TA Weinberg

David Greathead[1], Budi Arief[1] & Joey Coleman[1]

[1]School of Computing Science
University of Newcastle
Newcastle Upon Tyne, NE1 7RU
+44 191 2226384
{david.greathead, L.B.Arief,
J.W.Coleman}@ncl.ac.uk

ABSTRACT
This paper summarizes the main areas of research within TA Weinberg, namely an investigation into code review ability and personality; an investigation into code comprehension and personality; and an investigation into the results from an online programming competition and personality.

With regard to code review, it was discovered that there is a relationship between code review ability and personality type. The code comprehension and programming competition research is ongoing.

Keywords
Personality, MBTI, code-review, code comprehension, programming competition, diversity.

1. INTRODUCTION
There is a tendency in DIRC to focus on the dependability of the *created* systems (including, of course, the human system around the technical components). But we clearly state, in various places, that DIRC is also concerned with how to increase the dependability of the *creation* process. System design is obviously a human process and DIRC's mixture of different disciplines offers us the opportunity to look at factors which might increase the dependability of created systems by looking at the *creation* process.

Little empirical research has been carried out examining the links between personality and programming and, of the research that has been carried out, most treats programming as a single monolithic process. This fails to reflect the fact that different stages of the software development process, such as design, coding, testing etc. are markedly different in nature and therefore people with differing strengths might be better suited to the different tasks.

This TA is a continuation of the work started in PA8 — *Effective Collaboration in Design* which was initially inspired by Weinberg's classic work, begun in the early 1970s on programming and psychology [18]. Weinberg conjectured that different aspects of the software development process require different abilities and that factors such as personality may, to some extent, be used to predict performance in the various stages.

2. PSYCHOLOGY AND PROGRAMMING
Cooperation between these two sciences was proposed by Weinberg in 1971 [18], and since then many other theorists have tried to look at some of the psychological aspects involved in the programming process.

In his book, Weinberg [18] presented evidence supporting his hypotheses regarding psychology and the programming process and, while this information tended to be anecdotal in nature, it was nonetheless compelling. He covered such diverse areas as personality, intelligence, motivation, training, experience, as well as considering what makes a good program and the various aspects of programming in groups or teams.

3. VARIATIONS IN PERFORMANCE
Some theorists noted that there were surprisingly large variations in individual productivity and accuracy while executing parts of the software development process. Boehm [2] described the variation as being between a factor of 10 and a factor of 30.

There has also been some speculation that there is some other factor, some 'innate human trait' [16] which may explain some of the large variations observed. This variation has been observed even between those with the same programming background, suggesting that there are factors beyond normal teaching programmes which are influencing ability in areas such as debugging.

It has also been suggested by Weinberg [18] that the type of task may also be responsible for some of the variation observed. That is, as different types of task require different sets of skills, it stands to reason that different people will possess different skills, resulting in the fact that some people are better suited to certain types of task. For example, an individual could be an excellent program designer, but lack the skills required to effectively debug a program, and vice versa. Neither person is better than the other, simply better suited to certain kinds of task.

Various studies have been carried out examining individual characteristics influencing performance at work [7-12]. These individual characteristics can be assessed using some of the many personality tests available. Some personality tests are biased towards a clinical sample and are aimed at aiding diagnosis of disorders (such as the Minnesota Multiphasic Personality Inventory, or MMPI) while others tests are not aimed at diagnosis (such as the Myers Briggs Type Indicator or MBTI and 16

Personality Factors - 16PF). Many of the studies carried out within personality and computing science were done with the MBTI.

Bishop-Clark [1] analysed the personality types of college students in a programming class and made some suggestions about the right type for each phase of the programming process due to the fact that, according to Weinberg [18] each phase is distinct, requiring different types of people to work on them. Recently, Capretz [3] analysed the personality types of software engineers. He found that while the MBTI describes 16 personality types, 24% of the engineers were all of the same type (ISTJ) which characterises the person as quiet, serious, concentrated, logical and realistic. A person presenting this type is technically oriented, does not like dealing with people and when working prefers to deal with facts and reasons. However, there is no indication whether a programmer with such characteristics performs better than programmers with other personality characteristics. Other personality types were given, but again there was no evidence which could reinforce the idea that such types would be better at certain phase of programming than others.

4. MBTI

The MBTI is a popular personality assessment [1, 5, 6, 17]. Originally based on Jung's theory of psychological types it included three bipolar factors or dimensions, these being; extroversion/introversion (EI), sensing/intuition (SN), and thinking/feeling (TF). Afterwards, Myers and Briggs added another pair of characteristics relating to judgment and perception (JP).

The MBTI exhibits acceptable internal consistency; 'the estimates of internal consistency reliabilities for the continuous scores of the four MBTI scales are acceptable for most adult samples' [13, p.165-169]. The MBTI also shows consistency over time with good test-retest reliability. Changes in letter type normally only occur with one letter between tests, and then only if the strength of the preference associated with that letter was low [13, p.170-171].

4.1 The dimensions

Beyond the everyday understanding of the words extroversion and introversion, Myers [15] explained that the EI dimension is related to the way people tend to "recharge their energy". That is, extroverts will focus their attention on other people through the external environment and feel more energetic after interacting with other people in a social setting, while the opposite would be true of introverts who would be drained after such interactions and would prefer to spend time with close friends or family (or alone) in an internal environment. This obviously has an influence on general career choice as this dimension has an impact on the type of job a person would consider. So, extroverts would be drawn to jobs which involve interacting with people while introverts prefer to work with impressions and ideas. However interest in jobs is mainly determined by the dimensions SN and TF.

The sensing/intuition dimension is concerned with how people gather information from the world. In the case of SN, the concern is whether information is gathered through the five senses in a concrete manner, purely accepting what is directly observed, or through intuition using imagination and inspiration. The dimension of thinking/feeling however, is more related to the manner in which people make decisions. As the name thinking/feeling suggests, decisions can be made in one of two ways – either following some logical sequence or basing decisions on a people-centred opinion considering feelings over logic.

These two dimensions have an influence on career choice in that they impact not only on how satisfied people are by their career choice, but also how they are initially drawn to their choice [14]. For instance, it seems reasonable to assume that a psychological counsellor would be more likely to have a feeling bias rather than a thinking bias. It is again important to emphasise that one type is not regarded as being better than another - people simply need to be aware of how to take advantage of their type and then work with their preferences and their abilities.

The way a person lives their life on a daily basis is influenced by the judging/perceiving dimension. People with a judgement preference tend to plan the events in their lives as much as possible while those with a perception bias are more likely to be spontaneous and adaptable in their everyday lives. This preference tends to exhibit itself in the manner in which a person lives their working life insofar as judgement-based people will tend to have specific self-imposed schedules and deadlines while people more towards the perception end of the scale will deal with issues as they arise and feel more comfortable without a strongly defined timetable.

The MBTI is intended to be used with "normal individuals in counselling and within organizations" [5]. Smither [17] explains that such an instrument is not a useful tool in recruiting people. In fact the MBTI is a useful tool in relocating people in organizations, i.e. selecting current employees for a special task and to improve work communication/interaction. As such it is a popular personality measure within industry, being familiar to both employers and employees alike.

In the second edition of his book, Weinberg [18] states that he would have written a completely different chapter about personality if he had known the advantages of the MBTI. He adds that this assessment is dealing "with normal personality differences" (19, p.8i) as some personality tests are related to mental disorders (such as the MMPI), i.e. they do not see the person as being normal.

5. THE CODE-REVIEW STUDY

5.1 Methodology

5.1.1 Apparatus

In order to collect the relevant information in this study, two different instruments were employed, these being the Myers-Briggs Type Indicator (MBTI) and a code-review task.

The 'MBTI Step 1, European English Edition' was used to assess participants' personality. This is an 88 item forced response personality inventory, which returns scores on four bipolar scales. Four letters are returned to indicate the type of preference, for example an individual would have either I or E as one of their letters (for Introversion or Extroversion), with a corresponding value indicating the strength of this preference. Similar scores are obtained for Sensing (S) or iNtuition (N); Thinking (T) or Feeling (F); and Judging (J) or Perceiving (P).

The code-review task consisted of four pages (282 lines) of Java code, which had been written by an experienced programmer. The program was a pattern search program which would operate on an ASCII file. After being examined for the presence of bugs, 16 semantic bugs were inserted into the code by the programmer. The program was accompanied by a two page manual and API, including an example of the output the debugged code would produce when executed, and the ASCII file which was used for the example. A title page provided instructions on how to complete the task. One of the participants returned the task essentially blank and as such was excluded from certain statistical tests.

Java was chosen as the language as it was the only language that all of the participants had knowledge of; it being the language which was taught during the first year at the university.

5.1.2 Participants

Sixty-four participants completed both stages of the study. The participants were all undergraduate students from Newcastle University in the United Kingdom. While participants were unselected for age and sex, the majority (81%) were male and 77% were aged 19-21 due to the nature of the population. Participants were all awarded extra marks in their course for each aspect of the research in which they took part. Three prizes per programme of study were also awarded (of £10, £20 and £30) for the participants who scored more highly on the code review task. This was in order to encourage participants to apply themselves to the task.

5.1.3 Procedure

The two instruments were administered on separate occasions, in regularly scheduled, one-hour lecture slots. The code review task also occupied part of the adjacent lecture slot. At both stages of the research, participants were reassured that their data would not be used in such a way that they could be individually identified.

Participants were given one and a half hours to complete the code review task. They were informed that this was an individual task and were asked not to talk to one another. They were also asked to spread themselves out as much as possible in the lecture theatre. As well as being reminded that they would receive extra course credit for their participation, the participants were informed that there would be three prizes for each of the programmes of study, awarded to the highest scoring participants in each programme. They were also given additional information about the task in that all of the errors in the code were semantic errors, and that they did not have to correct the errors, only identify them in some way. They were not informed how many errors there were in total.

In a separate session, participants were allowed up to one hour to complete the MBTI questionnaire and were free to leave the lecture theatre once they had finished. Participants were also offered an individual feedback session if they desired. The researchers were available to answer any questions participants may have had with regard to the questionnaire.

5.2 Results

In order to examine the possible links with MBTI type and code review ability (score), a number of correlations were carried out. The results are presented in Table 1 below[1].

Table 1 - Correlations between

MBTI type and code review score

	E	**S**	**T**	**J**
Code Review correlation	**-0.197**	**-0.251**	**0.197**	**0.000**
Sig (2-tailed)	**0.121**	**0.047**	**0.122**	**0.998**

As can be seen, the only significant correlation was that between the Sensing scale and the code review score. High scores on the Sensing scale represent an individual with a high *sensing* preference, while low scores represent an individual with a high *intuitive* preference. As this is a negative correlation, it indicates that people who are more *intuitively* inclined performed significantly better on the code review task than *sensing* types. There were minor, non-significant correlations with the *extroversion* and *thinking* scales but no correlation whatsoever with the *judging* scale.

Further examination of the data revealed an interesting interaction between two variables. If participants are grouped according to two-letter types, performance on the code review task can be compared according to these types. The most notable differences using this method came with the SN and TF scales. Comparison of the mean code review scores shows that the NT students scored 9.10 as compared to the non-NTs who scored 6.14 on average. This illustrates that the NT individuals were able to perform better on average than non-NT people. The mean scores for these four types are shown in Table 2.

Table 2 - Mean code review score by SN/TF types.

	F	T
N	8.71	9.10
S	4.27	6.62

As can be seen from the table, the most marked difference was between NT participants and SF participants, the NT participants scoring on average more than twice a well as the SF participants. A t-test comparing NTs with non-NTs yielded a significant result, illustrating that NTs were better at the task than non-NTs (1-tail sig = 0.039, t = 1.801, df = 61).

5.3 Discussion

It would appear that Weinberg's hypothesis was correct in that people of a certain personality type performed better at one aspect of the software development process. This being the case, further

[1] For more information about correlations and other statistical analysis adopted in this study, please see a book on research methods, such as Robson, C., *Real world research : a resource for social scientists and practitioner-researchers*. 2nd ed. 2002, Oxford, UK ; Madden, Mass.: Blackwell Publishers. xxii, 599.

research was recommended in order to more precisely analyse what these differences are. The results of this were the follow up studies of the Code Comprehension and Online Programming Competitions studies (see below). In addition, it may be advantageous for software companies to consider the strengths of their employees when assigning tasks in the workplace. If some people, for whatever reason, are better able to perform code review tasks than others then it would seem prudent for software companies to capitalise on the strengths of their employees, and consider employees perhaps previously overlooked for this particular task. This, in turn, would lead to the creation of more dependable systems.

It seems clear that there are mental processes involved in software development which are not as yet, fully understood due to the fact that software engineering in itself is still in its relative infancy. If, as may well be the case, certain types of people are better at certain tasks (other than code-review) then utilising these people for these tasks will lead to fewer errors in the software creation process, and thus produce more dependable systems. In addition, if the development of a model illustrating the mental processes involved in the various tasks can also be produced then this too would lead to a better understanding of the practical processes involved and ultimately in the production of more dependable methods for software development, and consequently more dependable software.

A more detailed analysis of the code-review study can be found in [4].

6. THE CODE COMPREHENSION STUDY

In order to gain a greater understanding of the mental processes involved in the creation of a software product, it was deemed necessary to examine some other aspect of software development. Given the difficulty some of the participants had with the code-review task, the idea of measuring an individual's ability to understand a piece of code was considered. An experiment was conceived whereby participants would be given a piece of Java code in addition to the MBTI personality questionnaire. The participants would be required to demonstrate their understanding by answering a number of multiple choice questions concerning the function of a particular section of code, or to state what would happen if the program were run with a particular set of variables.

To this end, a piece of Java code was selected, which was sufficiently long and intricate so that gaining a proper understanding of its function would not be a trivial task. The program in question is a simulator for a number of lifts in a building, with a number of variables which can by set by the user. For instance, the user may decide that there are four lifts and fifteen passengers scattered throughout ten floors, who will decide to push the call buttons at random intervals within a user specified timeframe. The lift simulator then uses a random seed to distribute the passengers. Once the simulation is started, the program measures the waiting and travel times for each of the passengers.

In addition to simply understanding what certain sections of the code are for, participants will be required to answer specific questions on the program, such as 'where do the lifts rest?' The format for these questions will be multiple-choice (with a space for comments) in order that marking can be carried out for large numbers of participants with ease and accuracy. The generation of these questions is now underway. In a further test, the participants will be required to say how they would cope with modifying the code under certain circumstances. That is, they will be given a particular set of problems and must say how they would implement these changes.

In addition to the code comprehension task itself, participants will be given the MBTI to complete in order to assess if the same results observed with the code-review task are indicative of good performance on this task.

In addition to measuring the normal ability to understand the code, the influence of comments will also be assessed. That is, there will be two different versions of the code utilised for this between groups experiment, one with few comments present and one with more comments present, particularly concerning assertions at relevant points in the code. While it is generally anticipated that the presence of additional comments will be universally beneficial to the participants, the question if how much these comments aid comprehension, i.e. just how important are the comments in aiding code comprehension?

MBTI results will also feature with regard to comments in that it will be investigated as to whether certain types of personality benefit more from the presence or otherwise of these comments.

The whole task will be repeated with participants of varying levels of experience, these being second year undergraduate students, third year undergraduate students and a group of more experienced programmers, preferably from an industrial background. With these participants, it will be possible to examine the extent to which experience matters when attempting to understand a piece of code. It will also be possible to examine whether or not more experienced programmers rely more or less on the aforementioned comments.

7. THE ONLINE PROGRAMMING COMPETITION STUDY

The third aspect of TA Weinberg is that of the online programming competition. This study comprises of generating an online questionnaire form for participants to complete (which has now been done), including a link to a pre-existing personality questionnaire similar to the MBTI. Permission to use this personality questionnaire has now been obtained. Initially, one programming problem from the Valladolid online programming competition was selected. The individuals who attempted to solve this problem will be sent an e-mail with a link to the form, inviting them to take part in the research. The results from the online form and the personality questionnaire will be compared to the solutions they submitted to the programming competition. Specifically, the types of errors participants made on their early attempts at solving the problem have been analysed and will be compared to personality type, educational level, etc. in order to establish any possible links. This analysis will be mostly exploratory in nature and will potentially generate interest in a much larger study in the future encompassing the thousands of participants in this particular programming competition.

8. CONCLUSIONS

TA Weinberg most directly relates to the Diversity research theme in that it is concerned with the impact of different personality types on the process of developing software. A factor which is often overlooked is that the process of creating a piece of

software can be directly related to the type of person involved. If, by examining some of the different factors concerning these people it may be possible to shed some light on the mental processes involved, then this should in turn lead to the ability to build more dependable systems. If it is the case that NT type people perform better through all of the tasks involved in TA Weinberg then the question of what makes them better can be addressed. If, on the other hand, it transpires that some other type is better for code comprehension, and yet another type has the better performance at the online competition this is equally interesting.

However, the ultimate aim of this thrust of research (albeit perhaps not within the scope of TA Weinberg) is to examine groups of people rather than individuals and how they interact when designing and building a piece of software. Then, will the issue of diversity of personality truly become apparent.

9. ACKNOWLEDGMENTS

Thanks go to all of those involved with TA Weinberg and the production of this document. This work was carried out under the EPSRC-funded DIRC project. Special thanks go to Alessandra Devito da Cunha who worked closely with the members of TA Weinberg in the earlier stages of the work.

10. REFERENCES

[1] Bishop-Clark, C. and D. Wheeler, The Myers-Briggs Personality Type and its Relationship to Computer Programming. Journal of Research on Computing Education, 1994. 26(3): p. 358-70.

[2] Boehm, B.W., Software engineering economics. Prentice-Hall advances in computing science and technology series. 1981, Englewood Cliffs, N.J.: Prentice-Hall. xxvii, 767.

[3] Capretz, L.F., Personality types in software engineering. International Journal Human-Computer Studies, 2003. 58: p. 207-214.

[4] Devito Da Cunha, A., "The Myers Briggs Personality Type as a Predictor of Success in the Code-Review Task." MPhil Dissertation, University of Newcastle, 2003.

[5] Devito, A.J., Review of Myers-Briggs Type Indicator, in The Ninth Mental measurements yearbook, J. Mitchell, Editor. 1985, Lincoln, Neb. p. 10301032.

[6] Edwards, J.A., K. Lanning, and K. Hooker, The MBTI and Social Information Processing: An Incremental Validity Study. Journal of Personality Assessment, 2002. 78(3): p. 432-450.

[7] Furnham, A. and T. Miller, Personality, Absenteism and Productivity. Personality & Individual Differences, 1997. 23(4): p. 705-707.

[8] Furnham, A., C.J. Jackson, and T. Miller, Personality, learning style and work performance. Personality & Individual Differences, 1999. 27: p. 1113-1122.

[9] Furnham, A., et al., Do personality factors predict job satisfaction? Personality & Individual Differences, 2002. 33(8): p. 1325-1342.

[10] Furnham, A., L. Forde, and K. Ferrari, Personality and work motivation. Personality & Individual Differences, 1999. 26: p. 1035-1043.

[11] Furnham, A., Personality and individual differences in the workplace: Person-organization-outcome fit. 2001.

[12] Furnham, A., Personality at work : the role of individual differences in the workplace. 1994, New York: Routledge. xxiii, 423.

[13] Myers, I.B. and M.H. McCaulley, Manual : a guide to the development and use of the Myers-Briggs Type Indicator. 5th ed. 1985, Palo Alto, Calif.: Consulting Psychologists Press. xx, 420.

[14] Myers, I.B. and P.B. Myers, Gifts differing: understanding personality type. 1995, Palo Alto, Calif.: Davies-Black Pub. 228.

[15] Myers, I.B., Introduction to type : a description of the theory and applications of the Myers-Briggs Type Indicator. 12th ed. 1990, Palo Alto, CA: Consulting Psychologists Press. 31.

[16] Pressman, R.S., Chapter 19 - Software testing strategies, pp.654-658, in Software engineering : a practitioner's approach. 1992, McGraw-Hill: New York. p. xxi, 793.

[17] Smither, R.D., The psychology of work and human performance. 3rd ed. 1998, New York: Longman. xviii, 590.

[18] Weinberg, G.M., The psychology of computer programming. 2nd ed. 1998, New York,: Van Nostrand Reinhold. xv, 288.