Computer programming skills: A cognitive perspective

Graafsma, Irene

DOI:
10.33612/diss.168003240

Publication date:
2021

Chapter 4

Autistic traits and programming learning outcomes in an introductory computing course[1]

[1] This chapter has been submitted as an article for publication.

4.1 INTRODUCTION

Currently, we rely on individuals with programming skills to deliver and maintain a wide range of software-based tools and services, such as apps, websites, games, data analysis tools and online work and teaching environments. In tandem, the number of individuals pursuing training in programming and software design has been increasing as demand grows for these skills (European Commission, 2018). However, as in all areas of education and training, some individuals are more successful than others in the pursuit to become proficient programmers. Researchers have been searching for ways to predict which students are most likely to succeed, as well as those who may have difficulty in acquiring this skill (Wray, 2007). This knowledge could be used to inform career counselling by identifying students best suited to these courses and associated professions or to identify those that may need additional support. Some studies have shown that general intelligence and some specific cognitive skills (e.g., mathematical skills, working memory, logical reasoning etc.) are important when learning to program, and therefore predict programming skills (Guzdial & du Boulay, 2019; Pena & Tirre, 1992; Shute, 1991; Tirre & Pena, 1993; Webb, 1985). In previous work we found that logical reasoning, algebra and vocabulary learning skills were predictors of programming skill (Chapter 3), highlighting that individual differences in neurocognitive profiles can predict education and vocational outcomes. Since the 1980s, whilst recruiters and organisational psychologists have mostly focussed on the role of personality traits in workplace performance and learning (Barrick et al., 2001), there is now growing recognition that other individual differences – including unique neurocognitive profiles – may help better predict programming skill outcomes (Baren-Cohen, 2001; Catherine & Wheeler, 1994; Focquaert et al., 2007; Golding et al., 2006; Wray, 2007). Indeed, the recent neurodiversity movement has highlighted that all individuals differ in their cognitive profiles and neural make-up which gives rise to unique strengths, and this can be more pronounced in portions of the population with neurodevelopmental or psychiatric conditions (e.g., autism, schizophrenia; den Houting, 2019).

Autism is a neurodevelopmental condition characterised by differences and difficulties in social interactions and communication, specific sensory processing sensitivities and tendencies towards repetitive behaviours and restricted interests

(American Psychiatric Association, 2013; Lord et al., 2000). Autism is heterogenous, which means that the way it presents in individuals can vary widely (Lenroot & Yeung, 2013). The 'broader autism phenotype' account suggests that autism is the extreme end of a continuous spectrum, with autistic traits also being present in non-autistic relatives of autistic individuals, and in the general neurotypical population (Baron-Cohen et al., 2001; Landry & Chouinard, 2016). It has been argued that autistic traits may influence a person's interests and talents (Baron-Cohen et al., 2001). For example, Baren-Cohen et al. (2007) showed that there were up to seven times more autistic individuals undertaking mathematics degrees at Cambridge University compared to other degrees, suggesting that autistic individuals have increased interest and aptitude for mathematics. Therefore, it is of interest whether autistic traits also relate to programming aptitude.

Some support for the idea that people with autistic traits are more successful programmers comes from a study by Baron-Cohen et al. (2001), who evaluated autistic traits using a self-evaluation questionnaire - the Autism Spectrum Quotient (AQ; Baron-Cohen et al., 2001). They found that science students, including those in computer science, scored higher on the AQ scale than students in the social sciences and humanities. Furthermore, science students in fields that are more human or life-centred (e.g., biology and medicine), had a lower AQ score than students in more abstract fields of science (e.g., mathematics, computer science and physics). If autistic traits do indeed predict programming skill, it is of interest to explore why this is the case. Baron-Cohen (2012) proposed that genes underlying autism predispose unique neurodevelopmental pathways which lead individuals to process information differently. Specifically, Baron-Cohen suggests that people with autistic traits have a stronger tendency to 'systemise', that is, they have a tendency to process information and understand phenomena by identifying patterns and rules (Baron-Cohen, 2006). It is argued that 'systemisers' find it easier to study systems in nature, such as the laws of physics, or man-made systems (e.g., train schedules). This information processing style has also been proposed to shape the way autistic individuals understand human social behaviour by trying to fit stringent social rules, rather than intuitively and flexibly evaluating social information across contexts (Baron-Cohen, 2012). Alternatively, Baron-Cohen (2012) argues that people with a more empathizing-driven cognitive style intuitively relate to and understand other's emotions, and that this is negatively associated with autistic traits. For example, if someone is crying – an empathiser may intuitively feel compelled to comfort them, while a systemiser may learn that tears are a sign of sadness, and that the standard social protocol is to offer a tissue. These constructs of empathizing versus systemizing have been argued to (1) characterise autistic individuals and those with autistic traits (high systemizing and low empathizing would indicate high autistic traits) and (2) be associated with other outcomes, including career choice. For example, Focquaert et al. (2007) found that individuals in the sciences possessed a cognitive style that was more systemizing-driven than empathizing-driven, whereas individuals in humanities possessed a cognitive style that was much more empathizing-driven than systemizing-driven. They argue that this relationship reflects a difference in brain structure between empathisers and systemisers that makes them prone to choose a degree that matches their thinking style. A remaining question is whether thinking style only affects degree choice, or whether it also predicts learning success within such degrees.

Wray (2007) tested whether measures of systematizing (SQ) and empathizing (EQ) predict programming skill. Higher scores on these measures indicate a greater natural tendency to systemise or empathise with others. However, Wray found that neither SQ nor EQ alone predicted programming performance, but that the difference between these measures (SQ minus EQ) did. That is, having relatively higher SQ than EQ was associated with greater programming abilities. Nevertheless, the generalisability of these findings is limited by the relatively modest sample size (N = 19) for a study examining individual differences and the absence of females in the sample. Indeed, a later study by Borzovs et al. (2017) failed to replicate Wray's (2007) findings in a larger and more diverse sample (n = 73, 39.7% female). They found no significant correlations between SQ, EQ, nor the difference between SQ and EQ, and programming skill. This could partly have been influenced by the high attrition rates in the study, when they attempted to examine long-term learning outcomes (Borzovs et al., 2017). Together this suggests that SQ and EQ are not reliable predictors of programming skill. However, it is important to recognise that SQ and EQ can only provide an indirect measure of autistic traits, and the sensitivity and validity of this indirect measure for investigation of individual differences within neurodiverse populations remains unclear. As such, SQ and EQ scores may not be sufficiently sensitive to detect a possible relationship between autistic traits and programming skill. This possibility is supported by a study by Wheelwright et al. (2006), who found that there were only moderate correlations between SQ, EQ and the AQ. Indeed, the idea that autism – and by extension autistic traits - is characterised by reduced abilities or tendencies to empathise with others has been disputed by proponents of the 'double empathy problem' (e.g., Milton et al., 2012; Mitchell et al., 2019). They argue that the evidence for empathizing deficits in autistic individuals is inconsistent – with social challenges better characterised by an incompatibility in empathising between autistic and non-autistic individuals, rather than a reduced capacity to empathise in autism. For this reason, exploring the relationship between autistic traits and programming skill acquisition requires a more holistic measurement of autistic traits.

Critically, no previous study has explicitly examined the relationship between autistic traits and programming skill. Therefore, the aim of the current study was to test whether autistic traits, when measured using the AQ, predict programming skill at the end of a programming course. We define programming skill based on both test performance during the course and generalised programming skill at the end of the course assessed by an independent measure of programming skill (Parker et al., 2016). First, we examined how AQ scores in our student sample compared to the general population. Then we investigated the predictive value of the AQ for course-related and generalised programming skill. We hypothesised that higher autistic traits at the start of the semester would be associated with better programming skill on the course assessments as well as better generalised programming skill at the end of the semester.

We also addressed two additional exploratory questions. Firstly, based on previous research, it remains unclear which autistic traits predict programming skill. To explore whether there are specific domains of autistic traits that best characterise this relationship, we analysed whether any of the individual AQ subscales predicted programming skill (see Table 4.1, below). Significant subscale effect(s) may elucidate the specific cognitive features which drive any observed relationship between autistic traits and programming outcomes.

Secondly, we explored the relationship between autistic traits and cognitive skills related to programming skill. Even if autistic traits do not directly predict programming skill, it may be that the cognitive skills learners rely on as programmers vary depending on their

AQ score. Therefore, we tested whether autistic traits exhibited any relationships with the cognitive skills that play a role when learning to program (i.e., logical reasoning, pattern recognition, algebra, vocabulary learning, grammar learning). For this exploratory analysis, we had no a priori predictions.

4.2 METHODS

4.2.1 Ethics statement

The protocol for the current study received ethical approval from the Macquarie University Human Research Ethics Committee (Reference number: 5201800224). We followed the approved protocol where all students on a programming course received an information form at the start of the Qualtrics survey which gave them the choice to consent for their data to be used for research. Only students who consented for their data to be used were included in the current study.

4.2.2 Participants

Students enrolled in an undergraduate "Introduction to Programming" course at Macquarie University (COMP115) completed a testing session as part of a mandatory tutorial in the first and the final weeks of their 13-week course. Of the 838 students, 344 consented to their data being used in the current study. For the majority of students, this course was part of their mandatory study program. Most students were enrolled in Engineering or Information Technology degrees (67%), but there were also students from a wide variety of other Science and Arts majors, ranging from science to society, history and languages. Participants were excluded if they indicated, in a post-test probe questionnaire, that they had cheated or had not seriously attempted to answer the questions or if they self-reported a less than "Good" level of English on a 5-point rating scale from 1 (minimal) to 5 (native; 62 participants; see OSF link for details of the post-test probe questionnaire: https://osf.io/5d4s6/?view _only=0d88ef3d4da346779f82d20aa4f6df72). Thus, the results reported here are from the remaining 282 participants (49 female, 204 male, 2 other, 27 no gender given; mean age 19.32 years, *SD* = 3.09).

4.2.3 Materials

The results reported here are part of a larger study examining various aspects of cognition in relation to programming skill (see Chapter 3). For the current study, students completed a demographics questionnaire, the Autism Spectrum Quotient, five tests of cognitive skill (logical reasoning, pattern recognition, algebra, vocabulary learning, grammar learning), the Second Computer Science 1 Short (SCS1-S) programming test and we obtained their grades for the course. In addition, as part of the larger study they also completed a 'Sense of Agency' measure (Polito et al., 2013), however, these data are not reported here. All tests were presented in a Qualtrics survey, see Procedure for details.

Demographics

We included two questionnaires to obtain background information and demographics. At the start of the course we collected demographic information including age, gender, degree major, level of English, knowledge of programming languages and previous programming experience. At the end of the course we asked the students about their attendance and time spent on the course. We also asked whether they had cheated in any way whilst completing the tests.

Autism Spectrum Quotient

Students completed the full 50 item version of the Autism Spectrum Quotient (AQ; Baron-Cohen et al., 2001). Participants were asked to choose the answer option that most closely indicated how much they agreed with the statement. The AQ consists of five subscales, each consisting of 10 questions: Social skill, Attention switching; Attention to detail; Communication; and Imagination. Table 4.1 shows one example item per subscale. For each item students selected one response that best described how strongly each item applied to them. The response options were: Definitely agree - Slightly agree - Slightly disagree - Definitely disagree. Half of the items were reverse scored. For each item one point would be counted if participants gave one of the two answers in accordance with autistic traits (e.g., for the item "I prefer to do things the same way over and over again." one point would be scored if the participant answered Definitely agree or Slightly agree, and zero points would be scored if the participant replied Slightly disagree or Definitely disagree. Therefore, the possible range of scores for this measure was 0-50, with higher scores indicative of more autistic traits. A profile of high autistic traits would be low social skill, low attention switching, high attention to detail, low communication, and low imagination.

Table 4.1. Example items for each subscale of the AQ.

Subscale	**Example question**
Social skill	*I find it hard to make new friends.*
Attention switching	*I prefer to do things the same way over and over again.*
Attention to detail	*I often notice small sounds when others do not.*
Communication	*Other people frequently tell me that what I've said is impolite, even though I think it is polite.*
Imagination	*I don't particularly enjoy reading fiction.*

Note: These examples are all worded so that they produce an "agree" response for high autistic traits. Approximately half of the items were worded in the opposite way, where a "disagree" response indicated high autistic traits. Those items were reversed scored (e.g., "I prefer to do things with others rather than on my own.").

Cognitive skills tests[1]

Logical reasoning. We used the syllogism test described by Handley et al. (2002). Each item was a syllogism of the form "If it is a triangle then it is yellow. It is yellow. It is a triangle." Participants were asked to evaluate whether the final statement followed logically and with certainty from the previous statements. The test had 16 items and two parallel versions. Version 1 used the exact items from Handley et al. (2002) while Version 2 used the same questions but with different shapes and colours (e.g., "If it is a rectangle then it is not pink".). Participants were randomly assigned to complete Version 1 or Version 2

[1] Full tests, descriptive results for the tests, and correlations between tests can be found on the OSF: https://osf.io/5d4s6/

depending on their student identification number. They had five minutes to complete all items.

Pattern recognition. Pattern recognition skill was assessed with Part 1 Number Series from the Programming Aptitude Test from IBM (IBM, 1968) [2]. In each question the participant was presented with a series of six numbers and had to determine what the pattern of the sequence was, and then select the next correct number in the sequence from five alternatives (e.g., question: 3 6 9 12 15 18, answer options: 19 20 **21** 22 23). We split the test into two parallel versions, alternating even and uneven item numbers in each version (e.g., Version 1 included items 1,4, 5, 8… etc…) to ensure equal difficulty. Participants were randomly assigned to complete Version 1 or Version 2 depending on their student identification number. Each version consisted of 13 items and participants had five minutes to complete as many as possible.

Algebra. Algebra skill was measured with an adapted version of Part 3 from the Arithmetic Reasoning subtest of the Programming Aptitude Test from IBM (IBM, 1968)[2]. In the original version, the test presented numerical mathematics word-problems with five alternative answers. We changed the questions so that each question was followed by four formulas (equations) as answer options. Participants were asked to select the formula that would produce the correct answer. This enabled measurement of abstract and mathematical reasoning rather than arithmetic. The answer equation options were specifically created for this study. We split this test into two parallel versions, with alternating even and uneven question numbers in each version. Participants were randomly assigned to complete Version 1 or Version 2 first depending on their student identification number. They had 15 minutes to complete as many questions as possible.

[2] PAT Use Courtesy of International Business Machines Corporation, © International Business Machines Corporation.

As an example, we present item 3 from Version 1:

“The temperature at 1:00 pm was T1 and at 6:30 pm it was T2.

Assuming a constant rate of change, what was the temperature at 4pm?”

With answer options:

a) T2-((6.5-1)(T1-T2)/(4-1))

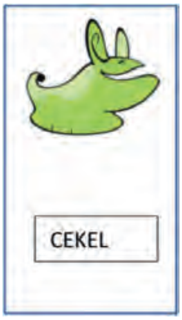
b) (4-1)(T1-T2)/(6.5-1)

c) T1-((6.5-1)(T1-T2)/(4-1))

d) T1-((4-1)(T1-T2)/(6.5-1))

Vocabulary learning. This test was based on the vocabulary learning subtest of the Language Learning Aptitude for MA students (LLAMA; Rogers et al., 2017). Participants learned the names of a series of creatures and were told that they would be tested on them later. They were instructed not to take any notes during the test. Participants were presented with 20 pictures of novel creatures (selected from Romanova, 2015) paired with 20 novel words (e.g., CEKEL, as shown in Figure 4.1) simultaneously on the screen and were given 2.5 minutes to memorise them. They then underwent two testing sessions – the first immediately after the learning phase. In the testing phase, all the creatures and names were displayed on the screen and participants were asked to drag and drop the names under the correct pictures within 3.5 minutes. Approximately 30 minutes later, at the end of the experimental session, they were tested again to assess delayed recall. We devised two parallel versions of the test using different creatures and novel words in each. Participants were randomly assigned to complete Version 1 or Version 2 depending on their student identification number.

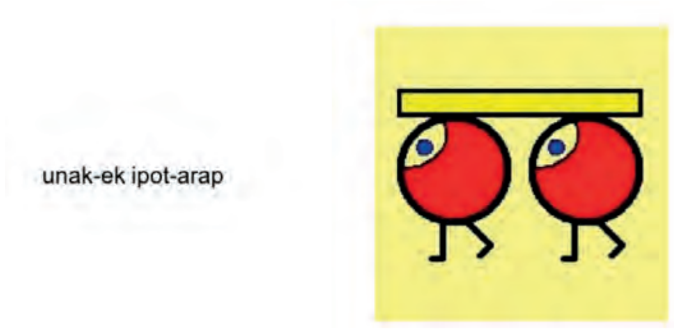
Figure 4.1. Example of an item in the vocabulary learning test.



Note: This figure shows one of 20 learning items on the vocabulary learning test. Participants were asked to memorise the names of the creatures. In the immediate and delayed recall stages they were asked to drag and drop the correct names under the corresponding pictures.

Grammar learning. This test was based on the grammar learning subtest of the LLAMA language aptitude test (Rogers et al., 2017) and Part 4 of the Pimsleur Language Aptitude Battery (Pimsleur et al., 2004). Participants were presented with three example blocks, each containing three example picture descriptions written in a novel, artificial language with simple grammatical rules (i.e., nine examples in total). One example sentence is “unak-ek ipot-arap”, which describes two red circle creatures walking underneath a rectangle as shown in Figure 4.2. The examples were then followed by 20 questions in which participants were asked to select the grammatically correct description of a picture from amongst 4 sentences in the novel language. Participants were free to scroll back and forth through all questions and examples. Students were given 8 minutes to complete the test.

Figure 4.2. Example of a learning item from the grammar learning test.



Note: This figure shows a learning example from the grammar learning test. The sentence on the left describes the image on the right. Participants were asked to use examples like this one to answer questions where they had to select the correct sentence describing a new picture from four answer alternatives.

Outcome measures

Shortened version of the Second Computer Science 1 (SCS1-S). This test was based on the SCS1 (Parker et al, 2016). We used a fully computer-based version, that was split into two parallel versions based on the difficulty of the questions as reported by Parker et al. (2016) and our own pilot experiments (see Chapter 2 for details of the psychometric properties of each subtest). Participants were randomly assigned to complete Version 1 or Version 2, depending on their student identification number. The SCS1 uses an artificial programming language invented by the test developers. Participants were instructed to use a pseudocode guide which provided them with information about the syntax and features of the programming language. This guide could be accessed in a separate browser window by clicking a button in the Qualtrics survey. Participants had 30 minutes to complete as many questions as possible. As the two versions were not of equal difficulty (see Chapter 2), we standardised the scores for each version and used these in the analysis.

Student grades. We used the students' final grades on the main course assessments of their university undergraduate programming course. The main assessments consisted of five module tests each with open questions in which students were asked to answer conceptual questions or to solve small programming problems. Each of the module tests could be attempted three times on three different occasions throughout the semester, as well as during the final exam testing session, which took place two weeks after the testing session with the SCS1-S. The student's highest score on each module was used to calculate the total grade. The five module test topics were: variables & conditionals, loops, functions, arrays & strings, and program design & problem solving. For more information see the Unit Guide in the Cognition of Coding project on the Open Science Framework (https://osf.io/5d4s6/?view_only=0d88ef3d4da346779f82d20aa4f6df72). For our analysis, we used the raw module test scores from each student's best attempt averaged over the subtopics.

4.2.4 Procedure

We presented all tests in Qualtrics. Students were given a link to the Qualtrics survey during their first and last tutorial of the course and completed the tests individually. During each test participants could scroll back and forth through the questions and, where appropriate, saw a countdown of the remaining time in the corner of the screen. Students were told that they were allowed to use pen and paper for all tests except for vocabulary learning.

The testing sessions were led by the regular course tutors. Tutors briefly introduced the study, after which students followed the instructions given in the Qualtrics program. Participants were informed that the study was being conducted to see how students learn computer programming, and which skills and personality traits may be important in that process. The students were not given any specific information about what the tests were meant to be measuring nor what the expectations of the study were.

Students were instructed to complete the tests individually at their own pace in the online Qualtrics system. All tests had a time limit that resulted in the survey automatically moving on to the next test if the student had not completed within a set time. In order to prevent students from skipping through all tests without attempting them, the cognitive tests and the AQ were presented in such a way that participants could not move on to the next test for at least one minute. For the programming test students could only move on after 5 minutes. For each test, instructions were provided on a separate page of the survey that was displayed for 20 seconds before the student could progress. The order of tests for all participants in the first session was: Vocabulary learning including immediate recall; pattern recognition; algebra; logical reasoning, vocabulary delayed recall; grammar learning; and demographic questionnaire. In the second session participants only completed the short form of the SCS1 and a demographic questionnaire. Session 1 took approximately one hour and Session 2 about 30 minutes. Testing sessions took place in the first and last weeks of the semester (i.e., 12 weeks apart). Students who could not attend these tutorials (3% of participants) were allowed to complete the tests at home.

4.2.5 Analysis

Analyses were performed according to a predetermined analysis plan (https://osf.io/n836g). Participants were only included if they attempted all cognitive tests, the SCS1-S and completed the AQ (N = 223). To compare our sample to the average AQ scores in the general population, we computed the average AQ score of our sample, as well as the average AQ score by gender. We compared this to that of the general population, as reported by Ruzich et al. (2015), using independent samples t-tests. To determine whether autistic traits (AQ total score) predicted generalised programming skill (scores on the SCS1-S) and programming skill in the course (final course grade), we used linear regression models. To investigate whether these effects were influenced by previous programming experience we also ran the regression models while controlling for whether or not this was the participant's first programming experience. To examine whether specific features of the AQ scale were related to programming skill we performed an exploratory analysis where we ran two linear regression models examining whether the AQ subscales predicted SCS1-S or final grade.

To examine whether AQ related to underlying cognitive skills, we performed an exploratory analysis for which we computed correlations between the AQ total score and the various cognitive skills (i.e., pattern recognition, logical reasoning, algebra, vocabulary learning and grammar learning). We only included delayed recall in the analyses because immediate and delayed recall of the vocabulary learning test were very highly correlated. For all analyses we used both Null Hypothesis Significance Testing and Bayesian statistics to test support for the null versus the alternative hypotheses. Bayes factors (BF_{10}) between 0 and 0.333 show support for the null hypothesis, with lower values showing stronger support. Bayes factors between 0.333 and 3 are considered inconclusive. Bayes factors above 3 show support for the alternative hypothesis, with higher values showing stronger support (Rouder et al., 2009).

4.3 RESULTS

The mean score on the AQ in the current sample was 19.35 (SD = 5.80). The means for male and female participants did not differ significantly ($t(67.61) = 0.62$, $p = .54$; Males: mean = 19.44 (SD = 5.85), Females: 18.84 (SD = 5.69)) and the Bayes factor (BF_{10} = 0.214) showed moderate support for the null hypothesis, together suggesting that AQ scores did not differ by gender. The mean AQ score in the current sample was significantly higher than the mean of 16.94 (SD = 5.59) in the general population from a systematic review of the literature by Ruzich et al. (2015) ($t(5152) = 6.26$, $p < 0.001$).

AQ scores at the start of the semester did not predict programming skill at the end of the semester on either the course grade (Linear Regression: $\beta = .070$, $p = .301$, BF_{10} = 0.243) or the SCS1 ($\beta = .050$, $p = .454$, BF_{10} = 0.190). This pattern did not change after controlling for whether this was the first programming experience (SCS1: $\beta = .050$, $p = .438$, BF_{10} = 0.214[3]; course grade: $\beta = .053$, $p = .418$, BF_{10} = 0.256). Similarly, none of the individual subscales were significant predictors of programming skill on either the SCS1 or course grades (see Table 4.2). This is also supported by the Bayesian analyses which showed weak support for the null hypothesis for all subscales in the model with the SCS1 scores (all BF_{10} = 0.250, suggesting 4 times as much evidence for the null hypothesis), and are inconclusive for all subscales in the model with course grades (BF_{10} > 0.333 and < 0.8, favouring the null by 1.25 to 3 times).

[3] In Chapter 2, Version 2 of the SCS1-S was found to have higher internal-consistency and reliability than Version 1. When preforming this regression analysis with only participants who were assigned Version 2, we found that autistic traits did predict performance ($\beta = .220$, $p = .015$). This suggests that the greater reliability of this version may have increased the ability to detect a relationship between autistic traits and programming skill.

Table 4.2. AQ subscales as the predictors of SCS1-Short scores and course grades.

	SCS1					Course grade				
	β	*SE*	*T*	*p*	BF_{10}	β	*SE*	*T*	*p*	BF_{10}
Social skill	.053	.093	0.569	.570	0.250⁻⁻	.131	.093	1.405	.161	0.800
Attention switching	.027	.076	0.351	.726	0.250⁻⁻	.069	.076	0.911	.363	0.471
Attention to detail	.040	.069	0.574	.567	0.250⁻⁻	-.005	.069	-0.078	.938	0.333
Communication	-.013	.092	-0.147	.883	0.250⁻⁻	-.043	.092	-0.465	.643	0.364
Imagination	-.006	.070	-0.090	.929	0.250⁻⁻	-.083	.070	-1.175	.241	0.615

Note: Estimates and *p*-values for the effects of the different AQ subscales in the regression models on SCS1-S scores and course grades. All scores were standardised. * indicates $p < .05$, ** indicates $p < .01$, *** indicates $p < .001$; ⁺indicates $BF_{10} > 3$; ⁻⁻indicates $BF_{10} < .333$.

There were no significant correlations between AQ and any of the measured cognitive skills (see Table 4.3). The Bayes factors show moderate evidence for the null hypothesis for pattern recognition and grammar learning, and weak evidence for the null hypothesis for algebra and logical reasoning. The Bayes factor for vocabulary learning is inconclusive.

Table 4.3. Correlations between AQ score and scores on the cognitive tests.

	Correlation (*r*)	*p*-value	BF_{10}
Pattern recognition	.010	.883	.085⁻⁻
Algebra	.100	.137	.225⁻⁻
Logical reasoning	.127	.058	.204⁻⁻
Vocabulary learning	-.131	.050	.561
Grammar learning	-.029	.670	.092⁻⁻

Note: * indicates $p < .05$, ** indicates $p < .01$, *** indicates $p < .001$; ⁺indicates $BF_{10} > 3$; ⁻⁻ indicates $BF_{10} < .333$.

4.4 DISCUSSION

This study explored the relationship between autistic traits and programming skill in beginner programmers. This knowledge could be useful in identifying students best suited to learning programming. To this end, we measured autistic traits at the start of a 12-week beginner programming course for undergraduate students and examined whether they predicted programming skill acquisition. We also examined whether autistic traits were related to specific cognitive skills at the start of the course (i.e., pattern recognition; algebra; logical reasoning; grammar learning; vocabulary learning).

We found that the students in the current study scored higher on autistic traits than the general population (Ruzich et al. 2015). However, overall autistic traits did not predict programming skill at the end of the course, even when we controlled for previous programming experience. Similarly, no individual subscale predicted programming skill, nor were there correlations between autistic traits and the cognitive skills that may underpin programming.

One possible explanation for these results is that one overall measure of autistic traits (i.e., total AQ scores) may not predict programming skill because only some autistic traits support the successful acquisition of programming. In the context of previous research looking at the Systemizing Quotient (SQ) and the Empathy Quotient (EQ; Wray, 2007; Borzovs et al., 2017), it may be the case that AQ items which load on systemizing behaviours (e.g., items relating to attention to detail) and empathizing behaviours (e.g., items relating to social skill) have the most predictive power for future programming skill compared to those items that capture other autism characteristics. Examining the predictive power of the individual subscales could be one way to explore this, however, this approach is limited given that the current subscales show low test-retest reliability (Austin, 2005; Hurst et al. 2007; Stevenson & Hart, 2017) and questionable construct validity (Austin, 2005; Hoekstra et al., 2008; Hurst et al. 2007; Stevenson & Hart, 2017). Due to these limitations, the subscales are unlikely to have the sensitivity to identify relationships between specific autistic traits and programming skill acquisition. To further tease apart these relationships, a more extensive evaluation of autistic traits is required, so that the predictive value of each specific trait can be examined reliably and validly. The Subthreshold Autistic Traits Questionnaire (SATQ; Kanne et al., 2011) is a potentially suitable instrument for such an analysis. The SATQ was also designed to measure autistic traits but it captures a broader range of autism characteristics than the AQ (Kanne et al., 2011; Nishiyama et al., 2014). It has also been suggested that the SATQ is more sensitive to features of the female autism phenotype given that the AQ is argued to be biased towards the male autism phenotype (Murray et al., 2017; Ruzich et al., 2015). Therefore, validating the current findings with the SATQ could provide evidence for their generalisability.

An alternative explanation of why AQ scores did not predict programming skill acquisition is that autistic traits may be a better predictor of people's 'fit' or preference to pursue a career in programming, rather than their ability to learn and acquire programming skill per se. Coles and Phalp (2016) also suggested this possibility following findings that the difference between SQ and EQ was related to degree choice, but not to programming skill when assessed using course grades. In the current study, the participant population did score above average on autistic traits. Given that, from a theoretical perspective, autistic traits are presumably stable dispositional characteristics, it is possible that these traits made students more likely to pursue training in this domain. The fact that autistic traits did not correlate with the amount of previous programming experience (r = -.087, p = .145) may simply be due to a limited opportunity for these students to gain such experience. In addition, for many students the current course was part of their mandatory study program. Therefore, the higher AQ scores in our sample may be due to people choosing engineering or mathematics majors that have programming prerequisites, rather than a specific choice to learn about programming, or to pursue a career path that involves programming. To disentangle this relationship, future studies should examine autistic traits across diverse groups of novice programmers within student populations and compare those who have enrolled in degrees which do and do not involve programming coursework. One interesting comparison group would be arts and humanity students who have to take programming subjects (e.g., for data analyses). If autistic traits were found to be predictive of course choice, and possibly course enjoyment, the AQ as a questionnaire could be used by individuals and counsellors to assist informed decisions about career pathways and associated vocational training and education.

Finally, it is possible that autistic traits do not predict programming skill in the current study because programming skill was operationalised as performance under academic assessment conditions. These assessments may be confounded with stress and anxiety and may therefore underestimate the degree to which someone learned (and to a variable extent across participants). If success were to be more broadly operationalised using long-term criteria such as employer and employee satisfaction, different results may be found. In this, less immediate but more ecologically valid, context, the personality traits associated with autism (e.g., satisfaction derived from completing repetitive and specialised tasks) may lead employees to be more focused, and motivated – which leads to better long-term performance outcomes. An interesting future direction would be to measure the autistic traits of programmers in the workplace and examine the relationship of this with subjective measures of performance and satisfaction from the perspective of programmers, employers and co-workers.

In sum, the current study suggests that autistic traits are not as reliably related to programming skill as stereotypes suggest. However, there may be specific autistic traits or cognitive strengths associated with autism that have a more direct relationship with programming skill. For example, the cognitive features of autism may relate to programming skill, while personality or preference aspects of autistic traits relate to career choice and workplace performance. These possibilities require further empirical investigation using more specific measures and more diverse samples. There is also a need to explore other outcome measures of 'programming skill' which should include measures of specific skills but also long-term measures of workplace success from both employee and employer perspectives. This line of enquiry will provide more information on the relationship between autism and autistic traits, job performance and satisfaction. This can lead to a wide range of benefits in society, for example by informing careers counselling, destigmatising autism and encouraging the employment of autistic people.