The Analysis of a Novel Computational Thinking Test in a First Year Undergraduate Computer Science Course

Aidan Mooney

James Lockwood

Maynooth University

Abstract

In Ireland, Computer Science is not yet a state examination subject. In recent years, steps to include it have been taken - it was introduced as a Leaving Certificate subject in the academic year of 2018-19 on a pilot basis and will be examined for the first time in 2020 (O'Brien, 2017). Prior to this, the only Computer Science course offered at second level was a Junior Certificate Coding short course (NCCA, 2017). Research shows that an early introduction to computing is an advantage for students. It can build confidence in dealing with complexity and with open-ended problems (Yevseyeva &Towhidnejad, 2012). Problem-solving skills can be extended and transferred as reported by Koh et al. (2013) and students' analytical skills can be improved according to Lishinski et al. (2016) and Van Dyne and Braun (2014). It has been shown by Webb and Rosson (2013) that students' self-efficacy for computational problem solving, abstraction, debugging and terminology can be increased. It has also been found that teaching Computational Thinking can provide a better understanding of how programming is about solving a problem (not just coding) and that it can improve female students' attitudes and confidence towards programming (Davies, 2008). One especially interesting finding is that exposure to Computational Thinking can be used as an early indicator and predictor of academic success since Computational Thinking scores have been found to correlate strongly with general academic achievement by Haddad and Kalaani (2015). This paper examines first year undergraduate Computer Science students who took a novel test to assess their Computational Thinking skills and in addition a survey gathering their views on Computer Science and Computational Thinking. This survey was administered twice within the academic year and comparisons are drawn on the changes between these survey results.

Keywords: Bebras, CS1, Computer Science, computation thinking, problem solving.

1. Introduction.

One of the main goals of most introductory computer science (CS) courses in third level education is to allow students to develop critical thinking and problem-solving skills that will be useful not just in future years of study, but throughout their life.

The first-year teaching group in the Department of Computer Science at Maynooth University (MU) have implemented numerous different strategies to pass on both subject-specific knowledge as well as develop students who can solve difficult problems through both programming and computational thinking (CT). As well as being involved in the education of third-level students, the Department of Computer Science at MU has, for many years, strived to pass their knowledge and experience to secondary school students and teachers. This has been done through many ways including summer camps, workshops, open days and college fairs. One way in particular is the PACT group, who have developed resources and materials for secondary school and primary school teachers. One of the main focuses of the group is on using CT to teach cross-curricular skills (Mooney et al., 2014).

In this paper we will use Bocconi's definition of CT which describes the thought process entailed in formulating a problem so as to admit a computational solution involving abstraction, algorithmic thinking, automation, decomposition, debugging and generalisation (Boconni et al., 2016). CT is the thought processes required in decomposing problems and constructing solutions that are encountered within CS.

Over the last few years, in conjunction with the PACT group, we have developed a course for secondary schools to introduce students to both CS and CT (Lockwood and Mooney, 2018a). Computer Science to Go (CS2Go) provides teachers with over 80 hours of materials on a wide range of topics including Cryptography, Web Development, Programming and Unplugged activities. As part of this course a problem solving, or CT test, was developed as well as a survey to analyse the views of students towards CS (Lockwood & Mooney, 2018b). Over the 2017-18 academic year, the course and test was trialled in several secondary schools. It was decided that the test would also be suitable at third-level and the decision was made to run the CT tests and views survey on the first-year CS students at MU. The hope was to show students that problem-solving is a key part of CS and to see whether the course improved their CT skills, whether their opinions of CS changed over the course of the year and also whether the problem-solving test could be a predictor of success the problem-solving test could be a predictor of success in the course.

Ethical approval was sought and received to carry out this work through the ethics committee at Maynooth University. Participation was gathered on a voluntary basis and could withdraw from the study at any stage by contacting the researchers. Informed consent was gathered before any data was collected and participants were informed that all data would be securely stored on servers with the Computer Science department at Maynooth University.

2. Methodology.

During the third week of semester one, the first-year students in CS were informed of this study and invited to participate. Participation was optional and took place during their mandatory lab time; however, students were not graded on the CT test created for this study nor did their participation/non- participation have any impact on their module grade. Students were asked to fill out two surveys. The first survey was a personal survey to provide demographic data and other personal data such as age, gender, and previous programming/Computer Science experience. As we developed the course, we were interested to see how students from different backgrounds and demo- graphics might engage with the lessons. To do this we needed to collect the information on a range of topics anonymously using an online form.

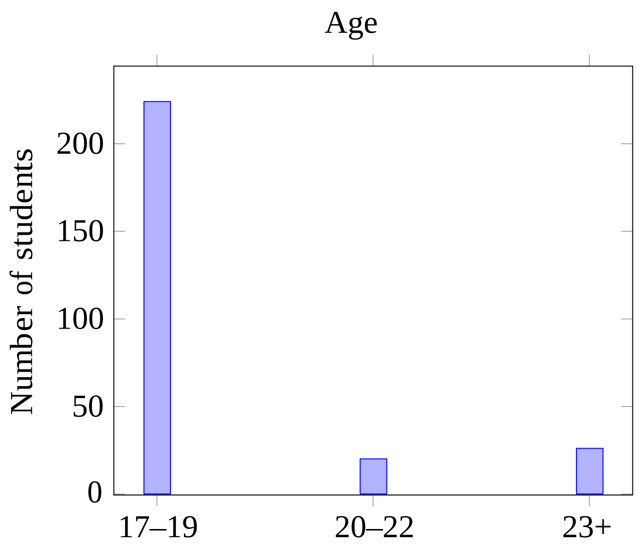
The second survey allowed us to gather perceptions on their views of CS. One of the outcomes we hoped would occur was that after students were exposed to two semesters of CS a change in students' opinions of CS would be observed. To do this a survey was designed based on work by Taub et al. (2012). This survey was designed to better understand and evaluate students' views of what Computer Science is, what it involves, and who a computer scientist is.

Upon completing the two surveys the students then took part in a CT Assessment. They were allowed 35 minutes to complete the problems and could use pen & paper. Students could leave early if they completed the test. The results as well as the answers to the questions were released the following week to all students.

In week 10 of the second semester the students were again asked during their lab time to take part in another CT Assessment which had a similar style and level of problems as the first assessment. It was explained that this test had been uniquely developed for them and that the reason for running the test was to allow them to see if their CT skills had improved over the course of a semester. The students also completed the View of Computer Science survey for a second time.

A total of 271 students completed the survey collecting personal and demographic data. This pool of students came from a wide range of courses and backgrounds and we will present some of the demographic data points of interest that arose from the survey. The age profile of these students can be seen in Figure 1. As is the case with most CS undergraduate courses, the majority of those taking the course were males, with 75% of the class who were surveyed being male.

Figure 1: Age of students taking CS.



In MU there are several different ways for students to graduate with a qualification in Computer Science. It is worth mentioning the four largest courses were represented amongst the students surveyed. Computer Science & Software Engineering (CSSE - MH601/602) is a denominated degree programme and these students take extra CS modules while Robotics & Intelligent Devices is again a denominated degree programme. The two largest degree programmes in MU are the Bachelor of Science (MH201) and Bachelor of Arts (MH101), and Computer Science is an option for both programmes. The remaining students who completed the surveys came from different degree programmes. In terms of Leaving Certificate mathematics, 61% percent had taken higher level, 34% ordinary level while the remainder had taken Foundation mathematics or had another qualification.

3. Results.

3.1. Computational Thinking Assessment.

The CT Assessments have been discussed and described in detail in Lockwood and Mooney (2018b) and can be found in full at: http://www.cs.nuim.ie/~amooney/CS2Go/. For the purposes of this section we will refer to Test 1 as the test taken in the third week of semester one and Test 2 will refer to the test taken in the tenth week of semester two. These tests contained the same number of problems and the assessments were of similar standard.

A total of 292 students took at least one of the two CT Assessments. Of those 292, 263 took Test 1 and 180 took Test 2. From Table 1 it can be seen that a total of 174 took both tests. Students performed marginally better in Test 2, but there is not a significant difference (t-value = 0.20, p = 0.85). This increase is also found across the whole population with the averages being 7.73 (out of 13) for Test 1 (n=263) and 8.05 (out of 13) for Test 2 (n=180), although again this is not a significant change (t-score = 1.49, p = 0.14).

Table 1: CT Assessment scores.

Whole population.	Average of those who took at least one test (whole population)..	Average of those who took Test 1 and Test 2 (n= 174).
Test 1 (n = 263)	7.73	7.99
Test 2 (n= 180)	8.05	8.04

For the following results we will focus on the 174 people who took both tests with Table 2 presenting some demographic data for these students. Of those 174 students, 14 did not fill out the survey so this data is for the remaining 160. For the *Programme of Study*, only those with 10 or more students enrolled are analysed; the next largest group was seven students. Table 3 presents the gender breakdown within the main groups. Table 4 presents the results of t-tests comparing the different demographic groups with each other. The right-most column (T-Test result), compares the Test 1 average with the Test 2 average of the group presented in that row. The final row in each group compares the results of Test 1 and Test 2 of the groups against each other e.g. Male scores in Test 1 compared to Female scores in Test 1. For the maths level question only higher and ordinary level are compared. Table 5 presents the Computational Thinking Assessment score for the four main course groups. The T-Tests presented compare each courses progression from Test 1 to Test 2.

Table 2: Demographic Data

	Demographic	N
Gender	Male Female	123 37
Mathematics Level	Studied OL Maths in Secondary School Studied HL Maths in Secondary School Studied FL Maths in Secondary School N/A to Maths Level	44 111 1 4
Previous Experience	Had previous programming experience Had no previous programming experience	72 88
Programme of Study	CSSE - MH601/602 Bachelor of Science - MH201 Bachelor of Arts - MH101 Robotics - MH306 Other Courses	56 41 23 19 21

HL = Studied Higher Level mathematics, OL = Studied Ordinary Level mathematics, FL = Studied Foundation Level mathematics.

Table 3: Gender Breakdown within the four main course groupings.

Programme of Study	% Male	% Female
CSSE - MH601/602	87	13
Bachelor of Science - MH201	59	31
Bachelor of Arts - MH101	71	29
Robotics - MH306	97	3

Table 4: CT Assessment scores - Demographic Breakdown Part 1

Demographic	Test 1 Average	Test 2 Average	T-Test Result (Test 1 v Test 2)
Male (n = 121)	8.16	8.19	T-Score = 0.09 p = 0.93
Female (n = 37)	7.70	7.78	T-Score = 0.17 p = 0.87
T-Test: Male v Female	T-Score = 1.10 p = 0.27	T-Score = 1.02 p = 0.31	
Studied OL (n = 44)	7.39	7.27	T-Score = 0.26 p = 0.80
Studied HL (n = 110)	8.45	8.51	T-Score = 0.23 p = 0.82
T-Test: OL v HL	T-score = 2.77 p = 0.006	T-Score = 3.41 p = 0.001	
PPE (n = 71)	8.23	8.09	T-Score = 0.39 p = 0.70
NPPE (n = 88)	7.92	8.10	T-Score = 0.55 Pp= 0.58
T-Test: PPE v NPPE	T-Score = 0.85 p = 0.40	T-Score = 0.05 p = 0.96	

PPE = Previous programming experience, NPPE = No previous programming experience

Table 5: CT Assessment scores - Demographic Breakdown Part 2

Demographic	Test 1 Avg	Test 2 Avg	T-Test Result (Test 1 v Test 2)
Bachelor of Science - MH201 (n=41)	8	8.15	T-Score = 0.33 p = 0.74
Robotics - MH306 (n=19)	8.70	8.60	T-Score = 0.07 p = 0.94
Bachelor of Arts - MH101 (n=23)	7.96	7.57	T-Score = 0.51 p = 0.62
CSSE - MH601/602 (n=56)	7.59	7.77	T-Score = 0.47 p = 0.64

3.2 View of CS Survey.

Students completed a survey to determine their views of Computer Science twice. The questions asked are presented in Table 6 along with the question type (all Likert scale questions ranged from 1 - 5). The total number of people who completed both surveys was 147. These people are not necessarily a perfect subset of the group who completed problem solving Test 1 and Test 2 described earlier, however it is in general. Table 7 provides a demographic breakdown of this group of students. It can be seen that the majority of students are male, and that most studied higher level maths. In terms of previous programming experience, it is nearly a 50-50 split. The four largest courses that students are enrolled in are presented, which accounts for 86% of students

Table 6: Questions asked in the View of CS Survey.

Question	Question Type
Q1. Is CS interesting to you?	Yes, No, Maybe
Q2. Is CS challenging?	Yes, No, Maybe
Q3. Using the internet is central to CS	Likert Scale
Q4. Using Word, Excel etc. is central to CS	Likert Scale
Q5. Installing software (e.g. Windows, iTunes) is central to CS	Likert Scale
Q6. Programming is central to CS	Likert Scale
Q7. Being able to solve different problems is central to CS	Likert Scale
Q8. CS is an area related to maths	Likert Scale
Q9. A computer scientist should be good at working with others	Likert Scale
Q10. Boys/men are more likely to study CS than girls/women	Likert Scale
Q11. Work in CS can be done without a computer	Likert Scale
Q12. Have you heard the term CT?	Yes, No

Table 7: Demographic Data - those who completed both View of CS surveys (147 students).

	Demographic	N
Gender	Male	114
	Female	33
School Mathematics Level	Studied HL Maths	104
	Studied OL Maths	39
Previous Experience	Had previous programming experience	70
	Had no previous programming experience	77
Programme of Study	MH201	39
	MH306	19
	MH101	17
	MH601/602	52
	Other	20

Table 8 presents the initial results of the various Likert scale questions. The T-Tests of the Likert scale questions are also presented. A score of one indicated that the participant strongly disagreed with the statement and a score of 5 indicated that the participant strongly agreed with the statement. Table 9 presents the number and percentage of Yes responses to questions 1, 2 and 12.

When we compare the Likert rated questions (Q3-Q11) of those who took both surveys, and those who took at least one, displayed in Table 3.8 we find the results are very similar. All but one of the responses are within 0.1 of each other (out of 5) when comparing the two samples, with the outlier being Q9. For Q9 the whole population started with an average of 3.21 compared to 3.93 for those who completed both. It is hard to see any reason for this, but it should be noted that the gap is closed considerably by Survey 2 with the whole population average being 3.87 and those who completed both having an average of 3.91. The statistically significant changes, namely to Q4, Q5, Q7 and Q11, are also consistent across both samples. Averages for Q4 and Q5 both decreased from Survey 1 to Survey 2 with averages for Q7 and Q11 increasing from Survey 1 to Survey 2.

Table 8: View of CS results Part 1: Those who completed both surveys - AS1B: Avg Survey 1 Both (n = 147 and AS2B = Avg Survey 2 Both (n = 147). Those who completed just one of the surveys - AS1O: Avg Survey 1 Only (n = 268) and AS2O: Avg Survey 2 Only (n = 183). TS = T-Score and PV = P.Value.

Question	AS1B	AS2B	T-Test	AS1O	AS2O	T-Test
Q3	3.31	3.40	TS = 0.83 PV = 0.41	3.31	3.37	TS = 0.57 PV = 0.57
Q4	2.42	2.04	TS = 3.67 PV = 0.00	2.39	2.011	TS = 4.48 PV = 0.00
Q5	3	2.73	TS = 2.10 PV = 0.04	3.03	2.65	TS = 3.46 PV = 0.00
Q6	4.66	4.63	TS = 0.41 PV = 0.68	4.59	4.62	TS = 0.45 PV = 0.65
Q7	4.68	4.81	TS = 2.18 PV = 0.03	4.63	4.77	TS = 2.47 PV = 0.01
Q8	4.17	4.10	TS = 0.82 PV = 0.41	4.08	4.04	TS = 0.39 PV = 0.69
Q9	3.93	3.91	TS = 0.11 PV = 0.91	3.21	3.87	TS = 0.61 PV = 0.54
Q10	3.18	3.22	TS = 0.29 PV = 0.77	3.18	3.22	TS = 0.16 PV = 0.87
Q11	2.85	3.17	TS = 2.47 PV = 0.01	2.67	3.10	TS = 4.07 PV = 0.00

Returning to the results of those who took both surveys, presented in Table 8, it is both encouraging and expected that students view for Q4 and Q5 would move towards a stronger disagreement. After being exposed to CS for a whole year through programming, logic, computer systems and more introductory CS concepts the students would hopefully understand that CS is more linked to programming and problem solving (Q6 & Q7). Student responses to Q6 increased slightly but not significantly, however their responses to Q7 changed significantly from 4.627 in survey 1 to 4.77 in survey 2 (T-Score = 2.47, P = 0.01). Of interest with Q11 is that views about work with CS changed significantly with participants agreeing more strongly with the statement. In Survey 1 the average was 2.85 (out of 5) and this shifted to 3.17 in Survey 2 (T-Score = 2.47, P = 0.01). Although a rating of around 3 implies an almost neutral response, the shift is still present. This could be down to several factors including pen and paper exercises in the various modules or having a better understanding of what CS is and what it involves.

Table 9: View of CS results Part 2. Those who completed both surveys - YAS1: Percentage of Yes Answers in Survey1 and YAS2: Percentage of Yes Answers in Survey 2. Those who completed just one survey - YASO1: Percentage of Yes Answers in Survey1 and YASO2: Percentage of Yes Answers in Survey 2.

Question	YAS1 (n =149)	YAS2 (n =149)	YASO1 (n =268)	YASO2 (n =183)
Q1.	87.25%	81.88%	83.58%	78.14%
Q2.	57.05%	84.56%	60.07%	84.69%
Q3.	61.07%	71.81%	54.78%	67.76%

With the three yes/no questions (Q1, Q2 & Q12) shown in Table 3.9 some interesting observations can be made. Firstly, the increase in those having heard of CT is encouraging but it is somewhat worrying that this skill has not been made clear to almost 30% of the cohort. This percentage was higher for the whole population. CT has been stated as a central skill for all students by Wing (2006) and if this is true then students should be taught and reminded about how to use this problem-solving methodology throughout the first-year course. It should be noted that CT is not explicitly mentioned in any module.

The increase in those agreeing that CS is challenging (Q2) is to be expected as many students would not have had previous experience of it be- fore and so after a year they can appreciate the difficulties of the subject. This finding alongside the high drop-out ratio for CS students make the new Leaving Certificate course, and the Junior Cert coding course, even more important as this will allow students to experience the subject before entering third-level education.

Computer Science third-level courses have a very high failure and dropout rate in Ireland as reported by Quille et al. (2015). We believe this is due in part to students not understanding the core concepts of what CS is, and so when introduced to them in third level they are surprised and almost “give up”. Many see CS as highly employable, which it is with a large number of jobs available[1], and feel that this is a smart or safe option without experiencing either programming or other areas before choosing an undergraduate course. The subject then may not live up to their expectations or be what they expected, which can lead not only to them changing course but also completely disengaging with the subject. This has been reported colloquially through all levels of teaching from the lecturers, lab demonstrators and volunteer tutors.

There were also some interesting differences between the different demographic groups. The results for questions 3-11 are presented in tables (Table 10 – Table 18) with three separate demographic groups considered. These are gender, maths level completed at secondary school and previous programming experience. The results of each group (e.g. male) are compared from Survey 1 to Survey 2 as well as comparing the demographics in each survey (e.g. male v female in Survey 1). These results will be discussed next.

One major demographic comparison group investigated was in relation to those who reported having some experience of programming prior to the course. Those who had no previous experience felt more strongly that using word, excel etc. (Q4 - Table 11) was central to CS. The interesting point here is that this shifts dramatically by the end of the course, with there being a significant difference in the opposite direction.

Male respondents’ views changed more significantly over the year than females when related to the centrality of using and installing software to CS (Tables 11 and 12). It can be seen that males’ views dropped from an average of 2.38 to 2 for Q4 and from 2.96 to 2.68 for Q5. These drops are significant but also both averages start lower than their female peers.

When looking at the maths level of students at second level we can observe two interesting findings. Firstly, those who studied Ordinary level (OL) maths perceive CS as more strongly linked to maths than their Higher-level (HL) peers (Q8 in Table 15). This is not a shift from the first survey results as the same result was found but the difference has increased with those

---

[1] https://eu.usatoday.com/story/tech/talkingtech/2017/03/28/tech-skills-gap-huge-graduates-survey-says/99587888/

who studied OL shifting slightly towards strongly agreeing with those who studied HL shifting the other direction.

Table 10: Q3 Using the internet is central in Computer Science (TS = T-Score and PV – P Value)

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result (survey 1 v
Male	3.31 (n=114)	3.42 (n=114)	TS = 0.79, PV = 0.43
Female	3.30 (n=33)	3.39 (n = 33)	TS = 0.37, PV = 0.71
T-Test Result Male v Female	TS = 0.02, PV = 0.98	TS = 0.12, PV = 0.91	
Studied OL Maths in Secondary School	3.49 (n=39)	3.46 (n = 39)	TS =0.10, PV = 0.93
Studied HL Maths in Secondary School	3.221 (=104)	3.38 (n = 104)	TS = PV = 0.29
T-Test Result OL v HL	TS = 1.49, PV = 0.14	TS = 0.40, PV = 0.69	
Had previous programming experience	3.34 (n =70)	3.53 (n = 70)	TS =1.04, PV = 0.30
Had no previous programming experience	3.27 (n =77)	3.31 (n = 77)	TS = 0.23, PV = 0.82
T-Test Result No experience v Had experience	TS = 0.44, PV = 0.66	TS = 1.14, PV = 0.26	

The difference between male and female responses include males agreeing more strongly that a Computer Scientist should be good working with others (Q9 in Table 6). The gap increases slightly over the course of the year with the males rating increasing from 4.02 to 4.04 and females decreasing from 3.61 to 3.55. Also, of interest, is that those who studied OL agree more strongly that a Computer Scientist should be good working with others. A final finding in relation to this question is that those with previous experience agree more strongly that a Computer Scientist needs to be good working with others. This gap remains at the end of the course but is again narrowed. One of the misconceptions about CS is that it can be a very lonely job working in a dark room alone typing at a terminal, which is rarely the case. Students were encouraged to work together for certain tasks, with the intention that this with the intention that this would highlight how working in teams is important within CS.

Table 11: Q4 Using Word, Excel etc. is central in Computer Science, (TS= T-Score and PV = P).

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result
Male	2.38 (n = 114)	2 (n = 114)	TS = 3.36, PV <0.005
Female	2.55 (n = 33)	2.18 (n = 33)	TS = 1.47, PV = 0.15
T-Test Result Male v Female	TS = 0.96, PV = 0.34	TS = 1.03, PV = 0.30	
Studied OL Maths in Secondary School	2.54 (n = 39)	2.03 (n = 39)	TS = 2.32, PV = 0.02
Studied HL Maths In Secondary School	2.38 (n = 104)	2.03 (n = 104)	TS = 2.89, PV = 0.004
T-Test Result OL v HL	TS = 0.98, PV = 0.33	TS = 0.02, PV = 0.99	
Previous programming experience	2.23 (n = 70)	2.17 (n = 70)	TS = 0.40, PV = 0.69
No previous programming experience	2.58 (n = 77)	1.92 (n = 77)	TS = 4.58, PV < 0.005
T-Test Result No exp v Had exp	TS =2.49, PV = 0.01	TS = 1.70, PV = 0.09	

Table 12: Q5 Installing software (e.g. Windows, iTunes) is central in CS, (TS = T-Score and PV = P Value).

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result
Male	2.96 (n = 114)	2.68 (n = 114)	TS = 1.92, PV = 0.06
Female	3 (n = 33)	2.81 (n = 33)	TS = 0.68, PV = 0.50
T-Test Result: Male v Female	TS = 0.16, PV = 0.87	TS = 0.63, PV = 0.53	
Studied OL Maths in Secondary School	2.92 (n = 39)	2.64 (n = 39)	TS = 1.08, PV = 0.32
Studied HL Maths in Secondary School	2.96 (n = 104)	2.72 (n = 104)	TS = 1.63, PV = 0.11
T-Test Result: OLv HL	TS = 0.19 P-Value = 0.85	TS = 2.71, PV = 2.64	
Previous programming experience	2.93 (n = 70)	2.83 (n = 70)	TS = 0.54, PV = 0.59
No previous programming experience	3.01 (n = 77)	2.60 (n = 77)	TS = 2.24, PV = 0.03
T-Test Result: No exp. v Had exp.	TS = 0.46, PV = 0.64	TS = 1.22, PV = 0.22	

One major debate in both CS education and the wider STEM education areas is the engagement and teaching of female students. When looking at the feedback to Q10 in Table 17 we can see that there is a significant difference between male and female participants' response to the question "Boys/men are more likely to study Computer Science than girls/women". Males tend to disagree more strongly with the statement going into the course and at the course's conclusion. From Survey 1 to Survey 2 males opinions shift slightly, but not significantly, towards agreement, whereas females' views did not change at all on average.

We believe there could be several reasons for this. One could be the fact that, numerically, it is almost always the case in third-level CS courses that there are more males. This can be seen in this cohort, even though the whole-college population of MU is majority female (56% according to official university figures[2]). Other reasons could include the stereotyped view of CS which has extensively been reported on by Anderson et al. (2008), Quille et al. (2017), and Beyer et al. (2003). It is interesting however that females report a stronger agreement than males with Q10, although we should not be quick to draw conclusions on this sample; this could show that more work needs to be done in second-level to show girls that CS is not a male-only discipline, even if at present it is a male-dominated one.

In relation to Q10, those who had no previous CS experience agreed more strongly on average than those with experience in Survey 1. The gap is effectively gone by Survey 2. This again could show another misconception about CS, that those with no previous experience assume that it is a more male-dominated field.

Table 13: Q6 Programming is central in Computer Science, (TS = T-Score and PV = P Value).

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result
Male	4.62 (n = 114)	4.61 (n = 114)	TS= 0.11, PV = 0.91
Female	4.79 (n = 33)	4.70 (n = 33)	TS=0.73, PV = 0.47
T-Test Result: Male v Female	TS = 1.58, PV = 0.12	TS = 0.70, PV = 0.49	
Studied OL Maths in Secondary School	4.56 (n = 39)	4.69 (n = 39)	TS =1.01, PV = 0.32
Studied HL Maths in Secondary School	4.69 (n =104)	4.62 (n= 104)=	TS= 0.97, PV = 0.33
T-Test Result: OL v HL	TS=1.29 PV = 0.20	TS = 0.68, PV = 0.50	
Previous programming-experience	4.66 (n = 70)	4.62 (n = 70)	TS = 0.14, PV = 0.89
No previous programming-experience	4.66 (n = 77)	4.64 (n = 77)	TS= 0.44, PV = 0.66
T-Test Result: No exp. v had exp.	TS = 0.060, PV = 0.95	TS = 0.20 PV = 0.84	

Those with previous experience agreed more strongly that CS can be done without computers (Q11 - Table 18). This gap was narrowed statistically by the end of the course but still exists. Alongside this is the fact that males also agreed more strongly that problem solving is central to CS as well as that CS can be done without a computer (Q9 - Table 16). This could show that the pen and paper exercises as well as theoretical parts of the course impact males more strongly than females.

Table 14: Q7 Being able to solve different problems is central in CS, (TS= T-Score and PV = P Value).

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result
Male	4.68 (n = 114)	4.84 (n = 114)	TS = 1.89, PV = 0.07
Female	4.67 (n = 33)	4.76 (n = 33)	TS = 0.67, PV = 0.51
T-Test Result: Male v Female	TS = 0.07, PV = 0.94	TS = 1.03, PV = 0.31	
Studied OL Maths in Secondary School	4.59 (n = 39)	4.85 (n = 39)	TS = 1.92, PV = 0.06
Studied HL Maths in Secondary School	4.69 (n = 104)	4.81 (n = 104)	TS = 1.65, PV = 0.01
T-Test Result: OLv HL	TS = 0.89, PV = 0.38	TS = 0.48, PV = 0.63	
Previous programming experience	4.67 (n = 70)	4.89 (n = 70)	TS = 2.61, PV = 0.01
No previous programming-experience	4.68 (n = 77)	4.77 (n = 77)	TS = 1.02, PV = 0.31
T-Test Result: No exp. v Had exp.	TS = 0.04, PV = 0.97	TS = 1.75, PV = 0.08	

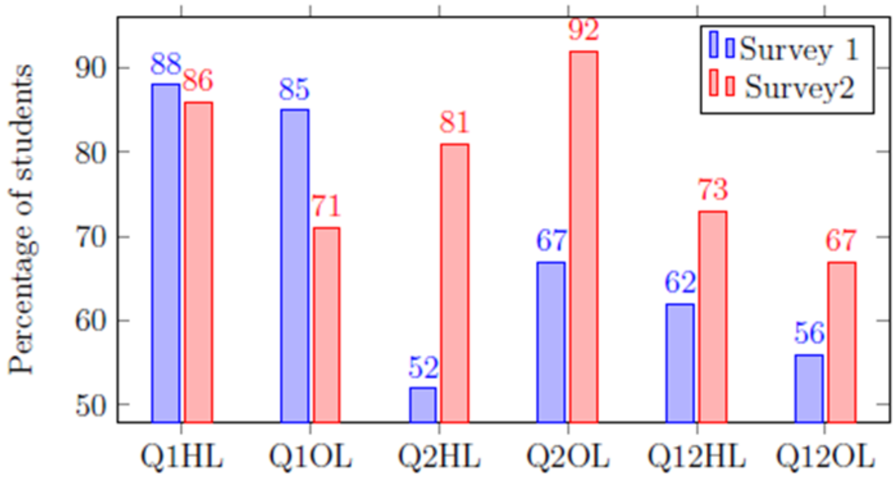
Table 15: Q8 Computer Science is an area related to maths, (TS = T-Score and PV = P Value).

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result survey 2)
Male	4.18 (n = 114)	4.13 (n = 114)	TS = 0.50, PV = 0.62
Female	4.18 (n = 33)	4.03 (n = 33)	TS = 0.85, PV = 0.40
T-Test Result: Male v Female	TS = 0.02, PV = 0.99	TS = 0.63, PV = 0.53	
Studied OL Maths in Secondary School	4.23 (n = 39)	4.28 (n = 39)	TS = 0.31, PV = 0.76
Studied HL Maths in Secondary School	4.14 (n = 104)	4.02 (n = 104)	TS = 1.14, PV = 0.26
T-Test Result: OL v HL	TS = 0.62, PV = 0.54	TS = 1.73, PV = 0.09	
Previous programming experience	4.24 (n = 70)	4.20 (n = 70)	TS = 0.33, PV = 0.75
Mo previous programming-	4.12 (n = 77)	4.03 (n = 77)	TS = 0.84, PV =

experience			0.41
T-Test Result: No exp. v Had exp.	TS = 0.92, PV = 0.36	TS = 1.30, PV = 0.20	

For Figures 2-4 the results from three questions are presented. These are the responses to Q1, Q2 and Q12. Q1 asks "Is CS interesting?", Q2 asks "Is CS challenging? and Q12 asks "Have you heard of CT?". In these figures the blue (left-hand) bar relates to Survey 1 and the red (right-hand) bar relates to Survey 2.

Figure 2: Percentage responses to Q1, Q2 and Q12. HL refers Higher Level Maths students and OL relates to Lower Level Maths students.



Of note is the responses in relation to the level of maths studied shown in Figure 2. More of those students who studied OL initially saw CS as challenging in Survey 1 compared to their HL peers (67% for OL compared to 52% for HL). Both percentages increased from survey one to two, but the gap between OL and HL is still present. OL responded "Yes" 81% compared to 92% for HL in Survey 2.

Figure 3: Percentage of Yes responses to Q1, Q2 and Q12 for students categorised by previous or no previous programming experience. PPE refers to students with Previous Programming Experience and NPE refers to students with no Previous Programming Experience.

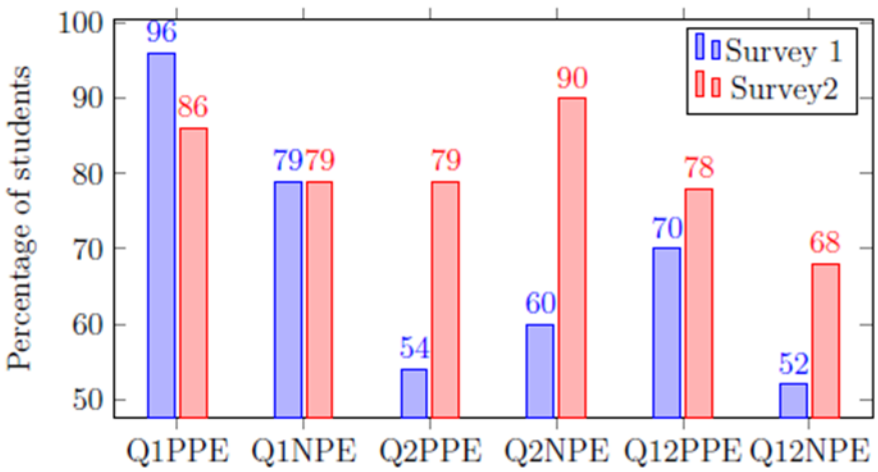


Figure 4: Percentage of Yes responses categorised by gender. MA refers to male respondents while FE refers to female respondents.

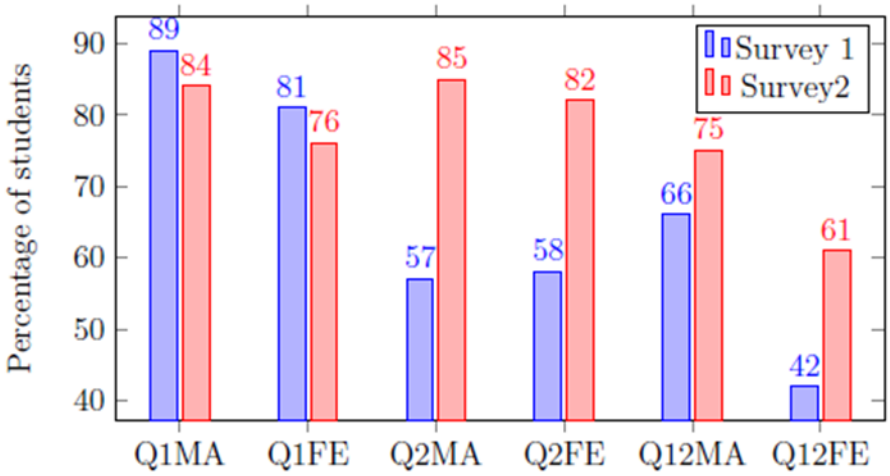


Table 16: Q9 A computer scientist should be good working with others, (TS = T-Score and PV = P Value).

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result
Male	4.02 (n = 114)	4.04 (n = 114) =	TS = 0.24, PV = 0.81
Female	3.61 (n = 33)	3.55 (n = 33)	TS = 0.24, PV = 0.81
T-Test Result: Male v Female	TS = 2.35, PV = 0.02	TS = 2.88, PV = 0.005	
Studied OL Maths in Secondary School	4.10 (n = 39)	4.08 (n = 39)	TS = 0.13, PV = 0.90
Studied HL Maths in Secondary School	3.85 (n = 104) =	3.86 (n = 104)	TS = 0.16, PV = 0.88
T-Test Result: OL v HL	TS = 1.51, PV = 0.13	TS = 1.26, PV = 0.21	
Previous programming-experience	4.13 (n = 70)	4.06 (n = 70)	TS = 0.53, PV = 0.60
No previous programming experience	3.74 (n = 77)	3.82 (n = 77)	TS = 0.49, PV = 0.66
T-Test Result: No exp. v had exp.	TS = 2.67, PV = 0.008	TS = 1.62, PV = 0.11	

The drop in those who found CS interesting (Q1) in Survey 2 is larger in those who studied OL compared to those who studied HL. OL “Yes” responses dropped from 85% in Survey 1 to 71% in Survey 2 compared to 88% to 86% for HL. CT is also clearly more widely understood (Q3) by those who studied HL, with 62% having heard of it before commencing the course compared to 56% for those who studied OL.

The answers for Q12 are presented in Figure 3 for two groupings, namely, those with and those without previous programming experience. It can be seen that CT is more widely known (Q12) by those with previous experience of CS, 70% compared to 52% for those who had no previous experience, prior to commencing their course. This is to be expected as it is a skill that is not yet widely taught in other subjects. Those with no previous experience did not change in terms of finding the course interesting (Q1) with 79% finding it interesting in both surveys. However, there was a significant gain in those who found it challenging (Q2); this was true in both groups (30% increase for those with no previous experience and 24% increase for those with experience).

Table 17: Q10 Boys/men are more likely to study CS than girls/women, (TS = T-Value and PV = P Value).

Demographic	Survey 1 Average	Survey 2 Average	T-Test Result
Male	3.04 (n = 114)	3.11 (n = 114)	TS = 0.39, PV = 0.70
Female	3.61 (n = 33)	3.61 (n = 33)	TS = 0.00, PV = 1.00
T-Test Result: Male v Female	TS = 2.29, PV = 0.02	TS = 2.23, PV = 0.03	
Studied OL Maths in Secondary School	3 (n = 39)	2.92 (n = 39)	TS = 0.25, PV = 0.80
Studied HL Maths in Secondary School	3.22 (n = 104)	3.31 (n = 104)	TS = 0.55, PV = 0.59
T-Test Result: OL v HL	TS = 0.93, PV = 0.35	TS = 1.80, PV = 0.07	
Previous programming-experience	3.00 (n = 70)	3.20 (n = 70)	TS = 0.94, PV = 0.35
No previous programming experience	3.33 (n = 77)	3.23 (n = 77)	TS = 0.49, PV = 0.63
T-Test Result: No exp. v had exp.	TS = 1.57, PV = 0.12	TS = 0.18, PV = 0.86	

Looking at the responses to Q1, Q2 and Q12 for the gender groups in Figure 4 there is not much of a difference between both groups except in Q12, which relates to whether a student has heard of CT. Here we find that males are much more likely to have heard of CT prior to the course (66% compared to 42%). It is interesting to note that students found CS substantially more challenging by the time they got to Survey 2 compared to Survey 1, with little variation observed between males and females.

Table 18: Q11 Work in Computer Science can be done without a computer, (TS = T-Test and PV = P Value).

Demographic	Survey1 Average	Survey2 Average	T-Test Result
Male	2.90 (n = 114)	3.29 (n = 114)	TS = 2.75, PV = 0.007
Female	2.76 (n = 33)	2.79 (n = 33)	TS = 0.13, PV = 0.90
T-Test result: Male v Female	TS = 0.64, PV = 0.52	TS = 2.46, PV = 0.02	
Studied OL Maths in Secondary School	2.87 (n = 39)	3.10 (n = 39)	TS = 0.98, PV = 0.33
Studied HL Maths in Secondary School	2.86 (n = 104)	3.21 (n = 104)	TS = 2.35, PV = 0.02
T-Test Result: OL v HL	TS = 0.08, PV = 0.94	TS = 0.55, PV = 0.59	
Previous programming-experience	3.02 (n = 70)	3.30 (n = 70)	TS = 1.49, PV = 0.14
No previous programming experience	2.71 (n = 77)	3.07 (n = 77)	TS = 2.08, PV = 0.04
T-Test Result: No exp. v Had exp.	TS = 1.77, PV = 0.08	TS = 1.36, PV = 0.18	

4. Discussion.

The results section provides a large amount of qualitative data on the students' scores and results from both the CT Assessment and the Views of CS survey. In this section we will briefly present and discuss a few findings that we believe are of most interest.

4.1 Maths Level and a Link to CT Assessment Scores.

It has been shown that students with stronger mathematical abilities often do better in introductory programming courses (Razak and Ismail, 2018). These links can be seen when it comes to students' scores in our CT skills. This is interesting from two points. Firstly, the fact that students who studied Higher Level maths did better in both Test 1 and Test 2, compared to their Ordinary level peers, can help to support the fact that our test, is challenging students in the right areas. The test problems are designed to test CT and students who take Higher Level maths would, in general, be stronger in maths than their Ordinary level peers. One of the hopes of the LC maths curriculum is to teach students to be critical thinkers and creative problem solvers3. Higher level maths students need to be stronger in these areas to cope with the harder curriculum. With that being said, students who studied Higher Level maths should perform better in the CT Assessment, which is encouraging to find in the data.

Secondly, it has been shown that students' mathematical abilities are a factor when predicting success when learning to program (Razak and Ismail, 2018). The fact that our test shows a significant difference between stronger and weaker students in maths could allow it to help as a success predictor. More analysis is required in this area, including obtaining students' Leaving Certificate maths results as well as their first-year grades in both CS and Mathematics. If this CT Assessment could be shown to help predict success or failure when learning to program, this would allow strategic and timely interventions to be applied and made available to students.

4.2. People who haven't programmed before have misconceptions about what CS is.

As shown in Section 3.2, people who have no previous programming experience may have misconceptions of what CS is. They agreed more strongly that installing and using software like Word and Excel were central to CS (Table 11), that men are more likely to study it than women (Table 16) and that a Computer Scientist doesn't have to be good working with others (Table 17). A higher percentage also found CS less interesting than those who had previous experience with programming (Figure 3). As previously discussed, CS has one of the highest drop-out rates of all university courses. We believe that is partly down to students not being aware of what the subject is before taking it. With the introduction of CS to the Leaving Certificate, the coding short course at Junior Cycle level, and more teachers and schools wanting to introduce some form of CS into their teaching, these misconceptions could be altered sooner. This could also include the misconceptions about genders and computer scientists working alone. Those who had not studied programming before agreed more strongly that men were more likely to study CS and that a computer scientist does not have to be good working with others.

4.3. The Gender Question.

As with all STEM subjects, there is a major push in CS to improve the uptake of female students. Female students generally make up a small minority of CS classes and this study has shown that with the significantly smaller number of female respondents. Of interest is the answers students gave to Q10 of the views of CS survey shown in Table 17. This shows that, to begin with, female respondents felt that CS is a subject that boys or men are more likely to study than girls or women. This could be for a number of reasons, but one likely reason is that fewer female students are introduced to CS in primary or secondary school. This can lead females (and males) to have stereotyped views based on media portrayals of CS-related jobs, teachers' misunderstandings of CS as a subject and many others. There are currently many initiatives to encourage girls to get involved with CS and coding and hopefully these, along with the new Leaving Certificate subject, will allow more students to have an opportunity to discover CS for themselves.

4.4 Future Work.

This work has highlighted a number of possible avenues for future work, including:

- Comparison across different delivery modes (in person/online/distance).
- Comparison with other third-level institutions using Java as a first programming language,
- Comparison with other third-level institutions not using Java as a first programming language.

An interesting comparison would be to look at a similar setting to our study i.e. traditional lectures and labs, or one with a different setup e.g. online or distance learning. These different teaching styles are both very common and with online courses being pursued more by institutions, including MU showing that they can have a similar effect on CT skills could be something to encourage this mode of learning. One drawback from distance learning can be the lack of critical thinking and debate that happens between peers, and this is something that needs to be encouraged in these courses (Broadbent & Poon, 2015). If this can be seen with our tests, then interventions could be developed e.g. by providing problem-solving tasks that can be done as a group online.

Of interest would be comparing our results to other institutions that use Java as a first programming language and those who use a different one language (e.g. Python). Much discussion has been had over recent years (Noone & Mooney, 2018) over which programming language is the “best” as an introductory language, and whilst all have their own merits, it would be interesting to see if one appears to impact students' CT skills more than another.

4.5. Impact of this study on CS2Go.

One of the hopes of this study, as well as the findings from the students' feedback and surveys themselves, was that it could influence the development of the CS2Go course. The most obvious way is the larger sample size for the Computational Thinking assessment that completed it compared to the secondary school cohort. This has allowed us to show that the test can work in both the time-frame desired and it appears to test the right skills; this is evident from the fact those who had previously programmed and studied Higher Level mathematics performed better than their peers. Also, the response to Q10 on genders in the view of CS survey, summarised in Section 4.3, helps to confirm that there is a stereotyped view of CS as a male subject. This could have many influences but we believe introducing all students to CS earlier can allow us to show both male and female students that anyone can study and succeed in CS and CS-related courses and jobs. This is further supported by the misconceptions those who haven't programmed before seem to have about CS and what it is as summarised in Section 4.2.

5. References.

Anderson, N., Lankshear, C., Timms, C. & Courtney, L. (2008). Because it's boring, irrelevant and I don't like computers: Why high school girls avoid professionally-oriented ICT subjects. *Computers & Education*, 50(4):1304– 1318.

Beyer, S., Rynes, K., Perrault, J., Hay, K. & Haller, S. (2003). Gender differences in computer science students. *ACM SIGCSE Bulletin*, 35(1):49–53.

Boconni, S., Chioccariello, A., Dettori, G., Ferrari, A. & Engelhardt, K. (2016). Developing computational thinking in compulsory education-implications for policy and practice (no. jrc104188). *Joint Research Centre (Seville site)*.

Broadbent, J. & Poon, W. L. (2015). Self-regulated learning strategies & academic achievement in online higher education learning environments: A systematic review. *The Internet and Higher Education*, 27,1–13.

Davies, S. (2008). The effects of emphasizing computational thinking in an introductory programming course. In *Proceedings Frontiers in Education Conference (FIE)*.

Haddad, R. J. & Kalaani, Y. (2015). Can computational thinking predict academic performance? In *Proceedings Integrated STEM Education Conference (ISEC)*, pp. 225–229.

Koh, K. H., Repenning, A., Nickerson, H., Endo, Y. & Motter, P. (2013). Will it stick?: exploring the sustainability of computational thinking education through game design. In *Proceedings of the 44th ACM technical symposium on Computer science education*, pages 597–602.

Lishinski, A., Yadav, A., Enbody, R. & Good, J. (2016). The influence of problem-solving abilities on students' performance on different assessment tasks in cs1. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, pp. 329–334.

Lockwood, J. & Mooney, A. (2018a). A pilot study investigating the introduction of a computer-science course at second level focusing on computational thinking. *Irish Journal of Education*, 43, 108–127.

Lockwood, J. & Mooney, A. (2018b) Developing a computational thinking test using Bebras problems. In *TACKLE: the 1st Systems of Assessments for Computational Thinking Learning workshop at EC-TEL 2018*.

Mooney, A., Duffin, J., Naughton, T., Monahan, R., Power, J. & Maguire, P. (2014). PACT: An initiative to introduce computational thinking to second-level education in Ireland. In *Proceedings of International Conference on Engaging Pedagogy (ICEP)*.

NCCA. (2017). Coding short course. Available: http://www.curriculumonline.ie/Junior-cycle/Short- Courses/Coding/.

Noone, M. & Mooney, A. (2018). Visual and textual programming languages: A systematic review of the literature. *Journal of Computers in Education*, 5 (2):149–174.

O'Brien, C. (2017). Computer science to be fast tracked onto Leaving Cert. https://www.irishtimes.com/news/education/computer-science-to-be-fast-tracked-onto-leaving-cert-1.2964672. [Online - accessed August 8, 2018].

Quille, K. Bergin, S. & Mooney, A. (2015). Press#, a web-based educational system to predict programming performance. *International Journal of Computer Science and Software Engineering*, 4(7),178–189.

Quille, K., Culligan, M. & Bergin, S. (2017). Insights on gender differences in cs1: A multi-institutional, multi-variate study. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, 263–268.

Razak, M. R. B. & Ismail, N. Z. B. (2018). Influence of mathematics in programming subject. In *AIP Conference Proceedings*, Volume 1974. AIP Publishing.

Taub, R., Armoni, M. & Ben-Ari, M. (2012) CS unplugged and middle-school students' views, attitudes, and intentions regarding CS. *ACM Transactions on Computing Education (TOCE)*.

Van Dyne, M. & Braun, J. (2014). Effectiveness of a computational thinking (cs0) course on student analytical skills. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, 133–138.

Webb H. & Rosson, M. B. (2013). Using scaffolded examples to teach computational thinking concepts. In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*, 95–100.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.

Yevseyeva, K. & Towhidnejad, M. (2012). Work in progress: Teaching computational thinking in middle and high school. In *Proceedings Frontiers in Education Conference (FIE)*