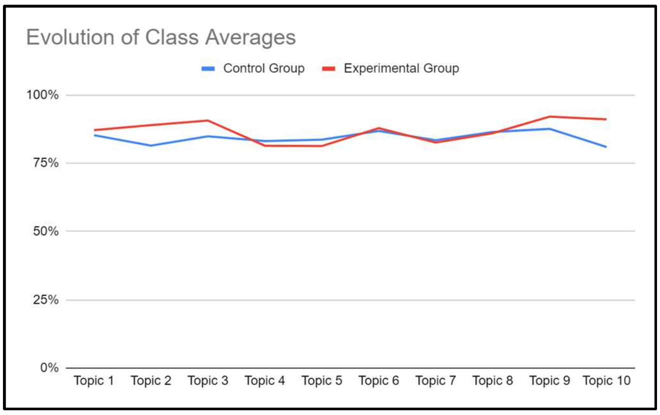

Fig. 3 - Comparison of class averages across the course's ten topics

Hypothesis H2 aims to measure content retention by developing Projects and applying a Mock Exam. The metrics chosen for evaluation are the average and standard deviation of the scores obtained by the participating classes. The average is a central measure that provides an overview of class performance, allowing a basic understanding of the level of knowledge retention. The standard deviation provides information about the variability of the data around the means. The use of both metrics makes this analysis more robust.

The results obtained in the final course Project demonstrate a significant variation between the scores of the two classes. The experimental group presented a higher average (83.04) than the control group (74.29), indicating a higher overall performance. Furthermore, the standard deviation of the experimental group's means (11.09) was lower than that of the control group (18.00), suggesting greater consistency in the scores obtained.

Regarding the coefficient of variation, the value for the control group was approximately 24.24%, which indicates a relatively high variability about the average, suggesting that the data in Class 1 were more dispersed concerning the average. On the other hand, the coefficient of variation for the experimental group was around 13.35%, which indicates relatively low variability to the average, suggesting that the data in Class 2 are less dispersed concerning the average, indicating greater homogeneity of the data or a smaller amplitude.

TABLE V. CLASS SCORES IN THE FINAL COURSE PROJECT

Class	Mean	Standard Derivation	Coefficient of Variation
1	74.29	18.00	24.24%
2	83.04	11.09	13.35%

Regarding the results obtained through the Mock Exam, Table VI shows that both classes have similar averages. However, the experimental group (83.94) maintains a slight advantage over the control group (81.21). Regarding standard deviations, both are extremely low in both classes (0.10), indicating a high consistency in the scores obtained by students during the Mock Exam. Furthermore, both coefficients of variation are extremely low, with the control group (0.1231%) and the experimental group (0.1190%) indicating that data variability concerning the mean is minimal for both classes and suggesting that students in both classes presented consistent results, with little dispersion around the mean.

TABLE VI. CLASS SCORES IN THE MOCK EXAM

Class	Mean	Standard Derivation	Coefficient of Variation
1	81.21	0.10	0.1231%
2	83.94	0.10	0.1190%

When comparing the results of the Project and the Mock Exam, it is possible to observe that the experimental group obtained a superior result in both evaluation modalities. The higher mean and lower standard deviation suggest greater effectiveness in content retention for this group compared to the control group, thus validating hypothesis H2.

These findings suggest that the Spaced Repetition methodology, adopted in class 2 (experimental group), may be contributing to better assimilation and retention of the proposed content, resulting in superior and more consistent performance, both in the Project and in the Mock Exam, compared to class 1 (control group). This analysis reinforces the importance of evaluating different pedagogical approaches to identify the most effective ones in learning.

To examine hypothesis H3, the researcher employed the two-criteria chi-square test, a robust statistical tool tailored for assessing the dispersion between two nominal variables, thereby elucidating their potential association. Administered immediately following the Mock Exam of the Introduction to Markup and Script Languages course, a comprehensive questionnaire comprising 14 queries rated on a Likert scale was distributed among participating students. Within this construct, the primary focus rested on question Q1 as the independent variable, while the remaining 13 questions were designated as dependent variables within the analytical framework. Through this meticulously designed approach, we aimed to scrutinize the intricate interplay between the variables, thereby contributing to a nuanced understanding of their relational dynamics.

The test at this stage seeks to verify whether there is a relationship between the values of the independent variable "evaluation of the teaching method" and the specific values of the other dependent variables. The first question aims to assess students' general assessment of the methodology used. Questions two to seven are related to indicators of intrinsic motivation: interest, involvement, effort, concentration, satisfaction, and number of questions. Questions eight to 14 aim to explore indicators of learning perception: trust, interrelationships, and applicability of knowledge.

The null hypothesis (H0) for the chi-square test states that the relationship between the independent and dependent variables in the answers to the forms for the two classes does not differ significantly. Considering a significance level of 0.05, the chi-square tests for each relationship between independent and dependent variables consider that samples with a p-value below 0.05 reject the null hypothesis.

Applying the chi-square test, p-values < 0.05 would indicate significant differences obtained through the Evaluation Questionnaire. However, only the Q1-Q10 relationship obtained a result lower than the pre-established significance level. It is therefore concluded that the significant difference in only one of the seven questions in the results of the experimental group does not indicate that the teaching method influences the students' perception of learning, thus invalidating hypothesis H3.

TABLE VII. CHI-SQUARE TEST RESULTS TO VALIDATE HIPOTESIS H3

Class	p-value	Q8	Q9	Q10	Q11	Q12	Q13	Q14
1	Q1	0.144	0.088	0.073	0.564	0.086	0.687	0.081
2	Q2	0.176	0.090	<u>0.046</u>	0.532	0.163	0.286	0.078

VII. CONCLUSION

This study investigated and proposed an approach to the Spaced Repetition methodology for teaching markup and scripting languages to improve students' academic performance, increase knowledge retention in the medium and long term, and evaluate students' perception of learning in relation to the teaching method adopted.

Based on the results obtained, the interleaved study with spaced repetition favors students in achieving better academic performance. It increases knowledge retention in the medium and long term, compared to Concentrated Study with Massed Repetitions. This analysis reinforces the importance of evaluating different pedagogical approaches to identify the most effective ones in learning.

There was no significant difference when applying the chi-square test regarding the increase in students' perception of learning concerning the teaching method. This result does not indicate that the teaching method influences students' perceptions of learning.

The research opportunities in the field of Spaced Repetition methodology are diverse, covering the development of software to support the methodology, the implementation of gamification mechanics, game development, the development of chatbots, and mainly, the exploration of different configurations of the Spaced Repetition methodology for teaching, either independently or in association with other pedagogical approaches, within the scope of teaching subjects in the area of Computer Science.

REFERENCES

[1] L. S. G. Carvalho, B. F. Gadelha, F. G. Nakamura, D. B. F. Oliveira and H. H. T. Oliveira. “Ensino de Programação para Futuros Não-Programadores: Contextualizando os Exercícios com as Demais Disciplinas de mesmo Período Letivo”. in Anais do XXIV Workshop sobre Educação em Computação (WEI 2016), 2016.

[2] Y. Bosse and M. A. Gerosa. “Why is programming so difficult to learn? Patterns of Difficulties Related to Programming Learning Mid-Stage”. in ACM SIGSOFT Software Engineering Notes, vol. 41, n. 6, pp. 1–6, 2017.

[3] D. J. L. Rabêlo Júnior, C. d. S. Soares Neto, A. C. Raposo and L. A. d. Santos Neto. “Cosmo: Um ambiente virtual de aprendizado com foco no Ensino de Algoritmos”. in Anais do XXVI Workshop sobre Educação em Computação (WEI 2018), 2018.

[4] A. Almeida, E. C. d. Araújo and J. C. A. d. Figueiredo. “Avaliando a Construção do Conhecimento em Programação Através da Taxonomia SOLO”. in Anais do XXXI Simpósio Brasileiro de Informática na Educação (SBIE 2020), 2020.

[5] F. Hermans. “Hedy: A Gradual Language for Programming Education.”. in ICER '20: Proceedings of the 2020 ACM Conference on International Computing Education Research, pp. 259–270, 2020.

[6] G. K. Pergher and L. M. Stein. “Compreendendo o esquecimento: teorias clássicas e seus fundamentos experimentais”. in Psicologia USP [online], vol. 14, n. 1, pp. 129–155, 2003.

[7] B. Settles and B. Meeder. “A trainable spaced repetition model for language learning”. in Proceedings of the 54th annual meeting of the association for computational linguistics, vol. I, pp. 1848–1858, 2016.

[8] T. T. Gomes, A. Saffarzadeh, M. Severo, M. J. Guimarães and M. A. Ferreira. “A novel collaborative e-learning platform for medical student - ALERT STUDENT”. in BMC Medical Education, 2014.

[9] S. H. K. Kang. “Spaced Repetition Promotes Efficient and Effective Learning: Policy Implications for Instruction”. in Policy Insights from the Behavioral and Brain Sciences (PIBBS), vol. 3, n. 1, pp. 12–19, 2016.

[10] C. Morin, J. M. Hostetter, J. Jeudy, W. C. Kim, J. A. McCabe, A. C. Merrow, A. M. Ropp, N. S. Shet, A. S. Sidhu and J. S. Kim. “Spaced radiology: encouraging durable memory using spaced testing in pediatric radiology”. in Pediatric Radiology, vol. 49, pp. 990-999, 2019.

[11] A. Lambers and J. J. Talia. “Spaced Repetition Learning as a Tool for Orthopedic Surgical Education: A Prospective Cohort Study on a Training Examination”. in Journal of Sirurgical Education, vol. 78, issue 1, pp. 134–139, 2021.

[12] R. R. Landoll, L. D. Bennion and L.A. Maggio. “Understanding Excellence: a Qualitative Analysis of High-Performing Learner Study Strategies”. in Medical Science Educator, vol. 31, issue 3, pp. 1101–1108, 2021.

[13] R. Robbes, M. Lungu and A. Jane. “API Fluency”. in IEEE/ACM 41st International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER), 2019.

[14] M. Lungu. “Designing Personalized Learning Environments through Monitoring and Guiding User Interactions with Code and Natural Language”. in EASEAI 2019: Proceedings of the 1st ACM SIGSOFT International Workshop on Education through Advanced Software Engineering and Artificial Intelligence, pp. 5–8, 2019.

[15] T. H. Park and S. Wiedenbeck. “Learning Web Development: Challenges at an Earlier Stage of Computing Education”. in Proceedings of the Seventh International Workshop on Computing Education Research, pp. 125–132, 2011.

[16] F. Schimanke, R. Mertens, O. Vornberger and S. Vollmer. “Multi Category Content Selection in Spaced Repetition Based Mobile Learning Games”. in 2013 IEEE International Symposium on Multimedia, 2013.

[17] I. YeckehZaare, P. Resnick and B. Ericson. “A Spaced, Interleaved Retrieval Practice Tool That is Motivating and Effective”. in

Proceedings of the 2019 ACM Conference on International Computing Education Research, pp. 71–79, 2019.

[18] M. Bothe, J. Renz and C. Meinel. "On the Acceptance and Effect of Recapping Self-Test Questions in MOOCs". in 2020 IEEE Global Engineering Education Conference (EDUCON), 2020.

[19] J. Campbell, A. Petersen and J. Smith. "Self-Paced Mastery Learning CS1". in Proceedings of the 50th ACM Technical Symposium on Computer Science Education, pp. 955–961, 2019.

[20] P. Denny, E. Tempero, D. Garbett and A. Petersen. "Examining a Student-Generated Question Activity Using Random Topic Assignment". in ITiCSE '17: Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education, pp. 146–151, 2017.

[21] G. L. Herman, Z. Cai and T. Bretl. "Comparison of Grade Replacement and Weighted Averages for Second-Chance Exams". in ICER '20: Proceedings of the 2020 ACM Conference on International Computing Education Research, pp. 56–66, 2020.

[22] A. S. Jacinto and F. P. A. Medeiros. "Levantamento do Estado da Arte sobre o uso da Metodologia de Repetição Espaçada no Ensino de Computação". in Workshop sobre Educação em Computação (WEI), vol. 30, pp. 404-415, 2022.

[23] S. Reddy, I. Labutov, S. Banerjee and T. Joachims. "Unbounded Human Learning: Optimal Scheduling for Spaced Repetition". in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

[24] H. F. N. Moura, L. A. B. Oliveira, A. R. Venosa, L. H. M. Lourenço and J. E. Baroneza, "Uma estratégia para avaliação da percepção de docentes e discentes acerca dos métodos de ensino." in Revista Brasileira de Educação Médica, vol. 46, n. 2, 2022.