Pauses and Spacing in Learning to Program

Leo Leppänen
University of Helsinki
Dept. of Computer Science
leo.leppanen@helsinki.fi

Juho Leinonen
University of Helsinki
Dept. of Computer Science
juho.leinonen@helsinki.fi

Arto Hellas
University of Helsinki
Dept. of Computer Science
arto.hellas@cs.helsinki.fi

ABSTRACT

Conventional wisdom holds that time is an integral part of the learning process. Spacing out learning over multiple study sessions seems to be better for learning than having a single longer study session. Learners should also take pauses from the learning process to absorb, assimilate, and analyze what they have just learned. At the same time, pausing too often can be harmful for learning. Participants of two subsequent introductory programming courses completed programming tasks in an integrated development environment that saved detailed logs of their actions, including time stamps of all the participants' keypresses in said environment. Using this data with background variables and a self-regulation metric questionnaire, we study how the students space out their work, identify trends in between the kinds of pauses the participants took and the course outcomes, and their connection to background variables. Based on our research, students tend to space out their work, working on multiple days each week. In addition, a high relative amount of pauses of only a few seconds correlated positively with exam scores, while a high relative amount of pauses of a few minutes correlated negatively with exam scores. Student pausing behaviors are poorly explained by traditional self-regulation measures such as the Motivated Strategies for Learning Questionnaire and other background variables.

CCS Concepts

•**Social and professional topics** → **Computing education;** *CS1;* •**Applied computing** → *E-learning;* Computer-assisted instruction;

Keywords

educational data mining; pausing; self-regulation; source code snapshots; spacing

Koli Calling 2016, November 24 - 27, 2016, Koli, Finland

ISBN 978-1-4503-4770-9/16/11…$15.00

DOI: http://dx.doi.org/10.1145/2999541.2999549

1. INTRODUCTION

Learning to program presents unique challenges to students. They need to learn to use complex building blocks to build even more complex systems, creating rules and algorithms that are often far from those they encounter in every-day life. Concurrently, they often multi-task by — for example — editing multiple source code files and referring to a technical manual, the course material or error logs. At the same time, they are working on a computer filled with distractions such as social networking sites and games, and might have a mobile device with instant messages popping up every now and then. It is clear that students working on programming assignments are naturally disposed towards working in a way that is classically linked to a bad recollection of learned information.

Due to this disposition, research conducted on other kinds of learning environments, such as lectures, may not be immediately applicable to students working on programming assignments. It is known that a few short pauses used for rehearsal during lectures helps recall [33], but does that apply to learning a task such as programming? At the other extreme, task switching, a "necessary evil" in the context of programming, is known to have a recovery period during which the performance at a task is poorer [26]. Wherein lies the point where pauses turn from beneficial to disadvantageous?

In this work, we explore how students take pauses and space their work in a programming course, and study how this behavior reflects on their exam scores. The analysis is conducted using data collected from within the working environment, as well as background questionnaires and exam scores. Through the analysis, we seek to identify pausing and spacing patterns correlated with good or bad course outcomes, which instructors could use to direct students towards good working habits. Moreover, we wish to raise discussion on the transferability of previous work on pausing and spacing, that is often associated with recall, to the context of learning a task such as programming.

This article is organized in the following manner. Section 2 describes the theoretical background and the previous research this article builds on. Section 3 presents our research questions, our data set, and the context in which we operate. Section 4 describes our results and the methodologies used to reach them. Section 5 further discusses the results and their implications, ties our results into the previous work presented in Section 2, as well as details some limitations of our work. Finally, Section 6 presents the key conclusions and details possible future research avenues.

2. BACKGROUND

The benefits and disadvantages of pauses in a learning process have been studied extensively from multiple points of view. Section 2.1 will shortly go over the previous work on how pauses and time are beneficial for learning, while Section 2.2 will detail how pauses from the primary task, especially in the form of multitasking, can be harmful for learning. Section 2.3 discusses previous research on pauses taken while programming. Finally, Section 2.4 will provide a quick overview of research on student self-regulation and its effects.

2.1 Benefits of Taking Pauses

Distributing — or 'spacing' — learning over a period of time has been shown to produce better learning results than massed learning or 'cramming'. That is, students who study the same set of material for the same overall time tend to perform better in tests if the studying is done in multiple spaced chunks instead of in a single session [10].

This effect is dependent of the lengths of both the retention interval (time between the last study session and testing) and the lengths of the inter-study intervals (time between two study sessions). The optimal inter-study interval is dependent on the retention interval, with longer retention intervals requiring longer inter-study intervals for the optimum effect. For example, for a medium length retention interval of around a month, previous studies suggest that one day inter-study intervals are better than short intervals of either a few minutes to hours, or long multi-day intervals [7].

Research also shows that pausing is beneficial within the context of a single problem, due to the dual phenomena of *fixation* and *incubation* [35]. Fixation is the phenomenon of a problem solver being 'stuck' in a non-productive approach to the problem and needing time spent off-task to get unstuck. Incubation is the act of spending time off-task, allowing for the subconsciousness to solve the problem. The classic idea of 'sleeping on a problem' is an example of the incubation effect. Via these two phenomena, time spent off-task can be beneficial for problem solving. [35]

Sleep has been linked to memory consolidation and enhancement of performance in multiple (but not all) tasks, with some skills only developing during sleep [36]. Similarly sleep (and time in general) have been shown to boost the ability to infer further relations between previously learned facts [14].

Debriefing and reflecting, both alone and in a group, on previous actions, ideas and schemes is also seen as a critical part of learning in literature [6, 25, 27].

Multitasking can also be seen as the act of taking short pauses from a primary task to momentarily focus on other, secondary tasks. Literature suggests that minor levels of multitasking can have a beneficial effect on learning by reducing boredom, which is detrimental to learning [17]. Previous research has for example shown that doodling improved recall in an auditory recall task [1].

2.2 Disadvantages of Pausing

While much research has gone into the benefits of pauses, a body of work also discusses the disadvantages of taking pauses.

As noted above, multitasking can be seen as the act of taking very short pauses from a primary task to complete other secondary tasks. Research has shown that 'task-switching' — the act of changing your focus — causes slower response and more errors [26]. This is in accord with other research that indicates that lowering the students' cognitive load helps them learn complex information [30].

Much research has also gone into how multitasking in a learning environment affects learning. The research shows that multitasking using an electronic device is detrimental to learning and predictive of worse academic success in multiple contexts. Rosen et al. [32] found that receiving and sending text messages while watching a video lecture had a detrimental effect on recall. Hembrooke & Gay [18] determined that using a laptop during a lecture affects recall negatively. Sana et al. [34] further determined that the negative effect of using a laptop is not limited to just the student using the laptop, but to the area surrounding the student. Similar studies have been conducted for example by Wood et al. [39] and Junco & Cotten [19].

These results appear to indicate that programming as an activity has by nature many aspects that can potentially be detrimental to learning: programmers often work on multiple files at a time while simultaneously referencing technical manuals and documentation. Because of this, programming may inherently require multitasking as well as task-switching ability from the programmer [20].

2.3 Effects of Pauses While Programming

Leppänen et al. [21] discovered that high relative amounts of short pauses of 1 to 2 seconds were correlated with high exam scores, and that high relative amounts of longer pauses of 5 seconds to 5 minutes were correlated with lower exam scores. Analysis of basic student background variables further revealed that most background variables such as handedness, age, educational background and year of studies did not have statistically significant correlations with the relative amounts of pauses correlated with low exam scores. Student programming background — both in general and in the course's programming language in particular — was weakly correlated with the relative amount of such pauses.

Blikstein [5] analyzed pausing as a part of the programming process in a qualitative fashion. Blikstein noticed that some students took long pauses wherein they either browsed other code for useful snippets or simply thought about their problem. Based on his findings, Blikstein built three 'coding profiles'. Those matching the *Copy and paster* profile tended to pause from writing code to browse existing code bases for snippets, but did not pause to think, while the *Self-sufficient* profile exhibits the opposite behavior. The third profile, called *Mixed mode* took pauses for both reasons.

2.4 Time Management and Self-Regulation

Self-regulation in learning refers to the ability to observe and manage how and when one learns. Previous studies have linked student self-regulation and effort-regulation with academic success [4, 13, 31], and shown that self-regulation can be affected by the learning environment [2].

Similarly, good time-management skills — for example, not leaving everything to the last minute — have been connected with better academic performance and even quality of life for students [23].

One way of measuring time-management and self-regulation skills is by using the Motivated Strategies for Learning Questionnaire (MSLQ) [28]. MSLQ is a questionnaire designed

to determine student motivation and learning strategies [29], and it can be used to measure the student's effort regulation, time and study environment management, and metacognitive self-regulation.

MSLQ has been previously used to predict performance in the context of introductory programming courses. For example, Bergin et al. [3] found that student performance was correlated with both the time and study environment metric and the effort regulation metric of the MSLQ. Similarly, Watson et al. [38] found a weak and marginally significant correlation between student performance and the MSLQ effort regulation metric.

3. RESEARCH DESIGN

Based on the dual nature of the previous work presented above, it is clear that pauses can be both good and bad for learning and retention. In this research we specifically look at how students take pauses from writing code. We hypothesize that the length of the pause is a factor in whether a pause is harmful or not, and that students with better time and study environment management skills take less pauses of lengths correlated with lower exam scores.

By this work, we replicate and validate the previous findings by Leppänen et al [21], and extend it by conducting a more comprehensive statistical analysis on a data set with more information on the participants. We focus exclusively on the writing process by excluding expected pauses such as those that are encountered when a student runs her program, and explore how background variables such as gender, age, programming background, previous academic degrees, etc., as well as more complex student background variables such as self-regulation and time and study environmental management metrics — as measured by the MSLQ — are correlated with how students take pauses from their primary task of writing code, as well as how they space out their work over the week.

3.1 Research Questions

The research questions through which we investigate our hypothesis are as follows:

RQ1 How do students space out their work?

- RQ1.1 On how many days on average per week do students work on course assignments?
- RQ1.2 When working on course assignments, do students take short pauses and if so, what are the typical pause lengths?

RQ2 How does the students' tendency to pause influence the course outcomes?

RQ3 Do pausing or spacing behaviors correlate with student background variables?

- RQ3.1 To what extent do the students' background variables and answers to the Self-Regulation-specific questions from the MSLQ correlate with students spacing out their work over days?
- RQ3.2 If there are pause lengths that are negatively correlated with exam scores, how are the relative amounts of such pauses correlated with students' background variables and answers to the Self-Regulation-specific questions from the MSLQ?

With the first research question, we are interested in learning what kind of spacing and pausing behaviors, if any, exist in our context. With the second research question, we explore whether the spacing and pausing behaviors of students correlate with course outcomes. With the third research question, we examine through key MSLQ metrics whether the students' self-regulation or time-management skills correlate with spacing and pausing behavior.

3.2 Context & Data

The data for the study comes from two introductory programming courses organized at a European research first university. One of the authors of this article is the person responsible for the courses and has conducted the data gathering. One of the courses was held in the fall of 2014 and the other in the spring of 2015. Both courses were seven weeks long CS1-level courses that taught Object Oriented programming with the Java programming language. A total of 300 students enrolled in the courses.

While the courses were identical in content and near-identical in assignments, due to their timing within the academic year, their populations tend to be different: the majority of the fall course students were CS freshmen, while the spring course was mostly attended by students minoring in CS. Because of this, these populations should not be seen as a single cohort, but rather two somewhat similar, yet different, cohorts of students.

During the course, the students read a mixed online textbook that contains embedded course assignment prompts. Within each topic, the assignments moved from short, basic assignments to more general, larger assignments. When sufficiently complex, the assignments were split into multiple subgoals. The assignment prompts were separated from the assignment templates, so that the students were forced to read the prompt at least partially before they could even attempt to solve the problem. The assignments were completed in the NetBeans IDE that uses an automated assessment tool called TestMyCode [37].

The courses had 113 and 125 assignments respectively, with the second course also containing two weekly pair-programming assignments. The TestMyCode plugin allowed the students to check their solution for validity on their own computer before submitting it for automatic grading on our servers. There was no penalty for checking for correctness locally or submitting an incomplete or otherwise incorrect submission as multiple submissions were allowed with the highest grade being the final.

The students were awarded points based on the amount of successfully completed assignments. These points accounted for either 66 or 70% of the final grade. A single point (1-3% of final grade) was given for answering all questionnaires within the material. In the fall course, the rest of the points were given based on a pen-and-paper exam including both programming and essay questions. In the spring course, half of the exam points were given based on a pen-and-paper exam that had essay questions about programming concepts and half of the exam points were given based on a computerized programming exam with a time limit that had exercises similar to the weekly assignments.

Students were allowed to opt out of data collection and hence also from the study. Participants of the study provided data for the study in three ways. First, we have question-specific records of how the students fared in the final exam. Second, the online material included an embedded Motivated Strategies for Learning Questionnaire and

	Total	Age			Programming background			Gender	
	n	- 22	23 - 45	46 -	None	Some	Much	Male	Female
Course 1	156	94	62	0	91	33	32	106	50
Course 2	47	23	23	1	28	13	6	32	15

Table 1: Key participant demographics

a general background questionnaire. Thirdly, the IDE the participants used for programming assignments collected detailed snapshots of the participants' programming progress. These details include timestamps of every key press made within the course assignment projects.

In this paper, we only consider a subset of the enrolled students: of the 300 students, 82 did not attend an exam and/or chose not to answer the background questionnaire and a further 7 students declined to participate in the IDE-based data collection. Finally, 8 students attended both courses and were excluded to prevent issues with data from both courses mixing. This left us with $n = 203$ students who were used to examine the correlations between pauses and exam scores. A significant majority ($n = 156$) of these attended Course 1, leaving Course 2 with a significantly lower n. Key participant demographics for both courses are presented in Table 1.

Finally, a total of $n = 176$ participants answered the MSLQ. These participants were used as a data set to search for a correlation between student self-regulation and pausing behavior. No statistically significant differences in course results were observed between the participants who answered MSLQ and the participants who did not answer MSLQ.

4. METHODOLOGY AND RESULTS

Data received from the three sources (programming environment events with timestamps, questionnaires, exams) was combined into a single data set for the analysis.

We defined all intervals between two subsequent events that were longer than one second in duration as pauses. This definition was chosen based on keystroke-analysis studies, which suggest that the time between two subsequent key presses when typing subconsciously is typically less than 750 milliseconds [11, 22].

For RQ1, and RQ3.1, all the pause data was utilized. For RQ2, pauses that occurred after the student ran, tested locally or submitted his or her code were excluded. By excluding such pauses, we can focus on how pauses taken within the process of completing an assignment affect students' success. At the same time, pauses resulting from – for example – submitting an answer are still relevant to RQ1 and RQ3.1, since even just submitting a single exercise during a certain day is still work put towards the course.

All correlations detailed below are Pearson correlation coefficients, and when needed, Bonferroni corrections [12] have been performed to counter the multiple testing problem.

4.1 How Do Students Space Out Their Work?

We examined on how many days each week the students worked on the course assignments. During the fall course (below, Course 1), students worked on assignments on average 2.8 days each week (SD=0.91). During the spring course (below, Course 2), the mean was 3.3 (SD=0.87). The difference is statistically significant ($p < .01$). No statistically significant differences between the amount of days that the top and bottom students use for the course per week were observed when accounting for the programming background. The values were also calculated for the top 25%, middle 50%, and bottom 25% of students by exam score, shown in Table 2.

	Group	Mean	StDev
Course 1	All	2.794	0.914
	Top 25%	2.796	0.997
	Middle 50%	2.739	0.891
	Bottom 25%	2.902	0.862
Course 2	All	3.299	0.867
	Top 25%	2.883	0.859
	Middle 50%	3.191	0.783
	Bottom 25%	3.888	0.711

Table 2: Mean days per week students worked on exercises, grouped by exam score.

Based on the above, our answer to **RQ 1.1**, '*On how many days on average per week do students work on course assignments?*' is: Students work on the course assignments on 3 days per week, on average. The exact values vary by course iteration, and likely differ between contexts as well.

While working on course assignments, students take pauses of a wide range of lengths. Frequencies of pauses of certain lengths are presented as histograms in Figures 1 and 2. Analysis of the histograms shows that student pauses come in two basic forms: pauses less than 6 hours in length and pauses of one or more days. For the short pauses, the shorter the pause, the more frequent it is, but other than that, no clear pattern emerges. Longer pauses show clear patterns in frequency; pauses tend to be in multiples of days.

Our answer to **RQ 1.2**, '*When working on course assignments, do students take short pauses and if so, what are the typical pause lengths?*', is as follows: Students tend to take short pauses while working on course assignments. There does not seem to be a typical pause length within pauses of less than 6 hours, other than that shorter pauses are more frequent.

4.2 How does Students' Tendency to Pause Influence the Course Outcomes

As the pause lengths were reported very accurately, with millisecond precision, it was necessary to group them into larger ranges to allow for a meaningful analysis of correlations between relative counts of pauses of certain lengths and the exam scores. We divided the pauses into varying, partially overlapping logarithmic-like ranges of pause lengths such as pauses of 1 to 5 seconds and pauses of 1 to 10 seconds. For each pause length range, we calculated which percentage of each student's pauses fell within that range. By searching for a correlation between these *relative* pause counts and exam scores, we normalized for the fact that different students spent different amounts of time working on

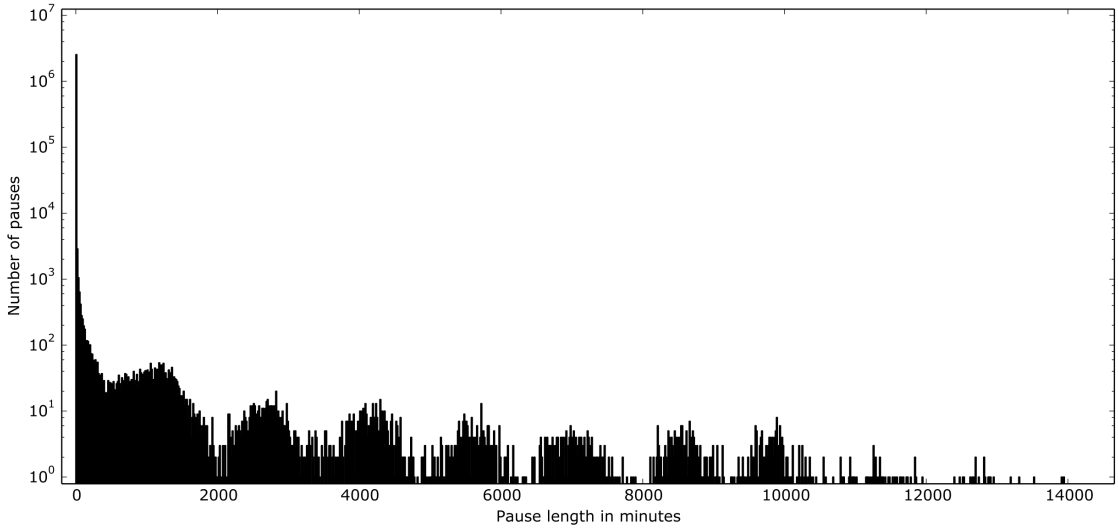

Figure 1: Histogram of frequencies of pauses, in one minute bins. Note the logarithmic y-axis.

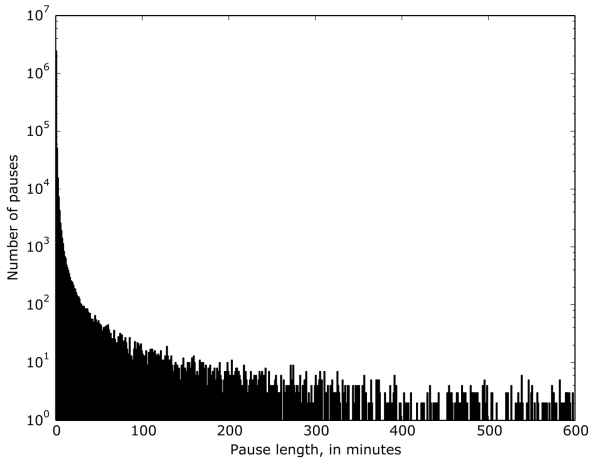

Figure 2: Histogram of frequencies of pauses shorter than 10 hours, in single minute bins. Note the logarithmic y-axis.

the exercises. The results were corrected for multiple tests using the Bonferroni correction. These results and all the used pause length ranges for Course 1 are shown in Table 3.

To ensure the reliability of our findings, we further studied these results using the data from Course 2. However, possibly due to the lower participant count in the second course, most correlations for that course become statistically non-significant when Bonferroni corrected. Only the 1 to 2 second, 1 to 5 second, 1 to 10 second, 60 to 70 second, 100 to 110 second, 110 to 120 second, 1 to 2 minute, 2 to 3 minute and 3 to 4 minute pauses were statistically significantly correlated with exam scores in both courses after the Bonferroni correction was applied.

At the same time, the non-corrected values for Course 2 agreed with the Bonferroni corrected Course 1 values except for the 5 to 6 minute, 10 to 11 minute, 16 to 17 minute and the 9 to 10 second pauses which had $p > 0.05$ even before a correction was applied. Some of the more interesting correlations with Bonferroni corrections are shown in Table 4.

Our answer to **RQ 2**, '*How does the students' tendency to pause influence the course outcomes?*', is therefore twofold. First, there is a clear correlation between the students' pauses and the students' success in the course final exam. A high relative amount of very short pauses of 1 to 2 seconds is positively correlated with the exam score. At the same time, pauses between 1 and 4 minutes are negatively correlated with the exam score.

Second, the tests on the larger course show a statistically significant negative correlation between pauses of lengths 10 to 60 seconds and the course exam score, but this effect did not persist in the smaller course. A similar phenomenon was observed with the 4 to 5 and 5 to 10 minute pauses. See Section 5.2 for a possible explanation for this phenomenon as well as discussion on how this result should be interpreted.

4.3 Self-Regulation and other Background Variables as Predictors of Pausing Behavior

We searched for correlation between the following student background variables and the number of days studied per week for both courses: handedness; level of highest attained degree; general programming background; Java programming background; year of birth; gender; year of studies; exam score; MSLQ metacognitive self-regulation metric; MSLQ time and study environmental management metric; and MSLQ effort regulation metric. These results are detailed in Table 5.

After a Bonferroni correction was applied, only general programming background for Course 1 ($r = -0.45$, $p < 0.01$) and Java programming background for Course 1 ($r = -0.33$, $p < 0.01$) displayed a statistically significant correlation. Java programming background was similarly correlated for Course 2 ($r = -0.39$, $p = 0.01$), but this effect was no longer statistically significant after a Bonferroni correction was applied. Looking at both courses, the (statistically

Pause lengths	r	$p_{\text{bonferroni}}$
1 - 2 s	0.44	$< 0.01*$
2 - 3 s	0.23	0.30
3 - 4 s	0.01	> 0.99
4 - 5 s	−0.06	> 0.99
5 - 6 s	−0.13	> 0.99
6 - 7 s	−0.23	0.26
7 - 8 s	−0.32	$< 0.01*$
8 - 9 s	−0.20	0.95
9 - 10 s	−0.31	$< 0.01*$
11 - 12 s	−0.17	> 0.99
12 - 13 s	−0.35	$< 0.01*$
13 - 14 s	−0.39	$< 0.01*$
14 - 15 s	−0.36	$< 0.01*$
15 - 16 s	−0.21	0.52
16 - 17 s	−0.24	0.16
17 - 18 s	−0.42	$< 0.01*$
18 - 19 s	−0.14	> 0.99
19 - 20 s	−0.35	$< 0.01*$
1 - 5 s	0.49	$< 0.01*$
5 - 10 s	−0.30	$0.01*$
10 - 15 s	−0.44	$< 0.01*$
15 - 20 s	−0.33	$< 0.01*$
20 - 25 s	−0.42	$< 0.01*$
25 - 30 s	−0.41	$< 0.01*$
30 - 35 s	−0.42	$< 0.01*$
35 - 40 s	−0.35	$< 0.01*$
40 - 45 s	−0.46	$< 0.01*$
45 - 50 s	−0.16	> 0.99
50 - 55 s	−0.45	$< 0.01*$
55 - 60 s	−0.43	$< 0.01*$
1 - 10 s	0.52	$< 0.01*$
10 - 20 s	−0.41	$< 0.01*$
20 - 30 s	−0.47	$< 0.01*$
30 - 40 s	−0.44	$< 0.01*$
40 - 50 s	−0.34	$< 0.01*$
50 - 60 s	−0.49	$< 0.01*$
60 - 70 s	−0.40	$< 0.01*$
70 - 80 s	−0.42	$< 0.01*$
80 - 90 s	−0.46	$< 0.01*$
90 - 100 s	−0.46	$< 0.01*$
100 - 110 s	−0.46	$< 0.01*$
110 - 120 s	−0.37	$< 0.01*$
1 - 2 min	−0.48	$< 0.01*$
2 - 3 min	−0.41	$< 0.01*$
3 - 4 min	−0.40	$< 0.01*$
4 - 5 min	−0.32	$< 0.01*$
5 - 6 min	−0.30	$< 0.01*$
6 - 7 min	−0.29	$0.02*$
7 - 8 min	−0.25	0.11
8 - 9 min	−0.20	> 0.99
9 - 10 min	−0.17	> 0.99
10 - 11 min	−0.30	$0.01*$
11 - 12 min	−0.14	> 0.99
12 - 13 min	−0.12	> 0.99
13 - 14 min	−0.12	> 0.99
14 - 15 min	−0.06	> 0.99
15 - 16 min	−0.18	> 0.99
16 - 17 min	−0.29	$0.01*$
17 - 18 min	−0.06	> 0.99
18 - 19 min	−0.07	> 0.99
19 - 20 min	−0.10	> 0.99
5 - 10 min	−0.31	$< 0.01*$
10 - 15 min	−0.21	0.72
15 - 20 min	−0.24	0.23
20 - 25 min	−0.15	> 0.99
25 - 30 min	−0.03	> 0.99
30 - 35 min	−0.13	> 0.99
35 - 40 min	−0.19	> 0.99
40 - 45 min	−0.19	> 0.99
1 - 2 h	−0.12	> 0.99
2 - 6 h	−0.03	> 0.99
6 - 12 h	−0.15	> 0.99
12 - 24 h	−0.18	> 0.99
24+ h	−0.02	> 0.99

Table 3: Correlations between the relative amounts of pauses of certain lengths and student exam scores in Course 1. P-values are Bonferroni corrected. An asterisk (*) signifies a statistically significant result at $p < 0.05$.

non-significantly correlated) MSLQ variables the r values ranged from −0.16 to 0.21.

Therefore our answer to **RQ 3.1** is: There seems to exist a weak statistically significant correlation between average number of days per week the student worked on the exercises and the student's previous programming background.

Based on the pause length correlations detailed above, we defined as *medium-length pauses* all pauses between 10 seconds and 5 minutes in length and searched for correlations between the same background variables as above, and the percentage of each student's pauses that were medium-length. This was done separately for both courses.

After a Bonferroni correction was applied, only the Java programming background variable for Course 1 was statistically significantly correlated with the percentage of medium-length pauses taken ($r = -0.24$, $p < 0.01$). No other correlation was of statistical significance. The only MSLQ variables to show statistical significance before the Bonferroni correction was applied were Course 1's Time and Study Environmental Management metric ($r = 0.19$, $p = 0.03$) and Course 2's Metacognitive Self-Regulation metric ($r = -0.37$ $p = 0.02$). Both of these MSLQ variables were statistically non-significant after Bonferroni corrected.

For Course 1, the gender variable also first appeared to be correlated with percentage of medium-length pauses, but this effect disappeared when previous programming background was accounted for. That is, when looking at students of similar programming backgrounds only, gender was not statistically significantly correlated with the percentage of pauses of lengths correlated with low exam scores.

Our answer to **RQ 3.2** is therefore as follows: Only student programming background seems to be statistically significantly correlated with the percentage of pauses correlated with lower exam scores. MSLQ's measures of self-regulation or time and study management skills were counter-intuitively not correlated with the percentage of such pauses.

5. DISCUSSION

5.1 Spacing Work Over Multiple Days

Students tend to space out their work over multiple working days. In our data sets, the students work on a course on around three days every week.

Somewhat surprisingly, students with higher exam scores averaged less working days per week, but this could not be explained by previous programming experience. This was unexpected, since previous research indicates that spacing out work over time is beneficial [7, 8, 10, 14, 36]. As much of the previous research is focused on recall, it might not be immediately applicable to the context of learning a task such as programming.

It is of note that, although the course material was near-identical between Courses 1 and 2, there was a statistically significant difference in the average days spent on assignments per week. The students on Course 1 used around half a day less on assignments compared to Course 2's students. We hypothesize that the difference is at least partially caused by the marginally increased workload for Course 2, which had a couple more assignments as well as pair programming exercises.

Another possible explanation is the difference in the grading scheme. Course 1 was graded from 0 to 5, while Course 2 used a pass / fail grading scheme. To get a pass in the new

	Course 1			Course 2			
	r	p_b	p	r	p_b	p	Both
1 - 2 s	0.44	< 0.01	< 0.01	0.64	< 0.01	< 0.01	True
2 - 3 s	0.23	0.30	< 0.01	0.19	> 0.99	0.20	False
3 - 4 s	0.01	> 0.99	0.87	-0.13	> 0.99	0.39	False
4 - 5 s	-0.06	> 0.99	0.46	-0.45	0.12	< 0.01	False
1 - 5 s	0.49	< 0.01	< 0.01	0.56	< 0.01	< 0.01	True
5 - 10 s	-0.30	0.01	< 0.01	-0.42	0.22	< 0.01	False
1 - 10 s	0.52	< 0.01	< 0.01	0.54	0.01	< 0.01	True
1 - 2 min	-0.48	< 0.01	< 0.01	-0.52	0.01	< 0.01	True
2 - 3 min	-0.41	< 0.01	< 0.01	-0.55	< 0.01	< 0.01	True
3 - 4 min	-0.40	< 0.01	< 0.01	-0.60	< 0.01	< 0.01	True
4 - 5 min	-0.32	< 0.01	< 0.01	-0.46	0.08	< 0.01	False
5 - 10 min	-0.31	< 0.01	< 0.01	-0.33	> 0.99	0.02	False
10 - 15 min	-0.21	0.72	0.01	-0.26	> 0.99	0.08	False

Table 4: Some of the more interesting correlations between the relative amounts of pauses of certain lengths and exam scores. p is the raw p-value, while p_b is the Bonferroni corrected p-value. The column *Both* specifies those correlations that were statistically significant *using a very strict Bonferroni correction* also on the second course with a small n. See Section 5.3 for details on the interpretation of the analysis using Course 2.

grading system, the students had to complete an amount of work equal to around a grade 3 in the old system. This means that the students who only want to pass the course with minimal effort have to complete more assignments and do more work in order to pass the course, creating what is essentially a floor effect, which could explain the increased average days spent.

Thirdly, Course 1 had more computer science majors, while Course 2 had a lot of students who are minoring in CS. During Course 2, the students who were in the top 25% in exam points used statistically significantly ($p < 0.01$) less days on assignments compared to the bottom 25%. During Course 1, there was no statistically significant difference between the top 25% and the bottom 25% groups.

Overall, it seems unlikely that such a difference between the courses could be caused by small differences in the amount of coursework, so the difference in course demographics and grading policy seem like the most likely explanation for this phenomenon.

5.2 Short Pauses and Course Outcomes

Clear negative correlations were found between the relative amounts of short pauses and exam scores. This effect was present despite the data being unable to distinguish between off-task behavior such as checking social media from secondary on-task behavior such as checking the course material. The effect seems to be in line with previous research conducted on the effects of task switching [26] and multi-tasking [18, 19, 32, 34, 39]: If the students are off-task during the pauses, they would suffer from the detrimental effects of task switching. For very short off-task sessions, this could even be considered multi-tasking, which previous research indicates would cause further detrimental effect. At the same time, if the pause is short enough, the positive benefits of the spacing effect [10] would not have time to take place.

Similarly, if the students are on-task but engaging in a secondary behavior such as reading the course material, they would still suffer from the effects of task-switching when their attention moves from the assignment to the material. Furthermore, having to constantly jump between the material and the assignment could be interpreted as meaning that the cognitive load of the task is great: the student is unable to concurrently hold all the pieces of the solution in working memory and has to jump to and recheck facts from the material even if they presumably have read the material beforehand. Since low cognitive loads have been linked to better recall [30], it seems reasonable to expect that a situation like this would have a detrimental effect on the student's learning.

Reflecting programming into the model of writing presented by Flower & Hayes [16], on-task pauses from writing code could be construed as instances of the student engaging in the processes of *Planning* what they should accomplish next and *Reviewing* what they have written rather than in the *Translating* process where abstract ideas are translated into syntactically correct text (in our context, code) and written down.

Such behavior could be indicative of a programming process in which the student tends to write code that he or she then very soon has to refactor. One work flow that could cause such a pattern is as follows: the student first reads a very minimal amount of the exercise prompt until he or she has found a minimal criterion his or her program must satisfy, for example that it must have a function with a certain signature. The student then writes a program that fulfills just that criterion and then repeats the process. Such a work flow would naturally lead to a lack of a higher order plan, which the student would formulate if he or she first read the whole prompt. Similarly, such students would spend much time refactoring previously written code, since it is unlikely that all code written in such an ad-hoc fashion would be forward compatible with the constraints and requirements set by the latter parts of the assignment prompt. It would be natural to expect that such a student would struggle more, when compared to a student who has a flexible higher level plan.

While previous research has shown that subgoals are beneficial in many ways [24], it also seems that for the writing process in general, flexible higher level plans are also important [15]. A student that is not in the habit of creating such plans would naturally be less able to abstract and model programming problems. Similarly, this kind of

	Course 1			Course 2		
	r	p	p_b	r	p	p_b
Handedness	−0.02	0.78	> 0.99	−0.01	0.97	> 0.99
Educational background	−0.01	0.91	> 0.99	−0.15	0.33	> 0.99
Programming background	−0.35	< 0.01	< 0.01	−0.10	0.52	> 0.99
Java background	−0.33	< 0.01	< 0.01	−0.39	0.01	0.29
Year of Birth	0.06	0.44	> 0.99	−0.04	0.80	> 0.99
Gender	−0.22	0.01	0.26	−0.32	0.05	> 0.99
Year of studies	0.078	0.37	> 0.99	0.04	0.78	> 0.99
Exam score	−0.11	0.21	> 0.99	−0.34	0.03	0.76
MSLQ: Metacognitive Self-Regulation	0.12	0.17	> 0.99	−0.14	0.41	> 0.99
MSLQ: Time and Study Environment Management	0.21	0.02	0.39	0.10	0.55	> 0.99
MSLQ: Effort Regulation	0.01	0.87	> 0.99	−0.16	0.33	> 0.99

Table 5: Correlations of student background variables and MSLQ metrics with number of days worked per week. Columns labeled p_b show the Bonferroni corrected values.

behavior is disadvantageous in an exam where planning is extremely beneficial: a pen-and-paper exam makes refactoring very hard and time consuming, meaning that students who are unable to formulate a plan before they start coding are at a disadvantage. Many programming exam questions tend to take the form of an algorithmic problem, wherein first figuring out the solution in, for example, pseudocode is very beneficial.

It seems logical to assume that the *downslide* effect detailed by Collins & Gentner [9] applies to programming as well: the writer — in this case the programmer — tends to 'slide down' from higher to lower level task processing, losing the big picture. In the context of programming, this could manifest as a student spending all his mental resources on fixing syntax errors at the cost of leaving logical errors unfixed or unnoticed. When considered with the research on the cost of task-switching, one expects that students who tend to downslide would naturally incur a somewhat large task switching penalty when they eventually need to 'rerise' from their local frame of reference to the higher frame of reference of the assignment in general. Such penalties could manifest as the short pauses we have observed.

Based on these observations, it seems reasonable to expect that the correlation between relative amounts of short pauses and exam scores is at least partially independent of the content of the pauses: both off-task and on-task pauses can either harm the studying process or are indicators of other harmful habits and effects.

Based on our findings taken in context with the literature on the cognitive process of writing, students should be instructed to fully read the problem description before starting to work on it. This would most likely encourage the students to develop a more flexible high-level plan that is constantly refined, similar to those that have been linked to good general writing ability [15].

5.3 Correlations of Pauses Lengths And Exam Scores

The question of what pause lengths are correlated with lower exam scores does not have a completely clear answer based on our data. The data clearly shows that the percentage of extremely short pauses of 1 - 2 seconds are positively correlated with a high exam score, and that the percentage of pauses between 1 to 4 minutes in length are negatively correlated with exam scores. These results stand for both courses and remain statistically significant even after the Bonferroni correction is applied.

We have come up with two possible interpretations for the positive correlation between the relative number of 1 to 2 second pauses and exam scores. The first interpretation is that the relatively high amount of 1 to 2 second pauses has no effect by itself, but that rather the high amount represents a lack of longer pauses. An alternative explanation for the effect is that these extremely short pauses are instances of the student spending an extremely short while thinking about his or her work. This hypothesis is in line with previous research showing the benefits of reflection and debriefing [6, 25, 27], even if on a much shorter time scale.

It is of note that the test setup is almost a pathological case for a Bonferroni correction: because of the overlapping windows, the tests are highly correlated. Similarly, the small n of Course 2 makes it extremely hard to reach statistical significance with the Course 2 data. Because of this, we feel that a more truthful understanding of the results is achieved by looking at the Bonferroni corrected values for Course 1 together with the non-corrected values for Course 2. Via such an analysis, the following picture emerges: The proportion of 1 to 2 second pauses is positively correlated with exam scores and the proportion of 10 second to 5 minute pauses is negatively correlated with exam scores. The proportion of 5 to 10 minutes also appears to be negatively correlated with exam scores, albeit with ($p = 0.02$) in Course 2 with the smaller n.

5.4 Correlations Between Background Variables and Pausing Behavior

The results indicate that of the observed background variables, only programming background is correlated with the spacing the students do and the relative amounts of pauses of lengths correlated with lower exam scores the students take. The fact that programming background is a factor seems very intuitive, but at the same time it being the *only* statistically significant variable is highly surprising.

Our initial expectation was that the observed MSLQ variables would be correlated with the days worked (as students with better time-management skills would space out their learning more) and with the amount of pauses correlated with lower exam scores (the same students would be less likely to take off-task pauses). The lack of such a correlations was surprising as connections between programming

performance and the observed MSLQ metrics have been previously observed [4].

In our view, a likely explanation for this phenomenon is that MSLQ only measures what the students *think* they are doing, rather than what they actually do, and that students are unaware of the harmful effects of multitasking and the positive effects of spacing out. It would therefore seem like a good idea to explicitly inform the students of these effects.

5.5 Limitations

The research described here has multiple limitations. First, we used the Motivated Strategies for Learning Questionnaire (MSLQ) to determine key self-regulation-related metrics of the students. It is however not completely clear whether MSLQ measures what the students do or what they *think* they do: MSLQ in general tends to capture larger tendencies and does not take into account the subconscious use of social media or other such factors that could be important in our context.

Second, we used data from a smaller second programming course to increase the external validity of our findings. Whilst both courses were on introductory programming, their grading schemes and student populations (regarding major subjects) differed from each other. Some of the interpretations where we expected the results to hold in both courses may have been too strict due to the differences in the population size; an additional influence was likely caused by the strictness of the Bonferroni correction for our family of tests.

Third, all correlations reported here are Pearson's correlation coefficients. Due to this, our tests would not have identified any possible non-linear correlations. However, such correlations were not evident in the data based on a visual analysis.

Finally, an internal validity issue is that we do not actually know what the students did during a pause. Despite the observed correlations, different kinds of pauses are likely to have different kinds of effects on learning – for example, there would probably be a difference between a pause where the student looks at the course material and a pause where the student is interrupted by a third party.

6. CONCLUSIONS AND FUTURE WORK

In this work, we studied students' tendencies to pause and pace their work, and analyzed the connection of such behavior with course outcomes and self-regulation metrics.

Based on data from two separate programming courses, we observed that while working on weekly assignments, students tend to space out their work over multiple days. Surprisingly, the self-regulation metrics did not correlate with the spacing behavior.

We observed that the relative amounts of short pauses of 10 seconds to 4 minutes have a negative correlation with exam scores. At the same time, longer pauses are not significantly correlated with exam scores when validated using data from the smaller course. The only background variables with statistically significant correlations with the relative amounts of pauses in the ranges with negative correlations with exam scores were related to programming background. The observed MSLQ variables were not statistically significantly correlated with the amounts of such pauses after a Bonferroni correction was applied.

Our results suggest that it would be possible to identify students with spacing and pausing habits correlated to lower exam scores. Since our data is coming from an integrated development environment, it should also be possible to construct a tool that monitors the students' work flow by for example monitoring key presses, and would then automatically notify the student of pausing habits correlated to lower exam scores. Alternatively, such a tool could work retrospectively by sending the students periodic emails detailing how they should modify their study habits to optimize the time they spend on assignments. It would also be useful to integrate such 'good working habits' meters in possible student dashboards. Such tools could also be tied to gamification systems, with students gaining 'points' or 'badges' for good study habits.

Based on our findings, we propose the following avenues of future work:

1. Identify whether other background variables explain how students space out their work.
2. Determine whether there are typical program states or error states that lead to pauses.
3. Investigate how nearness to a deadline affects how students take pauses while working, and the effects of the pauses on course outcomes.
4. Explore how different reasons for pausing the primary task affect the results presented here: are pauses spent reading the course material equal in effect to pauses spent on a social media?
5. Seek to determine whether it is possible to eliminate the pauses correlated with lower exam scores from the students' work flow.

7. REFERENCES

[1] J. Andrade. What does doodling do? *Applied Cognitive Psychology*, 24(1):100–106, 2010.

[2] T. Auvinen. Educational technologies for supporting self-regulated learning in online learning environments. 2015.

[3] S. Bergin and R. Reilly. The influence of motivation and comfort-level on learning to program. 2005.

[4] S. Bergin, R. Reilly, and D. Traynor. Examining the role of self-regulated learning on introductory programming performance. In *Proc. of the first international workshop on Computing education research*, pages 81–86. ACM, 2005.

[5] P. Blikstein. Using learning analytics to assess students' behavior in open-ended programming tasks. In *Proc. of the 1st international conference on learning analytics and knowledge*, pages 110–116. ACM, 2011.

[6] D. Boud, R. Keogh, and D. Walker. *Reflection: Turning experience into learning*. Routledge, 2013.

[7] N. J. Cepeda, H. Pashler, E. Vul, J. T. Wixted, and D. Rohrer. Distributed practice in verbal recall tasks: A review and quantitative synthesis. *Psychological bulletin*, 132(3):354, 2006.

[8] N. J. Cepeda, E. Vul, D. Rohrer, J. T. Wixted, and H. Pashler. Spacing effects in learning a temporal ridgeline of optimal retention. *Psychological science*, 19(11):1095–1102, 2008.

[9] A. Collins and D. Gentner. A framework for a cognitive theory of writing. *Cognitive processes in writing*, pages 51–72, 1980.

[10] F. N. Dempster. The spacing effect: A case study in the failure to apply the results of psychological research. *American Psychologist*, 43(8):627, 1988.

[11] P. Dowland and S. Furnell. A long-term trial of keystroke profiling using digraph, trigraph and keyword latencies. In Y. Deswarte, F. Cuppens, S. Jajodia, and L. Wang, editors, *Security and Protection in Information Processing Systems*, volume 147 of *IFIP - The International Federation for Information Processing*, pages 275–289. Springer, 2004.

[12] O. J. Dunn. Multiple comparisons among means. *Journal of the American Statistical Association*, 56(293):52–64, 1961.

[13] S. H. Edwards, J. Snyder, M. A. Pérez-Quiñones, A. Allevato, D. Kim, and B. Tretola. Comparing effective and ineffective behaviors of student programmers. In *Proc. of the Fifth International Workshop on Computing Education Research Workshop*, ICER '09, pages 3–14, New York, NY, USA, 2009. ACM.

[14] J. M. Ellenbogen, P. T. Hu, J. D. Payne, D. Titone, and M. P. Walker. Human relational memory requires time and sleep. *Proc. of the National Academy of Sciences*, 104(18):7723–7728, 2007.

[15] L. Flower and J. R. Hayes. The dynamics of composing: Making plans and juggling constraints. *Cognitive processes in writing*, 31:50, 1980.

[16] L. Flower and J. R. Hayes. A cognitive process theory of writing. *College composition and communication*, 32(4):365–387, 1981.

[17] J. Hattie and G. C. Yates. *Visible learning and the science of how we learn*. Routledge, 2013.

[18] H. Hembrooke and G. Gay. The laptop and the lecture: The effects of multitasking in learning environments. *Journal of computing in higher education*, 15(1):46–64, 2003.

[19] R. Junco and S. R. Cotten. No a 4 u: The relationship between multitasking and academic performance. *Computers & Education*, 59(2):505–514, 2012.

[20] M. Leinikka, A. Vihavainen, J. Lukander, and S. Pakarinen. Cognitive flexibility and programming performance. In *Proc. of the Psychology of Programming Interest Group Annual Conference 2014*, PPIG'25, pages 1–11, Brighton, UK, 2014.

[21] L. Leppänen, J. Leinonen, and A. Vihavainen. Short pauses while studying considered harmful. In *EDULEARN 2016*, pages 1900–1904. IATED, 2016.

[22] K. Longi, J. Leinonen, H. Nygren, J. Salmi, A. Klami, and A. Vihavainen. Identification of programmers from typing patterns. In *Proc. of the 15th Koli Calling Conference on Computing Education Research*, Koli Calling '15, pages 60–67, New York, NY, USA, 2015. ACM.

[23] T. H. Macan, C. Shahani, R. L. Dipboye, and A. P. Phillips. College students' time management: Correlations with academic performance and stress. *Journal of educational psychology*, 82(4):760, 1990.

[24] L. E. Margulieux, M. Guzdial, and R. Catrambone. Subgoal-labeled instructional material improves performance and transfer in learning to develop mobile applications. In *Proc. of the ninth annual international conference on International computing education research*, pages 71–78. ACM, 2012.

[25] J. Mezirow et al. How critical reflection triggers transformative learning. *Fostering critical reflection in adulthood*, pages 1–20, 1990.

[26] S. Monsell. Task switching. *Trends in cognitive sciences*, 7(3):134–140, 2003.

[27] M. Pearson and D. Smith. Debriefing in experience-based learning. *Reflection: Turning experience into learning*, pages 69–84, 1985.

[28] P. R. Pintrich et al. A manual for the use of the motivated strategies for learning questionnaire (mslq). 1991.

[29] P. R. Pintrich, D. A. Smith, T. García, and W. J. McKeachie. Reliability and predictive validity of the motivated strategies for learning questionnaire (mslq). *Educational and psychological measurement*, 53(3):801–813, 1993.

[30] E. Pollock, P. Chandler, and J. Sweller. Assimilating complex information. *Learning and instruction*, 12(1):61–86, 2002.

[31] M. Richardson, C. Abraham, and R. Bond. Psychological correlates of university students' academic performance: a systematic review and meta-analysis. *Psychological bulletin*, 138(2):353, 2012.

[32] L. D. Rosen, A. F. Lim, L. M. Carrier, and N. A. Cheever. An empirical examination of the educational impact of text message-induced task switching in the classroom: Educational implications and strategies to enhance learning. *Psicología educativa*, 17(2):163–177, 2011.

[33] K. L. Ruhl, C. A. Hughes, and P. J. Schloss. Using the pause procedure to enhance lecture recall. *Teacher Education and Special Education: The Journal of the Teacher Education Division of the Council for Exceptional Children*, 10(1):14–18, 1987.

[34] F. Sana, T. Weston, and N. J. Cepeda. Laptop multitasking hinders classroom learning for both users and nearby peers. *Computers & Education*, 62:24–31, 2013.

[35] S. M. Smith and S. E. Blankenship. Incubation and the persistence of fixation in problem solving. *The American journal of psychology*, pages 61–87, 1991.

[36] R. Stickgold. Sleep-dependent memory consolidation. *Nature*, 437(7063):1272–1278, 2005.

[37] A. Vihavainen, T. Vikberg, M. Luukkainen, and M. Pärtel. Scaffolding students' learning using test my code. In *Proc. of the 18th ACM conference on Innovation and technology in computer science education*, pages 117–122. ACM, 2013.

[38] C. Watson, F. W. Li, and J. L. Godwin. No tests required: comparing traditional and dynamic predictors of programming success. In *Proc. of the 45th ACM technical symposium on Computer science education*, pages 469–474. ACM, 2014.

[39] E. Wood, L. Zivcakova, P. Gentile, K. Archer, D. De Pasquale, and A. Nosko. Examining the impact of off-task multi-tasking with technology on real-time classroom learning. *Computers & Education*, 58(1):365–374, 2012.