SQ Minus EQ can Predict Programming Aptitude

Stuart Wray

Royal School of Signals, Blandford Forum, UK.
swray@bournemouth.ac.uk

Abstract. Students from an introductory programming class were given several tests in an attempt to establish whether any of their test scores correlated well with their measured programming ability. It was discovered that, when used in combination, the Autism Research Centre's SQ [1] and EQ [2] tests showed a high correlation (r = .67) with a test for programming ability. Individually, SQ and EQ show moderate correlation (r = .44 and r = -.45 respectively) with this programming test. In contrast, for this group of students, Dehnadi and Bornat's test [3] and a self-rank test could not be used to successfully predict programming ability.

1 Introduction

It is well known that some people have difficulty learning to program, while others find it straightforward. However, reliably distinguishing these two populations is problematic, and the search continues for a reliable programming aptitude test which can be taken before a course starts. Wilson and Shrock [4] found three factors significantly contributing to success in a C++ programming class. The factor most predictive of success was "comfort factor", a number derived from questions about participation in classes and labs, anxiety about assignments, perceived difficulty and extent of understanding. However, this factor was a judgement made by students during the course, so could not actually be used for predictive purposes before the course started. The factor next most predictive of success was "math background", defined as the number of semesters of high-school maths the students reported. (This factor could indeed be used for prediction before a course.) Their third predictive factor was attribution of success on the mid-term exam to luck (a negative influence). However, this factor too suffers from the problem of not being available before a course starts.

Dehnadi and Bornat [5] describe a test for programming aptitude, further elaborated in [3], which can be used before a programming course starts. Although this test asks questions about short fragments of code, no previous programming knowledge is assumed, because the test rates students on their ability to construct a consistent theory, rather than the ability to intuit the "correct" answer. This test is reported [3,5] to have good predictive power when used before a programming course starts.

Following these examples, the current author wondered whether any other tests might be able to predict programming ability in advance. In particular, might it be possible that measurements of mild autistic-spectrum tendencies would show a useful

correlation with programming ability? The popular image of an expert programmer is a man (seldom a woman) careless of physical appearance, socially inept, with narrow and intense interests and a peculiarly literal way of interpreting spoken or written statements. While this is a stereotype, there is some truth in it, and all of these characteristics are commonly shared by people with a mild autistic-spectrum condition or Asperger syndrome. This is not fanciful association, since it has been observed that scientists and engineers are more likely to have autistic-spectrum conditions than the general population and that fathers and grandfathers of autistic-spectrum children are more likely to work as scientists and engineers [6].

Only a self-administered test would be practical. However, rather than use a direct measure of autistic tendencies such as AQ [7], it was decided to use a pair of instruments, "Systemizing Quotient" (SQ) and "Empathy Quotient" (EQ). These are designed to be used together and the quantity SQ – EQ has been shown to have a strong correlation with independent diagnoses of Asperger syndrome and other autistic-spectrum tendencies [8].

The Autism Research Group at Cambridge produced these two personality questionnaires based on the assumption that autistic tendencies can be split into two aspects [9], namely: the ease with which an individual understands systems of objects (SQ) and the ease with which they understand emotions of people (EQ). Experiments confirm that although there is a very broad overlap, on average men have a significantly higher SQ and a significantly lower EQ compared to women. Autistic-spectrum people, who are mostly men anyway, on average have an even higher SQ and lower EQ than the general population of men [8].

An experiment was therefore designed to test the hypothesis that SQ and EQ scores, either together or individually, would be strongly correlated with a measure of programming ability. The opportunity was also taken to evaluate some other potential predictors of programming ability, including Dehnadi and Bornat's test.

2 Method

Students on the BSc(hons) course in Telecomunications Systems Engineering at the Royal School of Signals were invited to participate in the experiment. All 19 chose to do so. All the students were male.

Five tests were used, which are described further below: programming, self-rank, SQ, EQ and Dehnadi-Bornat. The programming test had already been completed by the students in June 2006, before this experiment was conceived. The self-rank, SQ, EQ and Dehnadi-Bornat tests were completed by the students in November 2006, some five months after they had finished the programming module of their course.

Students took both the June and November tests as a group. In November, when each student had finished, they handed in their bundle of completed tests and left the lab. Hand-in times were not noted, but the hand-in order was preserved.

2.1 Programming test

The test consisted of 10 questions on the output produced by a short program fragment in the Python programming language. It was designed to test understanding of function and method calls in an object-oriented program. Cronbach's alpha for the programming test on this group of students was 0.78. A copy of this test is in Appendix A.

2.2 Self rank

The self-rank sheet asked the subjects to indicate their perceived rank in their class by circling the outline of a person in a line, as shown in figure 1.
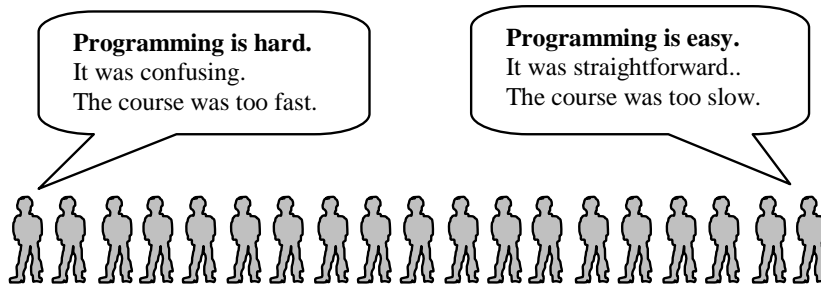
Fig. 1. The line of figures in the self rank sheet

2.3 SQ and EQ

The SQ test is described in [10] and the EQ test in [11]. The tests themselves are available online from the Autism Research Centre at www.autismresearchcentre.com [1], [2]. Each test consists of 60 questions in the form of a statement followed by four alternative answers. All questions are forced-choice and have the same alternatives: strongly agree, slightly agree, slightly disagree, strongly disagree. For example, SQ question 33 is as shown in figure 2.

33.	If I were buying a stereo I would want to know about its precise technical features.	strongly agree	slightly agree	slightly disagree	strongly disagree

Fig. 2. Example question from the Systemizing Quotient (SQ) test.

The tests are designed to be self-administered, and are easy to complete in a short time and easy to score. Both SQ and EQ have 60 questions, 40 of which assess systemising or empathising respectively, with the remaining 20 questions being fillers which are not scored. Of the 40 scoring questions, half of them score one mark for "slightly agree" and two marks for "strongly agree", but nothing for "disagree". The

other half score one mark for "slightly disagree" and two marks for "strongly dis-agree", but nothing for "agree". Thus the range of potential scores for each test is from 0 to 80.

2.4 Dehnadi-Bornat test

The Dehnadi-Bornat test is described in [3]. The test itself is available online from http://www.cs.mdx.ac.uk/research/PhDArea/saeed [12].

The test consists of 12 increasingly complex questions concerning the values of variables following a sequence of assignments. No previous programming experience is assumed: the purpose of the test is to evaluate the subject's ability to invent a con-sistent model for what the symbols could mean, not to test their ability to guess what they do mean in any particular programming language.

For example, the first question is shown in figure 3. Subjects are classified as "consistent" or "inconsistent", depending on whether 80% or more of their answers correspond to the same theory of variable assignment.

| **1. Read the following statements and tick the box next to the correct answer in the next column.**

`int  a = 10;`
`int  b = 20;`

`a = b;` | **The new values of a and b are:**

☐ a = 10   b = 10
☐ a = 30   b = 20
☐ a = 0   b = 10
☐ a = 20   b = 20
☐ a = 0   b = 30
☐ a = 10   b = 20
☐ a = 20   b = 10
☐ a = 20   b = 0
☐ a = 10   b = 30
☐ a = 30   b = 0

Any other values for a and b:

a =   b =
a =   b =
a =   b = |

Fig. 3. Example question from the Dehnadi-Bornat test.

3 Results

3.1 Programming test

As can be seen from the graph, this test should perhaps have been harder, since there is a cluster of marks at the high end. The results shown in figure 4 are consistent with the expected "two hump" distribution noted in [5].
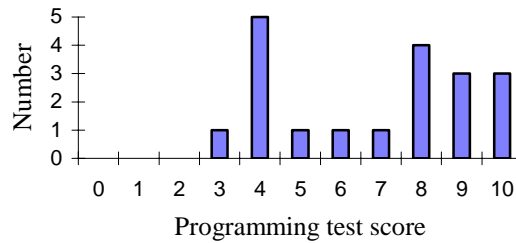

Fig. 4. Number of students attaining each programming test score

3.2 Self rank

Because the programming test had been taken some five months previously, students were asked to rank themselves in the November tests. The results, shown in figure 5 show only a small correlation ($r = 0.16$) with their earlier programming test scores.
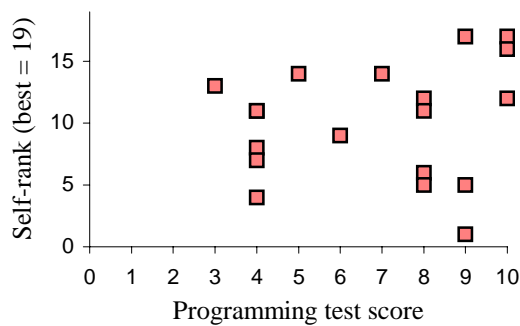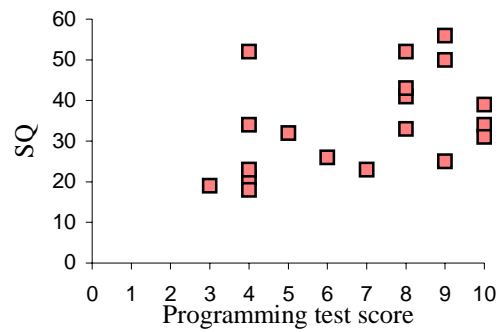

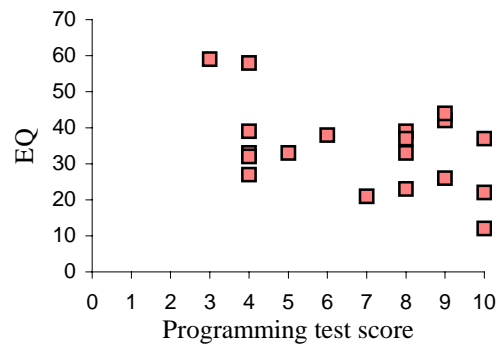**Fig. 5.** Self-rank position in class plotted against programming test score.

3.3 SQ and EQ

In figure 6, plotting SQ against programming test score we see a moderate correlation ($r = 0.44$, $p = 0.056$).

Fig. 6. Systemizing Quotient (SQ) plotted against programming test score.

Plotting EQ against programming test score, in figure 7, we see a moderate negative correlation ($r = -0.45$, $p = 0.052$).

Fig. 7. Empathy Quotient (EQ) plotted against programming test score

The SQ and EQ tests both have values in the range 0 to 80, with fairly similar distributions. They were designed to be used in conjunction and the value SQ – EQ has been shown to be significantly different for males, females and Asperger syndrome individuals [8]. Plotting SQ – EQ against programming test score, in figure 8, we see a high correlation ($r = 0.67$, $p = 0.002$).
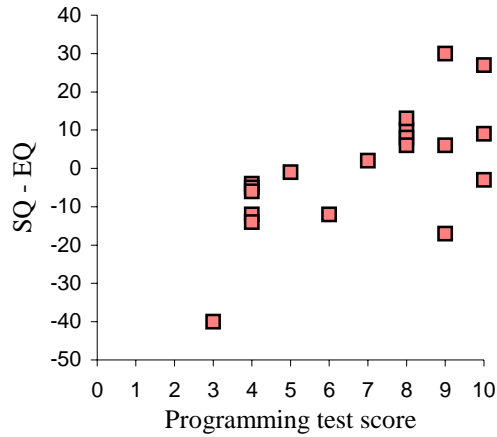
Fig. 8. SQ – EQ plotted against programming test score.

For this class, mean programming test score was 6.8 (SD = 2.4), median 8; the mean SQ was 34.3 (SD = 12.0), median 33; the mean EQ was 34.5 (SD = 11.7), median 33; the mean SQ – EQ was -0.2 (SD = 15.9), median -1. In contrast, for the general population of males the mean SQ is 30.3 (SD = 11.5) and the mean EQ is 41.8 (SD = 11.2) [10, 11]

3.4 Dehnadi-Bornat test

The subjects gave surprisingly consistent answers measured by the criteria of the test, which is why the results of this test are inconclusive. There is not enough difference between the answers to divide the subjects into "consistent" and "inconsistent" groups. All students chose the "correct" M2 theory of variable assignment, and 13 of them were completely consistent. The remaining 6 were within the range considered adequately consistent by the standards of the marking criteria [3], i.e. greater than 80% of answers selecting the same theory.
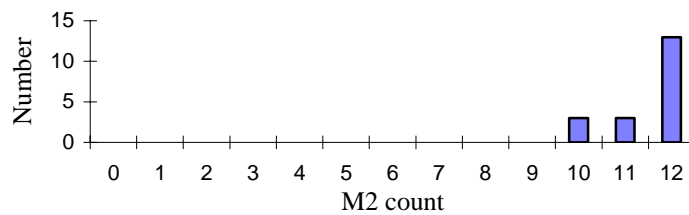

Fig. 9. Number of students attaining each M2 count on Dehnadi-Bornat test.

3.5 Hand-in order

Although the students were not timed, the hand in order of tests was preserved and recorded. If we plot programming test score against hand-in order we see a very interesting effect, as shown in figure 10. It is as though the subjects fall into three groups, who we might informally describe as "quick" (top left), "thorough" (top right) and "neither" (middle).
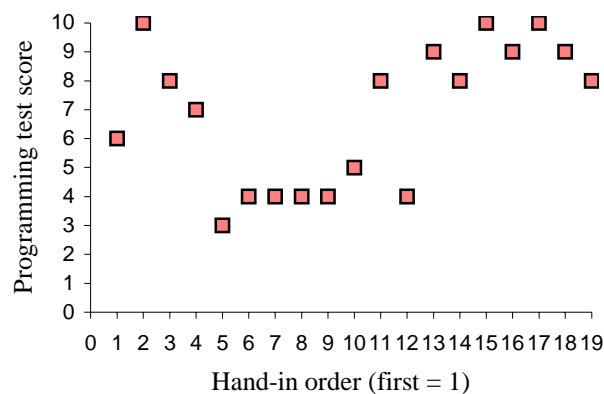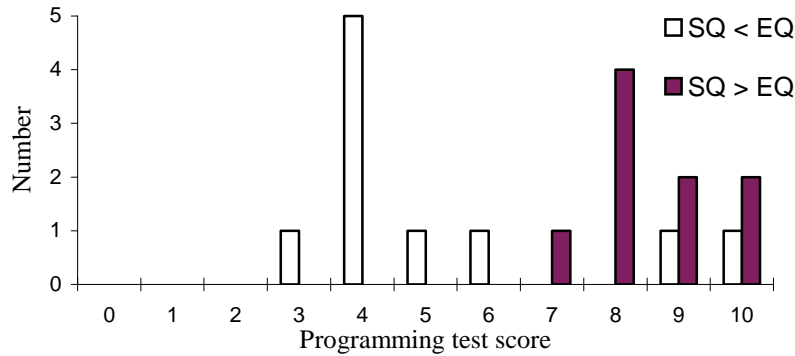

Fig. 10. Programming tests score plotted against hand-in order.

4 Discussion

Although SQ – EQ is highly correlated with the programming test score, could it in practice be used to predict programming ability? It happened that the subjects of this experiment had several months previously taken a programming test as part of the software module of their course, before the rest of the experiment was conceived. The SQ – EQ score was therefore in this case "predicting" something which had already happened. However, it seems reasonable to suppose that had the SQ or EQ tests been administered six or twelve months earlier, they would have given very similar results. (This assumption will be tested directly in further work on another cohort of students.)

If SQ and EQ scores were stable over this period, then SQ – EQ could have been used to predict programming ability before the course started. For example, figure 11 divides the students into those with SQ > EQ and those with SQ < EQ.

Fig. 11. Number of students with SQ < EQ and SQ > EQ plotted against programming test score.

It was somewhat disappointing that the Dehnadi-Bornat test showed such inconclusive results, given its successful application both before and after programming courses at other establishments. Two plausible explanations present themselves: firstly, it is possible that the class of students simply were naturally very good at the kind of activity measured by the Dehnadi-Bornat test. They had secured places on the course through a rigorous and highly contested selection procedure, and are perhaps not typical of the populations of students found on many university programming courses. A second explanation is that since the test was administered after the programming course, perhaps the teaching had been so effective that they had all retained its lessons, at least as far as variable assignment is concerned. (Although flattering, the second explanation seems less likely.)

Lastly, let us consider why SQ – EQ might be so highly correlated with programming ability. People who have an SQ much lower than EQ will in everyday life prefer interacting with other people, who they find intuitively easy to understand compared with mechanisms and machinery, which they find mysterious, cold and soulless. In contrast, people with SQ much higher than EQ will in everyday life prefer to deal with orderly systems of objects, which they find intuitively easy to understand compared with people, who they find fickle, confusing and worrisome. Each will tend to engage more in the sorts of activities with which they are comfortable and less in those activities which they find upsetting.

As the saying goes, "practice makes perfect". This is confirmed by recent research on expertise [13] which shows that, across a wide variety of fields, a very large part of the abilities of experts is due to the sheer quantity of time that they have spent in "effortful study", practicing things which are only just within their grasp. Initial ability plays a part, but a rather smaller part than is often assumed. Mastering a programming language requires not just some initial ability, but also the inclination to put in the necessary hours of effortful study. Those people with high SQ – EQ will find that this effortful study "goes with the grain", that it is the kind of activity which they find attractive, familiar and satisfying. Those with a low SQ – EQ will find the effortful study very hard work, because it is in an area in which they have had little practice since they find it less rewarding than activities involving people.

In fact, even if they are willing to make the effort, their unfamiliarity with this type of activity may require an entirely different approach, and "effortful study" may involve very much smaller steps than for the other group. If larger steps are attempted, they may find the material not "just within their grasp", but far outside their grasp. Like many people trying to learn how to program, they may be completely baffled.

5   Further Work & Conclusions

Some of the tests (SQ, EQ, Dehnadi-Bornat) were also taken in September 2006 by all 17 students of a different class, before they were taught the programming module of their course. These tests have been put aside, not yet scored, so that teaching of this course would be blind to the results of the tests. The programming module of this course will not be complete until the summer of 2007, and it is planned to re-test SQ and EQ later in 2007 so as to establish the stability of these measures. Results from this class will be reported on another occasion.

The SQ test used in this paper has been superseded by a revised version (called SQ-R), described in [14]. The revisions to SQ-R mainly seek to make the systemizing questions more relevant to systemizing women, and less slanted to stereotypical male interests. Since all the students in the current experiment were male, it is not likely that this bias would have had any noticeable effect. However, future work should use the new SQ-R test. Furthermore, rather than using SQ – EQ, it would be preferable to use the measure D defined in [14], which is the normalised difference between SQ-R score and EQ score.

In conclusion, we have seen that SQ – EQ has considerable predictive power concerning programming ability. Since SQ and EQ are very straightforward to set and score, they offer an effective way to assess aptitude for programming prior to a taught course.

Acknowledgements

References

1. *"Cambridge Personality Questionnaire" (SQ test).* Autism Research Centre, Cambridge. Available from: http://www.autismresearchcentre.com/tests/sq_test.asp (2003).
2. *"The Cambridge Behaviour Scale" (EQ test)* Autism Research Centre, Cambridge. Available from: http://www.autismresearchcentre.com/tests/eq_test.asp (2003)
3. Dehnadi, S.: Testing Programming Aptitude. Proc. PPIG16, Brighton, UK. (2006)

4. Wilson, B. & S. Shrock: *Contributing to success in an introductory computer science course: a study of twelve factors.* Proc. ACM SIGCSE'01 184-188 (2001)

5. Dehnadi, S. and R. Bornat. *The camel has two humps.* in *Little PPIG* 2006. Coventry, UK. Available from http://www.cs.mdx.ac.uk/research/PhDArea/saeed.

6. BBC: *Scientific brain linked to autism.* BBC, London. Available from: http://news.bbc.co.uk/1/hi/health/4661402.stm (2006)

7. S. Baron-Cohen, et al.: *The Autism Spectrum Quotient (AQ): Evidence from Asperger Syndrome/High Functioning Autism, Males and Females, Scientists and Mathematicians.* Journal of Autism and Developmental Disorders 31:5-17 (2001)

8. Goldenfeld, N. et al.: *Empathizing and Systemizing in Males, Females and Autism.* International Journal of Clinical Neuropsychiatry 2, 338-345 (2005).

9. Baron-Cohen, S.: *The Essential Difference: Men, Women and the Extreme Male Brain.* Penguin, London (2004)

10. Baron-Cohen, S., et al: *The Systemising Quotient (SQ) : An investigation of adults with Asperger Syndrome or High Functioning Autism and normal sex differences.* Philosophical Transactions of the Royal Society, Series B, Special issue on "Autism : Mind and Brain" 358:361-374 (2003)

11. Baron-Cohen, S. and S. Wheelwright: *The Empathy Quotient (EQ). An investigation of adults with Asperger Syndrome or High Functioning Autism, and normal sex differences.* Journal of Autism and Developmental Disorders 34:163-175 (2004)

12. Dehnadi, S. & R. Bornat: *"Multiple-choice test for the Research Project" (Dehnadi test).* School of Computing, Middlesex University, UK. Available from: http://www.cs.mdx.ac.uk/research/PhDArea/saeed/test(week-0).doc (2006)

13. Charness, N. et al. (Eds): *The Cambridge Handbook of Expertise and Expert performance.* Cambridge University Press. (2006)

14. Wheelwright, S., et al: *Predicting Autism Spectrum Quotient (AQ) from the Systemizing Quotient-Revised (SQ-R) and Empathy Quotient (EQ)* Brain Research 1079:47-56. (2006)

Appendix A

This is the programming test, written in Python. Students are asked what is printed in the last 10 lines of the program.

```
class Alberto:
    def G(self, x):
        if x == 4:
            self.F()
        else:
            F()

    def F(self):
        print "Alberto's F"

    def H(self):
        H(self)

class Beryl(Alberto):
    def H(self):
        print "Beryl's H"
```

```
class Chris(Beryl):
    def H(self):
        print "Chris's H"

    def F(self):
        print "Chris's F"

class Debby(Beryl):
    def F(self):
        print "Debby's F"

class Ernesto(Alberto):
    def F(self):
        print "Ernesto's F"

class Florence(Ernesto):
    def G(self, x):
        if x == 0:
            self.F()
        else:
            F()

def F():
    print "outer F"

def H(x):
    x.F()

a = Alberto()
b = Beryl()
c = Chris()
d = Debby()
e = Ernesto()
f = Florence()
x = c

# What is printed by each of the following lines

a.G(4)        # (1)
b.H()         # (2)
d.G(4)        # (3)
d.F()         # (4)
f.G(4)        # (5)
f.F()         # (6)
a.H()         # (7)
H(a)          # (8)
H(b)          # (9)
H(x)          # (10)
```