Pesonen, Hannu; Hellas, Arto

Weekly Fluctuations in Motivation in Introductory Programming

Published in:
UKICER 2022 - Proceedings of the 2022 Conference on United Kingdom and Ireland Computing Education Research

DOI:
10.1145/3555009.3555012

Published: 01/09/2022

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Pesonen, H., & Hellas, A. (2022). Weekly Fluctuations in Motivation in Introductory Programming. In K. Quille, J. Maguire, & B. A. Becker (Eds.), *UKICER 2022 - Proceedings of the 2022 Conference on United Kingdom and Ireland Computing Education Research* Article 2 ACM. https://doi.org/10.1145/3555009.3555012

Weekly Fluctuations in Motivation in Introductory Programming

Hannu Pesonen
hannu.pesonen@aalto.fi
Aalto University
Espoo, Finland

Arto Hellas
arto.hellas@aalto.fi
Aalto University
Espoo, Finland

ABSTRACT

In this study, we look into fluctuations in types of motivation in an introductory programming course. Using weekly surveys to measure intrinsic motivation through interest and enjoyment, willingness to learn, interest in context, and time spent on the programming course, we study the evolution of students' motivation throughout the course. We identify subgroups based on the collected data, and study to what extent the early course motivation and time spent on the course can be used to predict course retention. We observe substantial fluctuations in motivation and time spent over the course rounds, and further identify three subgroups of students in the data with considerable variation in motivation and time spent. At the same time, we observe that despite marked differences in motivation and the time that students spend on the course, these differences, early on in the course, do not seem to help in predicting course dropouts.

CCS CONCEPTS

• **Social and professional topics** → **Computing education**; **CS1**.

KEYWORDS

motivation, CS1, time spent, weekly surveys

ACM Reference Format:

Hannu Pesonen and Arto Hellas. 2022. Weekly Fluctuations in Motivation in Introductory Programming. In *The United Kingdom and Ireland Computing Education Research (UKICER) Conference (UKICER2022), September 1–2, 2022, Dublin, Ireland*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3555009.3555012

1 INTRODUCTION

Across the world, computer science students take an introductory programming course (CS1) among the first courses in their degree (e.g. [9]). The contents and the way the course is taught often depend on the context and the university. Learning and teaching programming has been an active focus area in computing education research (CER) for a long time [21, 29]. One of the reasons why a CS1 course is often in focus is the observation that a noticeable proportion of students attending the course fail to pass [1, 2, 37, 44].

Understanding why some fail an introductory programming course can lead to insights about the students and the course [15, 26], as well as provide information for the teachers on possible ways to alleviate the issue. There has been discussion on whether the expectation level in introductory programming courses is on par with students' actual capabilities or indeed exceeds that [20]. At any rate, the objective should be to provide students with enough challenge, taking into account that some students have previous experience on programming from e.g. working or playing with computers at home [22]. The diverse background of students ought to be viewed as an asset for teachers of introductory programming courses, helping them design achievable course goals.

In this study, we view an introductory programming course from an angle whereby we look into weekly fluctuations in three types of motivation. This is achieved through repeated surveys of students' experiences in the course. Our goal is to complement the more traditional cognitive standpoint in CER with what might be dubbed a noncognitive side of learning programming: emotions such as enjoyment, interest or boredom elicited by learning experiences in programming; for a review of motivation vs cognitive research in CER, c.f. [18]. Our motivation for the work stems from the observation that each round in a course can bring about pivotal learning experiences that may increase or decrease students' motivation even to the extent that it impacts willingness to stay in a course. The research questions for our study are as follows.

RQ1 How are the types of motivation and time spent on the course related to each other?
RQ2 How do the types of motivation and time spent on the course change across the course weeks?
RQ3 What sorts of subgroups of motivation and time spent on the course are there in the data?
RQ4 To what extent do the types of motivation and the time spent on the first round of the course predict activity on the last mandatory week?

2 BACKGROUND

2.1 Motivation and Emotions

According to self-determination theory (SDT) [32], our behavior can crucially be driven by two different motivators – extrinsic and intrinsic. The extrinsic motivators can be subdivided according to four subcategories of regulatory styles: external regulation ("Something about your external situation forced you to do it"), introjected regulation ("You made yourself do it, to avoid anxiety or guilt"), identified regulation ("Interesting or not, you felt that it expressed your true values") and integrated regulation ("You knew that what you did was in line with what you are like."). Intrinsic motivation, on the other hand, is expressed by feelings of enthusiasm ("You did it purely for the interest and enjoyment in doing it"); a positive, direct relationship with the focus of attention, under intrinsic regulation (e.g. [28, 30]).

UKICER2022, September 1–2, 2022, Dublin, Ireland

ACM ISBN 978-1-4503-9742-1/22/09.
https://doi.org/10.1145/3555009.3555012

While Ryan and Deci [30] argue that intrinsic motivation is essentially an innate capability, they stress that it can be crucially enhanced or undermined by the social environments [31], e.g. rewards and supports that students encounter. In pedagogical terms, according to [31], teachers can increase students' intrinsic motivation by nurturing their basic psychological needs (autonomy, competence and relatedness) by, e.g., offering them enough challenge, support and cooperation.

As emotions are inherently a changing part of the human experience, it has been logical to look into them with a more tight comb than, e.g., once in a semester. Verhagen et al. [40] call for the need to explore the subjects' "natural flow of life". To meet this demand, experience sampling methods (ESMs) have been devised, also called, e.g., intensive longitudinal assessment [40], or diary methods/studies (For an introduction, cf. [3], and a more extensive analysis, see [11]). In diary methods, data collection can be many times during a course, several times a week, or in the case of ESM, even multiple times a day. As Verhagen et al. [40] point out, ESM and like methods reduce memory errors and yield data that is sensitive to change. Diary studies are designed such that they are effortless for respondents to take part in, e.g. through the use of Likert scale items, which further increases their ecological validity [11].

Hektner et al. [11] point out that an activity usually reported as being intrinsically rewarding can nonetheless at other times be viewed as even boring. This would depend on the specific task, the external context and more specifically on the cognitive and emotional components of the experience [4]. Repeated measurements should thusly yield relevant information on students' changing experience regarding a given activity. In essence, diary methods can inform us, how self-determined we are in our everyday work or studies etc. (e.g. [28]).

2.2 Study Motivation and CS1

In a student's life, several external rewards are available. Points from courses, course grades, GPA, collected credits, scholarships awarded due to study success and, ultimately, lucrative jobs. Learning programming can likewise be motivated by external rewards; an introductory programming course can, e.g., be included in a CS minor curriculum as a mandatory subject, but on the other hand it can be a voluntary course with likely implications for students' motivation. To better understand what yields and improves good retention rates in CS1 courses, in addition to course pedagogy [27, 41], it is arguably important to look into the "noncognitive side" of students' learning, i.e. how they experience it themselves.

Previously, in CER, motivation and retention in CS1 have been studied from multiple viewpoints (for an extensive review of CS1 research literature, see [21]). In general, students entering introductory programming courses have different kinds of motivation and goals [19], many entering the course wanting to learn and understand [4]. These entering motivations can, at least for some student sub-populations, predict retention [36], although there are also results suggesting that initial motivation might not contribute to students' performance after all [34]. This highlights the context-specificity of such studies, indicating that there might also be tacit factors that contribute to the observations.

The impact of course context has been acknowledged by researchers and educators alike. Some instructors, e.g., have changed the contents of their CS1 or the way the course is taught with the aim of increasing students' motivation [23, 33]. Such changes might need to be calibrated for the course population, however, as some changes with reportedly good prior experiences might not transfer [24].

2.3 Motivation During CS1 and Dropouts

While many of the motivation-related CS1 studies have been conducted e.g. as pre- or post-course surveys, there are in fact a handful of prior studies that have looked into changes in motivation over the span of an introductory programming course. As an example, Lishinski and Rosenberg [17] looked into students' motivation entering a course, during the course and at the end of it.

In general, these studies tend to indicate a higher motivation at the beginning of a course (e.g. [4, 10, 35]). Carbone et al. [4], e.g., note that students would start off intrinsically motivated, but when working on a task, experience a change to a state of low motivation (or achievement motivation) – or vice versa. They noted that while most students want to learn and understand, one problem might be a mismatch in expectations and existing skills [4]. Such problems could even lead to a situation where initial high motivation is replaced by learning avoidance related goals [10].

At the same time, when examining motivation measured on a weekly level, there can be ups and downs during a course [35]. Experimenting with both closed and open-ended programming assignments, Sharmin et al. [35] observed that the open-ended assignments could lead to higher motivation. On the other hand, they also found that some of the open-ended assignments were rated poorer when compared to the closed ones; one possibility that the authors posit is that those assignments might have provided too much freedom – they conclude that constructing motivating (open-ended) assignments is a task that one must be very careful at.

Generally, then, motivation fluctuates within CS1, which can be partially influenced by difficulties in programming assignments [4, 35]. There is also some evidence that high levels of motivation at the beginning of a course carry over to the end, at least for some students [36], but it is unclear whether such observations generalize [34]. After all, there can be a wide variety of reasons why students drop out of an introductory programming course [15, 26] with only some of the reasons being related to motivation.

3 METHODOLOGY

3.1 Context

The study was conducted in an introductory programming course of a Finnish university. The course is worth 5 ECTS[1] and targets CS minors; a separate CS1 course is offered for CS majors. The course consisted of eight compulsory rounds of assignments and a

[1]European Credit Transfer System; 5 ECTS amounts to approx. 135 hours of study.

ninth one which was non-compulsory – the duration of each round was one week. Each round consisted of four to five assignments. The course used Python as the programming language, and the contents of the course were typical to introductory programming courses; students were expected to learn to use variables, input and output, conditional statements, loops, functions and parameters, lists, dictionaries, as well as basic Math-specific Python functions and working with files.

The course had online lectures and materials. In addition, the course included "lab sessions" where teaching assistants helped students with course assignments. Lab sessions were held in-situ, with mandatory face masks due to the Covid-19 Pandemic. The lab sessions were popular; the course utilized an online queuing system that stored statistics – sometimes one had to wait for help for up to 30 minutes in a two hour lab session.

To pass the course, one needed to complete approximately one half of the assignments from each round. The assignments were submitted to an automated assessment system that rewarded points on progress.

3.2 Data

To collect data for this study, we were given a permission to embed a voluntary weekly questionnaire to the online course materials[2]. The questionnaire was the same on each week of the course.

The questionnaire given to students included (1) time spent during a week, and (2) eleven questions gauging students' motivation. Time spent in the course was evaluated through three questions which asked for time spent on lectures and materials, time spent on assignments, and time spent on other course related activities. Each of the questions had a range of answer options, mostly at 60 minute intervals. The three time-related questions were combined to form the factor "Time Spent" (meaning Time Spent Total) that is used in the analyses.

For measuring students' motivation, we took from the Intrinsic Motivation Inventory (IMI) the one subscale ("Interest/Enjoyment") that assesses intrinsic motivation per se [25, 39]. In addition, we created two subscales: a deeper drive towards programming or "Programming Appetite" manifesting, e.g., as wanting to learn more of a given programming topic, and "Programming Interest in Context", i.e. how interested students felt about their other studies during the CS1-course and whether they enjoyed their CS1-course more than those other studies. The items were answered using a Likert scale from 1 ("Not at all true") to 7 ("Completely true"). The items are displayed in Table 1.

Due to privacy, no data on student backgrounds, students' course points, or exam scores were available for the purposes of this study. Information on whether each student was active in a given week was available, however.

3.3 Answering Research Questions

To answer RQ1, How are the types of motivation and time spent on the course related to each other?, we study correlations between the types of motivation and time spent in the course. To answer RQ2, How do the types of motivation and time spent on the course change across the course weeks?, we visually analyze the data over the weeks, and further correlate the rounds with time spent and the three types of motivation. To answer RQ3, What sorts of subgroups of motivation and time spent on the course are there in the data?, we cluster students into groups to study whether there are distinct subpopulations. Finally, to answer RQ4, To what extent do the types of motivation and the time spent on the first round of the course predict activity on the last mandatory week?, we construct and evaluate machine learning models for predicting retention in the course using the types of motivation and the time spent in the first round of the course as predictive factors.

To confirm the three-factor structure (Intrinsic motivation, Programming appetite, Programming interest in context), we performed week-by-week Confirmatory Factor Analysis (CFA) [14] on the collected data, which showed that the loadings of the items into the proposed factors were acceptable, with the exception of one week. On the second week of the course, the IMI subscale loadings for the items 3 and 4 were poor. As this was the sole exception, we proceeded with the analysis. In the subsequent analyses, all correlations are conducted using Spearman's rank correlation. When interpreting statistical significance with p-values, we apply Bonferroni correction to account for the multiple testing problem, but report the uncorrected p-values for accuracy. Uncorrected p-values are reported using scientific E notation for writing very large (or small numbers). Round-by-round CFA scores and Cronbach's alphas are included in an online appendix[3].

Table 1: Items used in the research. The Intrinsic Motivation Inventory (IMI) subscale "Interest/Enjoyment" contains the below items 1 - 7, modified for this study to address CS1. Items 8-11 were created for this study (discussed in 3.2). The suffix (R) indicates a reverse coded item.

#	Item
1.	I very much enjoyed CS1 during this weekly round (the chapter and the assignments).
2.	This week of CS1 was fun to do.
3.	I thought this week of CS1 was boring. (R)
4.	This week of CS1 did not hold my attention at all. (R)
5.	I would describe this week of CS1 as very interesting.
6.	I thought this week was quite enjoyable.
7.	While I was working on the course this week, I was thinking about how much I enjoyed it.
8.	During this week, I felt that I wanted to learn more about the topic.
9.	The contents of this week felt exciting.
10.	During this week, I felt interested in my other, non-CS1 studies (regardless of what I thought of CS1). (R)
11.	I enjoyed this week of CS1 more than I enjoyed my other studies during the same time.

[2]The researchers of this study are not associated with the course.

[3]https://osf.io/vb4gd/?view_only=d0cb066168bf45598eaaad8c506161aa

Table 2: Weekly attending student and survey answer counts and weekly averages for intrinsic motivation (IMI), programming appetite (PA), programming interest in context (PIC), and time spent (TS) in minutes.

Round	Students	Answers	IMI	PA	PIC	TS (min)
1	835	712	5.31	5.44	4.12	312
2	769	592	5.06	5.16	4.14	409
3	712	508	4.02	4.27	3.55	690
4	731	502	4.60	4.68	3.85	603
5	697	450	4.08	4.28	3.55	702
6	652	403	4.10	4.24	3.48	785
7	651	386	4.36	4.44	3.71	715
8	605	344	4.17	4.19	3.59	751
9	502	271	4.68	4.81	3.81	803

4 RESULTS

4.1 Descriptive Statistics

The number of students attending the course, answers received of the survey, average intrinsic motivation score, average programming appetite score, average programming interest in context score, and average time spent (in minutes) over the course for each round are shown in Table 2. The number of students and survey answers are absolute numbers, while the types of motivation response values are averages from items answered to using a scale from 1 ("Not at all true") to 7 ("Completely true"). Overall, a total of 835 students participated in the first round of the course, while 605 students attended the eighth round which was the last compulsory round. The voluntary ninth round was attended by 502 students. In other words, 27.5% of the students dropped out before the last compulsory round and 39.9% of the students did not attend the voluntary ninth round. In the first round, 85.3% of the students answered the survey, while in the last compulsory round, 56.9% of the students answered the survey. In the final, voluntary round, 54.0% of the students answered the survey. Our analyses focus on those who answered the survey.

4.2 Motivation and Time Spent on the Course

Considering the time spent on the course, we can see from Table 2 that there is a clear increase in time spent. In the first two weeks, students spent on average 300-400 minutes on the course per week, while on the subsequent weeks, students spent on average 600-800 minutes per week. When considering the correlations of the three types of motivation and time spent on the course, all of the variables are correlated and all of the correlations are statistically significant. Time spent is negatively correlated with each of the three types of motivation. The types of motivation are positively correlated with each other. The strongest correlation can be observed between intrinsic motivation and programming appetite, whereby intrinsic motivation explains approximately 65.6% of the variance in programming appetite. On the other hand, the smallest correlation can be seen between time spent and programming interest in context, where the time spent explains approximately 4.4% of the variation in programming interest in context. The correlations are summarized in Table 3.

Table 3: Spearman's rank correlations of the types of motivations (intrinsic motivation = IMI, programming appetite = PA, programming interest in context = PIC) and time spent (TS). The lower left echelon presents the correlations and the upper right echelon presents the uncorrected p-values using scientific notation.

	TS	IMI	PA	PIC
TS	-	4E-121	1E-65	9E-40
IMI	-0.35	-	< 1E-300	3E-181
PA	-0.25	0.81	-	2E-122
PIC	-0.21	0.44	0.36	-

4.3 Changes over Course Rounds

Overall, the highest motivation point in terms of intrinsic motivation is in the first round of the course (avg 5.31) and the second highest intrinsic motivation is in the second round of the course (avg 5.06), while the lowest motivation point is in the third round of the course (avg 4.02). The observation for the first two rounds being rated the highest holds also for programming appetite and programming interest in context; all motivation scores seem to fluctuate over the course and there is no clear overall decline after an initial drop. Figure 1 displays an overview of the changes in the three types of motivation over the course rounds.

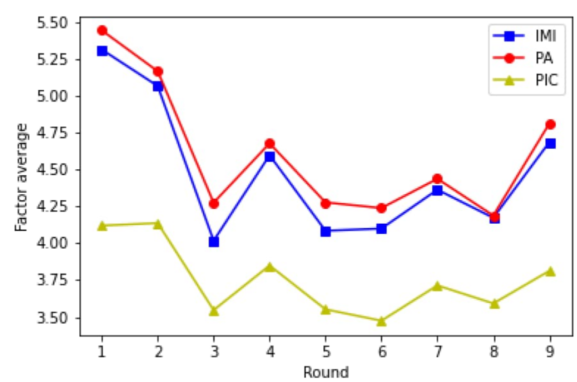

Figure 1: The three measured types of motivation, intrinsic motivation (IMI), programming appetite (PA), and programming interest in context (PIC) plotted over the nine rounds of the course.

We further studied the correlation of the round number and the time spent on each round and the three types of motivation. Overall, all of the variables are correlated and all of the correlations are statistically significant. Time spent is positively correlated with the course rounds, whereby approximately 18.5% of the time spent is explained by the course round. In other words, it seems that, on average, each subsequent round takes more time than the

Table 4: Spearman's rank correlations of the course rounds (R) and the types of motivation (intrinsic motivation = IMI, programming appetite = PA, programming interest in context = PIC), time spent (TS) over the data. The value in parentheses represents the p-values using scientific notation, while the values outside of parentheses describe the Spearman r.

	TS	IMI	PA	PIC
R	0.43 (4E-279)	-0.26 (2E-65)	-0.24 (3E-57)	-0.13 (5E-16)

Table 5: Clustering results for third round of the course. The table shows the number of students for each of the three clusters as well as the averages for the types of motivations (intrinsic motivation = IMI, programming appetite = PA, programming interest in context = PIC) and time spent (TS) in those clusters.

Cluster	Students	TS	IMI	PA	PIC
1	250	448	4.33	4.50	3.85
2	171	880	3.93	4.37	3.36
3	53	1622	2.94	3.19	2.77

previous one. The three types of motivation, on the other hand, are negatively correlated with the course rounds. The largest negative correlation is between the intrinsic motivation and the course rounds, whereby approximately 6.7% of the variance in intrinsic motivation is explained by the course round, while the smallest negative correlation can be seen between the course rounds and programming interest in context, where the rounds explains less than 1% of the variation in programming interest in context. The correlations are summarized in Table 4.

4.4 Identifying Subgroups

To identify groups within the data, we used the k-nearest neighbors algorithm [8], which is an unsupervised machine learning method. We first used the elbow method (for the principle, see [38]) to determine a suitable number of clusters – three seemed to work the best – and then ran the k-nearest neighbors algorithm with k as 3 to form the clusters for each week.

Overall, after a visual analysis, the three clusters can be summarized as follows. The first cluster has approximately 55% of the students. These students spend relatively little time on the course and have relatively high motivation (when compared to others). The second cluster makes up approximately 35% of the students. These students spend quite a bit of time on the course and have somewhat lower motivation scores than the first cluster. The third cluster has approximately 10% of the students. These students spend plenty of time on the course – approximately, on average, 3 times the amount of time that students in the first group spend. They also have low motivation. The sizes of the groups differ somewhat across the rounds, but the trends in time usage and motivation persist. Table 5 demonstrates, as an example, the cluster sizes and cluster factor averages for the third round of the course.

4.5 Predicting Dropouts

We used supervised machine learning to study to what extent the types of motivation and time spent can be used to predict retention. We trained and evaluated a body of often used machine learning algorithms for the task – we included Logistic regression, Random forest, Naive Bayes, and Support vector machine into the evaluated algorithms, which have all previously been used in studies focused on predicting academic performance [12]. As the input variables to predict whether a student would drop out, we used time spent, intrinsic motivation, programming appetite, and programming interest in context from the first week, and the target variable was whether the student was active on the last mandatory week. For completeness and to have a baseline, we included a majority class algorithm that always predicted that a student would be active on the last mandatory week.

Using 10-fold cross-validation to separate the data that the models were trained and tested on, and accuracy as the metric for model performance, we observed that the types of motivation and time spent did not yield a model that would perform better than the majority class model. The majority class algorithm had the best average performance (on average, 72.5% accuracy), while Logistic regression had the second place (on average, 72.4% accuracy). Other approaches were marginally poorer.

5 DISCUSSION

5.1 Fluctuating Motivation

Our findings resemble those by Sharmin et al. [35]: motivation seems to drop during a CS1 course, although this comparison comes with caveats. Perhaps closest to ours came their subscales "relevance" and "satisfaction" that would seem to tap into aspects of intrinsic motivation. We interpret our findings such that there is a triangle connection between weekly intrinsic motivation, time spent on a CS1 course round and, potentially, perceived difficulty of weekly assignments on the course; this is supported by previous research, which indicates that time spent on an assignment largely predicts its perceived difficulty [13]. Our analysis yielded interesting results about the connection between intrinsic motivation and time spent on the course, yet it is important to be open about how to read the results. While lowered intrinsic motivation can make completing assignments more sluggish, it could also plausibly be that a lot of time is spent unproductively – e.g. waiting for help, which in turn can also decrease motivation.

In contrast to intrinsic motivation and programming appetite that often went even hand in hand, we observed that the programming interest in context was often separate with students indicating that they were more interested in other subjects. Was it, e.g., because the participants were CS minors, or was it something about the course itself? This topic merits further study.

Also of note is that some students spent very high amounts of time in the CS1 course with at least some of them persevering through the course. In the smallest of the clusters, students effectively on average used nearly double the time allocated to the course; prior research on the time that students take to write solutions to programming problems suggests that differences between individual students can be even more considerable [7]. While spending a lot of time on a course is on the one hand admirable, on the other hand it seems rather worrisome from a self-regulated learning and overall study wellbeing standpoint, as this time is away from other activities, including rest.

5.2 Challenging Programming Assignments

Comparing the fluctuations in types of motivation with the observations of Sharmin et al. [35] – that motivation was linked with the difficulty of programming assignments – we consider that a part of the fluctuations is likely due to the way the course was organized. This was reflected in the jump in the time that students spent on the course, and could partially be explained by perhaps a too fast transition from easier programs to building relatively complex ones from scratch. In our view, this kind of transition encompassed multiple domains of difficulty outlined by du Boulay [6]. It would probably have been better to err on the side of caution and provide students more instructional scaffolding [45] through e.g. smaller programming assignments, seeking to keep students in their Zone of Proximal Development [43].

Sharmin et al. [35] seem to come to a rather similar conclusion: there was a sweet spot with freedom regarding doing exercises that yielded the best motivation scores. Further, at least when learning the basics, it would likely be better to have a larger number of small assignments rather than relatively few, complex ones such as in this CS1 course. This has been suggested previously by others as well (e.g. [42]) and has been shown to influence when students start their work and how they perform in subsequent more complex assignments [5]. Broader educational psychology literature also suggests that having little guidance (or too much freedom) in instruction is less effective for learning, at least when compared to more direct and guided instruction [16].

5.3 Notes on Dropout

In the course, 27.5% of the students dropped out before the last compulsory round. We explored the possibility of building a model to predict that dropout based on types of motivation and time spent on the first week, but did not find one that would perform well; this is in line with some earlier studies that have observed that early motivation does not predict performance [34]. While there is a wide variety of reasons why students drop out from introductory programming courses [15, 26], we believe that at least some of the drop-outs could have been averted.

As waiting for help itself can be demotivating, one aspect to consider would be the programming assignments themselves, as discussed in Section 5.2. Another aspect is the community – while programming is in a sense a solitary enterprise, creating a sense of togetherness might help share the burden and also perhaps help students react constructively to challenges they face. There was clearly no lack of trying, which was evidenced by both the measure time spent and lab session participation rates – there were up to 50 students attending a single two hour session despite ample opportunity to select between many different sessions per week, and sometimes one had to wait for up to 30 minutes for help in a single session.

We found this popularity of the labs an important indicator of how the course was experienced by the students, especially when viewed against a backdrop of a concurrent tentative return to the campus amid the Covid-19 pandemic.

5.4 Limitations of Study

We acknowledge that our study comes with multiple limitations, including sampling bias and self-reporting bias. The first one is of special concern in light of the results of the study; as often most active students participate in voluntary activities, it is conceivable that motivation scores could even be lower for the whole course population. We also acknowledge that the study was conducted in a single course in a single university, which raises questions about the generalisability of the observations. At the same time, similar results have been observed in other contexts as well.

6 CONCLUSION

In this work, we studied fluctuations in motivation and time spent over the rounds of an introductory programming course. To summarize, our research questions and their answers are as follows.

RQ1 How are the types of motivation and time spent on the course related to each other? **Answer:** Overall, we observe that the types of motivation are positively correlated with each other, while negatively correlated with time spent on the course.

RQ2 How do the types of motivation and time spent on the course change across the course weeks? **Answer:** While the motivation is relatively high early on in the course, motivation tended to drop over the rounds, while time spent on the course tended to increase over the rounds.

RQ3 What sorts of subgroups of motivation and time spent on the course are there in the data? **Answer:** We identified three subgroups of students whereby one group spent relatively little time and had high motivation, another group was in between, and a third group spent relatively much time and had low motivation.

RQ4 To what extent do the types of motivation and the time spent on the first round of the course predict activity on the last mandatory week? **Answer:** Motivation and time spent on the course, collected from the first round of the course, were not useful in predicting activity on the last mandatory week of the course.

Our study provides further evidence on the link between types of motivation and time spent, and we view that the links between time spent, motivation and assignment complexity [13, 35] should be explored further. Some students spent substantially more time on the course tasks than expected and had rather low motivation scores, yet our attempts to predict activity based on motivation and time spent proved futile, which is also in line with some earlier research [34]. Hence the reasons for dropping out might not only be related to motivation and may lie elsewhere as suggested e.g. in [15].

In our future work, we will be replicating the study in a new iteration of the course, and are also looking into a CS1 course intended for CS majors, which should complement the picture of how students' motivation is sustained across differing CS1 contexts. We are also looking into how students' self-assessed competence and stress fluctuate week-to-week. In addition, we are examining more deeply the effects of programming assignment difficulty on motivation.

REFERENCES

[1] Jens Bennedsen and Michael E Caspersen. 2007. Failure rates in introductory programming. *ACM SIGCSE Bulletin* 39, 2 (2007), 32–36.

[2] Jens Bennedsen and Michael E Caspersen. 2019. Failure rates in introductory programming: 12 years later. *ACM inroads* 10, 2 (2019), 30–36.

[3] Niall Bolger, Angelina Davis, and Eshkol Rafaeli. 2003. Diary methods: Capturing life as it is lived. *Annual review of psychology* 54, 1 (2003), 579–616.

[4] Angela Carbone, John Hurst, Ian Mitchell, and Dick Gunstone. 2009. An exploration of internal factors influencing student learning of programming. In *Proceedings of the Eleventh Australasian Conference on Computing Education-Volume 95*. 25–34.

[5] Paul Denny, Andrew Luxton-Reilly, Michelle Craig, and Andrew Petersen. 2018. Improving complex task performance using a sequence of simple practice tasks. In *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. 4–9.

[6] Benedict Du Boulay. 1986. Some difficulties of learning to program. *Journal of Educational Computing Research* 2, 1 (1986), 57–73.

[7] Fabian Fagerholm and Arto Hellas. 2020. On the Differences in Time That Students Take to Write Solutions to Programming Problems. In *2020 IEEE Frontiers in Education Conference (FIE)*. IEEE.

[8] Evelyn Fix and Joseph Lawson Hodges. 1951. Discriminatory analysis. Nonparametric discrimination: Consistency properties. (1951).

[9] CC2020 Task Force. 2020. *Computing Curricula 2020: Paradigms for Global Computing Education.* Association for Computing Machinery, New York, NY, USA.

[10] Melissa Patterson Hazley, Duane F Shell, Leen-Kiat Soh, L Dee Miller, Vlad Chiriacescu, and Elizabeth Ingraham. 2014. Changes in student goal orientation across the semester in undergraduate computer science courses. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*. IEEE.

[11] Joel M Hektner, Jennifer A Schmidt, and Mihaly Csikszentmihalyi. 2007. *Experience sampling method: Measuring the quality of everyday life.* Sage.

[12] Arto Hellas, Petri Ihantola, Andrew Petersen, Vangel V Ajanovski, Mirela Gutica, Timo Hynninen, Antti Knutas, Juho Leinonen, Chris Messom, and Soohyun Nam Liao. 2018. Predicting academic performance: a systematic literature review. In *Proceedings companion of the 23rd annual ACM conference on innovation and technology in computer science education*. 175–199.

[13] Petri Ihantola, Juha Sorva, and Arto Vihavainen. 2014. Automatically detectable indicators of programming assignment difficulty. In *Proceedings of the 15th Annual Conference on Information technology education*. 33–38.

[14] Karl G Joreskog, Sumorbom Dag, and Jay Magidson. 1979. *Advances in factor analysis and structural equation models.* Abt books.

[15] Päivi Kinnunen and Lauri Malmi. 2006. Why students drop out CS1 course?. In *Proceedings of the second international workshop on Computing education research*. 97–108.

[16] Paul A Kirschner, John Sweller, and Richard E Clark. 2006. Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist, Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching. *Educational Psychologist* 41, 2 (2006), 75–86.

[17] Alex Lishinski and Joshua Rosenberg. 2021. All the Pieces Matter: The Relationship of Momentary Self-efficacy and Affective Experiences with CS1 Achievement and Interest in Computing. In *Proceedings of the 17th ACM Conference on International Computing Education Research*. 252–265.

[18] Alex Lishinski and Aman Yadav. 2019. Motivation, Attitudes, and Dispositions. In *The Cambridge handbook of computing education research*, Sally A Fincher and Anthony V Robins (Eds.). Cambridge University Press, Chapter 28, 801–826.

[19] Alex Lishinski, Aman Yadav, Jon Good, and Richard Enbody. 2016. Learning to program: Gender differences and interactive effects of students' motivation, goals, and self-efficacy on performance. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*. 211–220.

[20] Andrew Luxton-Reilly. 2016. Learning to program is easy. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*. 284–289.

[21] Andrew Luxton-Reilly, Ibrahim Albluwi, Brett A Becker, Michail Giannakos, Amruth N Kumar, Linda Ott, James Paterson, Michael James Scott, Judy Sheard, and Claudia Szabo. 2018. Introductory programming: a systematic literature review. In *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. 55–106.

[22] Jane Margolis, Allan Fisher, and Faye Miller. 1999. Caring about connections: Gender and computing. *IEEE technology and society magazine* 18, 4 (1999), 13–20.

[23] Stefanie A Markham and KN King. 2010. Using personal robots in CS1: experiences, outcomes, and attitudinal influences. In *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education*. 204–208.

[24] José Alfredo Martínez-Valdés, J Ángel Velázquez-Iturbide, and Raquel Hijón-Neira. 2017. A (relatively) unsatisfactory experience of use of Scratch in CS1. In *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality*. 1–7.

[25] Edward McAuley, Terry Duncan, and Vance V Tammen. 1989. Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: A confirmatory factor analysis. *Research quarterly for exercise and sport* 60, 1 (1989), 48–58.

[26] Andrew Petersen, Michelle Craig, Jennifer Campbell, and Anya Tafliovich. 2016. Revisiting why students drop CS1. In *Proceedings of the 16th Koli Calling International Conference on Computing Education Research*. 71–80.

[27] Leo Porter and Beth Simon. 2013. Retaining nearly one-third more majors with a trio of instructional best practices in CS1. In *Proceeding of the 44th ACM technical symposium on Computer science education*. 165–170.

[28] Harry T Reis, Kennon M Sheldon, Shelly L Gable, Joseph Roscoe, and Richard M Ryan. 2000. Daily well-being: The role of autonomy, competence, and relatedness. *Personality and social psychology bulletin* 26, 4 (2000), 419–435.

[29] Anthony Robins, Janet Rountree, and Nathan Rountree. 2003. Learning and teaching programming: A review and discussion. *Computer science education* 13, 2 (2003), 137–172.

[30] Richard M Ryan and Edward L Deci. 2000. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American psychologist* 55, 1 (2000), 68.

[31] Richard M Ryan and Edward L Deci. 2009. Promoting Self-Determined School Engagement. In *Handbook of Motivation at School*. Routledge, Chapter 9.

[32] Richard M Ryan and Edward L Deci. 2017. *Self-determination theory: Basic psychological needs in motivation, development, and wellness.* Guilford Publications.

[33] Bianca L Santana and Roberto A Bittencourt. 2018. Increasing motivation of cs1 non-majors through an approach contextualized by games and media. In *2018 IEEE Frontiers in Education Conference (FIE)*. IEEE.

[34] Pablo Schoeffel, Raul Sidnei Wazlawick, and Vinicius Ramos. 2017. Impact of pre-university factors on the motivation and performance of undergraduate students in software engineering. In *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 266–275.

[35] Sadia Sharmin, Daniel Zingaro, and Clare Brett. 2020. Weekly Open-Ended Exercises and Student Motivation in CS1. In *Koli Calling'20: Proceedings of the 20th Koli Calling International Conference on Computing Education Research*. 1–10.

[36] Duane F Shell, Leen-Kiat Soh, Abraham E Flanigan, and Markeya S Peteranetz. 2016. Students' initial course motivation and their achievement and retention in college CS1 courses. In *Proceedings of the 47th ACM technical symposium on computing science education*. 639–644.

[37] Simon, Andrew Luxton-Reilly, Vangel V. Ajanovski, Eric Fouh, Christabel Gonsalvez, Juho Leinonen, Jack Parkinson, Matthew Poole, and Neena Thota. 2019. Pass Rates in Introductory Programming and in Other STEM Disciplines. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education* (Aberdeen, Scotland Uk) *(ITiCSE-WGR '19)*. Association for Computing Machinery, New York, NY, USA, 53–71. https://doi.org/10.1145/3344429.3372502

[38] Robert L Thorndike. 1953. Who belongs in the family? *Psychometrika* 18, 4 (1953), 267–276.

[39] Nikolaos Tsigilis and Argiris Theodosiou. 2003. Temporal Stability of the Intrinsic Motivation Inventory. *Perceptual and motor skills* 97 (09 2003), 271–80. https://doi.org/10.2466/PMS.97.4.271-280

[40] Simone JW Verhagen, Laila Hasmi, Marjan Drukker, Jim van Os, and Philippe AEG Delespaul. 2016. Use of the experience sampling method in the context of clinical trials. *Evidence-based mental health* 19, 3 (2016), 86–89.

[41] Arto Vihavainen, Jonne Airaksinen, and Christopher Watson. 2014. A systematic review of approaches for teaching introductory programming and their influence on success. In *Proceedings of the tenth annual conference on International computing education research*. 19–26.

[42] Arto Vihavainen, Matti Paksula, and Matti Luukkainen. 2011. Extreme apprenticeship method in teaching programming for beginners. In *Proceedings of the 42nd ACM technical symposium on Computer science education*. 93–98.

[43] Lev Semenovich Vygotsky and Michael Cole. 1978. *Mind in society: Development of higher psychological processes.* Harvard university press.

[44] Christopher Watson and Frederick WB Li. 2014. Failure rates in introductory programming revisited. In *Proceedings of the 2014 conference on Innovation & technology in computer science education*. 39–44.

[45] David Wood, Jerome S Bruner, and Gail Ross. 1976. The role of tutoring in problem solving. *Child Psychology & Psychiatry & Allied Disciplines* (1976).