Journal of Pedagogic Development
Volume 8, Issue 3

"We don't need to write to learn computer sciences": Writing Instruction and the Question of First-year, Later or Not-at-all

Melissa Wong, School of Arts and Social Sciences, Monash University, Malaysia

Contact: Melissa Wong <melissa.wong@monash.edu>

Abstract

This paper discusses the perceptions of computer science students towards a multidisciplinary writing curriculum that was built into a pathway programme to undergraduate study. A qualitative descriptive investigatory study revealed that computer science students in particular felt a strong disconnect between their disciplinary learning and the learning they did in the academic literacy classroom. The degree to which they experienced the pedagogical and assessment differences between the two learning contexts resulted in mild to strong resistance towards the literacy development aspect of the pathway curriculum. This paper highlights a case where first year computer science students articulates their dissatisfactions in this regard, and explores the question of when computer science students should be taught academic literacy, if it is taught it at all.

Keywords: computer science, academic literacy, pathway, first year, writing, pedagogy, assessment

1. Introduction

The significance of academic writing in the computer sciences and how it should be taught have been the subject of past and recent study. Educators in the field have often struggled with designing and incorporating writing instruction into their teaching, assessments and classroom activities. This is unsurprising given that the computer science curriculums are composed primarily by mathematics, programmeming studies, and the engineering of complex systems. A key pedagogical approach in computer science is problem-based learning that focuses on "student-driven problems facilitated by an instructor in order to achieve the learning outcomes of a course" (Fee & Holland-Minkley, 2010, p. 129). Writing is often seen as a practice that is complementary to computer science curriculums rather than a key component of them. Computer science educators are to this day negotiating how writing should be taught to computer science students.

The pathway programme offered by a branch campus of a foreign university in Malaysia caters to students of diverse and is considered to be equivalent to year one of study. Students self-select a stream where they undertake learning in a discipline alongside two compulsory literacy modules, one of them designed specifically around academic literacies (Lea & Street, 2006) a recent and popular approach to teaching writing in higher education. Through direct observation in the classroom, the researcher also noticed that computer science students tended to lack motivation in class activities and group discussions compared to students of other disciplines. The final sense that this issue required investigation was revealed during a

formal course review, to a panel of external and internal reviewers who felt that this issue has to be addressed by the course coordinators. These instances solidified for the researcher the need to examine the problem outlined above.

This prompted an investigatory qualitative descriptive study to examine the claims made by the computer science students in the first six cohorts to enroll in the programme. Semi-structured interviews, student journals and institutional documents provided data for this study. Verbal feedback were given by computer science students in the programme on the lack of relevance between the writing and computing were communicated to the coordinator of the pathway programme. The knowledge gained from this study allows the programme coordinators to reconsider the curriculum design for computer science students in the programme. It is also important for teachers of academic writing and disciplinary teachers to determine if academic writing should be taught in the first-year of university study, if it is taught at all.

2.1 Literacy acquisition

Literacy is the state of being literate particularly with the ability to read and write and by that extension, a person's ability to use language proficiently. Gee (2008) defines literacy as a product of acquisition, not learning, in which the student are exposed to discourses in natural and meaningful social settings. This particular definition is an ideological understanding of literacy as it sees it as a set of socially-embedded practices imbued with values and attitudes about the manner in which ideas and thoughts should be written and expressed and read. It is also accounts for how these practices should take place (Street, 1983, 1995). This study focuses on literacy acquisition amongst computer science in higher education, which also includes a discussion on the transfer of literacy skills from a generically deployed context for employment in a disciplinary context.

For a long time the generic approach to teaching academic writing has been used by both EAP and ESL instructors. In practice, it is often assumed that there are common features in academic writing that can be taught similarly across the board even to native speakers of English (Etherington, 2008). However, there are studies that argue against the notion that a set of generalizable academic skills that can be learned by students in one context, such as an EAP class, that can be later transferred for use in writing in disciplinary contexts. Student writers in higher education tend to acquire academic skills slowly through a combination of intensive and extensive reading, and imitation of academic expressions of others. They also learn from the feedback given by their instructors. Learning processes like these can be more difficult for L2 users. There are some that question the correlation between skills and strategies learned in English as a Second Language (ESL) and English for Academic Purposes (EAP) courses and their application to further learning in disciplinary contexts in higher education (Hyland, 2002; Leki, 2003, 2006; Read & Hayes, 2003; Spack, 1997). Other studies however claim that academic writing can be marginally improved through writing instruction in a college writing course, although the results for these studies varied in terms of what skills were transferred by different sets of students learning in assorted learning contexts (Elder & O'Loughlin, 2003, James, 2006; James, 2010).

More recently a new approach to teaching and learning literacy has been offered that is an ethnographically-based analysis that shifts attention from written texts to the language capacities of student learners. Researchers in the field of academic literacies (Lea & Street, 1998, 1999, 2000, 2004, 2006; Crème & McKenna, 2010) redirect the focus on literacy development to the role of identities, power relationships, and social practices of learners and teachers in higher education. The study of academic literacies is also often known as the study of academic discourse (Hyland, 2009). Lea and Street (2006) make the argument for the academic literacies model as thought of in terms of three overlapping viewpoints or models i.e. the (1) study skills model, (2) the academic socialization model and (3) the academic literacies model. The study skills model is one that views academic literacy as a cognitive skill that emphasizes the foundations of language form. It assumes that students can transfer study skills such as academic reading and writing from one learning context to another without problems (Wingate, 2006). The second model, academic socialization (Beatty, Collins & Buckingham, 2014; Duff, 2010) is primarily related to students' ability to assimilate and acculturate into disciplinary and subject-based academic discourses and genres and "students acquire the ways of talking, writing, thinking, and using literacy that typified members of a discipline or subject area" (Lea & Street, 2006, p. 369). The final model, the academic literacies model focuses on "meaning making, identity, power, and authority, and foregrounds the institutional nature of what counts as knowledge in any particular context" (p. 369).

Disciplinarity is a key notion in this study. Pedagogically, students are taught to distinguish differences in skills and knowledge across diverse fields of study. They are asked to consider differences in academic practices, different student identities, concepts of power, and how learning is unique in each discipline. As previously mentioned, this study has the explicit aim of ascertaining the extent to which computer science students are able to make sense of the foundational literacy aspect of the programme and their potential application to learning after their first year at university. When it comes to publishing in the computer sciences, Hynninen and Kuteeva (2017) study found that there is a need to write to conform to the standard norms of academic writing including understandibility and clarity. However the question that this study asks is: what type of writing competencies do computer science students need in undergraduate study and when is the right time for them to acquire this skillset?

2.2 Writing in the Computer Sciences

Although professional and academic organisations including the IEEE, ABET, CSAB and National Association of Colleges and Employers (NACE) have long conveyed the importance of teaching computer science undergraduates writing skills and yet the problem of how to teach the skill has been subject to few teaching innovations. Despite the lack of emphasis on writing in the computer sciences, its instructors and educators agree that writing is needed to form holistic curriculums in the field. Understanding of writing genres and practices can be varied and incoherent depending on how its practitioners themselves write and how they choose to teach writing in the discipline.

These and many other efforts by proponents of writing across the computer science curriculum (have been part of the Writing across the Curriculum (WAC) movement. The WAC movement is one notable effort spanning more than 30 years of effort to innovate the teaching of writing in the computer sciences. The movement has not managed to reconcile a predominantly expressive and persuasive writing techniques in traditional academic literacy to the writing needs of disciplines such as computer science and engineering. The transactional nature of writing (Indrisano & Squire, 2000) in the computer sciences differs from that of other disciplines in that it often forgoes persuasion and argumentation for expressing solution to problems in text form and user manuals, more commonly referred to as technical writing.

There have been efforts by institutions of higher learning offering Computer Science degrees to incorporate writing into their courses. Not only do learning instructors in the computer science field recognise the importance of writing in the discipline, they have and are deploying various writing activities for their students (Anewalt, 2003; Garvey, 2003; Ladd, 2003; Michael, 2000; Nelson, 2000; Walker, 1998). Dugan Jr. and Polanski (2006) offer a taxonomy of writing tasks that can be applied across several computer science courses. Similarly, Fell, Proulx and Casey (1996) offer an outline of the kinds of writing activities given to students in their CS1 and CS2 and advanced computer science courses at the Northeastern University at Boston, Massachusetts. These activities range from summary writing to simple descriptions of the codes and programme developed by students, logs, detailed reports and essays. Other literature document efforts to teach writing appear in the form of journals also known as lab notebooks and manuals for their software projects (Drexel & Andrews, 1998, p. 61).

3. Methodology

A qualitative descriptive investigatory study was conducted using in-depth semi-structured interviews and documentary analysis. Data was collected using the following methods between May 2015 – 2016.

3.1 Sampling

Purposive sampling was used to select participants for the semi-structured interviews. This sampling method was used since there was a finite subset of students i.e. computer science students enrolled in a multidisciplinary pathway programme comprised of students of other disciplines. Student academic transcripts also showed that there were high failure rates for computer science students that further justified the selection of participants from the overall cohort of students. Sampling for in-depth interviews allowed the researcher to determine the reasons for their poor academic performance. In order to avoid the ethical complications in sampling students whom I taught, who may have agreed to participate out of perceived coercion or in expectation of a returned favour, I sampled only from a pool of computer science graduates from the programme. Post-sampling, with formal authorization from the Head of School at my institution, I was permitted to issue an email call to participation to a short list of six former computer science students of the programme. Five former computer science students responded to my call for participation.

3.2 In-depth Interviews

Semi-structured in-depth interviews were conducted a means of investigating the perceived resistance towards writing instruction showed by computer science students in the programme. The interviews were purposefully half-scripted, where the researcher combined prepared questions and follow-up unscripted questions prompted by interviewee responses during the interviews. An interview protocol consisting of structured questions (see Appendix 1) were carefully crafted to ensure that each interviewee response can be analysed according to a fixed number of themes.

The interview questions were designed with specific themes in mind. One of those themes was the students' learning histories. The interview began with a request to students' to describe their perception of themselves as university students. They were then asked about their prior expectations and perceptions of university life and how those matched up with their experiences as graduates of the course. If there were changes in those expectations and perceptions, they were asked to articulate those differences.

Past learning experiences formed the second phase of the interview in hopes that an inquiry into past formal learning experiences would reveal their expectations of learning at university. They were asked to describe the type of school that they went to and what it was like to learn at those schools. They were also asked how they prepared to enter university, and if they went through a formal course in order to qualify. This section of the inquiry also featured the respondents' answers to difficulties in pre-university learning, and their preferred methods of learning both inside and outside the classroom.

The third category of questions revolved around the kinds of academic support the students received at the institution. They were asked if they received sufficient support from their lecturers, tutors and other instructors specifically in terms of writing instruction outside the literacy classroom, and if those forms of support improved their learning. The students were also asked if those services led to successes in their learning in the programme.

The fourth and final category of questions in the interview protocol was on the students' understanding of academic literacies. Students were asked if they found any practical applications from the literacy modules they were made to undertake as part of the transition course, and if they used the skills they acquired in their subsequent learning in the computer sciences. One of the more important questions that was asked during this part of the interview was what they recalled learning from the academic literacies development portion of the programme. The final thrust of the interview required the interviewees to respond to how they saw themselves applying the acquired skills to future learning.

3.3 Document Analysis

In terms of document analysis this study located selected, categorized, and analysed subject guides, student transcripts and student progress reports, and student journals among other programme, and institutional documents. Other institutional documents gathered also included course accreditation documents, external panel review responses, a programme self-review report and the institutional strategic plan.

An important type of document that this study drew heavily from was student learning journals. These journals were collected from an assessment task given to the students in one of the literacy development modules in the course. The whole student cohort including the computer science students were required to write a journal entry of 200 to 250 words a week over twelve weeks in a semester. They were allowed to journal about any learning activity that made an impression on them in a particular teaching week, whether it be an interesting lecture, a learning activity that required their participation or the difficulties that they encountered learning throughout a particular week.

The students were encouraged to journal about learning in their discipline in addition to their learning in the foundational literacy subjects. No topic in particular was off limits except that they were instructed to keep to writing about their learning, and avoid journaling about their personal lives. The process of sampling student learning journals consisted of determining which journals stood out from the others. The journals that were ultimately chosen evidenced the students' self-awareness, self-reflection and independence in terms of their learning.

The student journals were sampled using a simple exclusion process. Out of 150 scripts, collected from the entire cohort of students of all represented disciplines including the arts and humanities, business, and science students, I selected only the learning journals of the computer science students. Out of the 35 journals that were retrieved, I further excluded those journals that merely described what was learned in a particular week and did not contain any form of reflection by the student. In the end I was left with a sample of 10 student learning journals written by computer science students in the course that were then analyzed using narrative analysis, which will be detailed in the data analysis section below.

3.4 Ethical Considerations

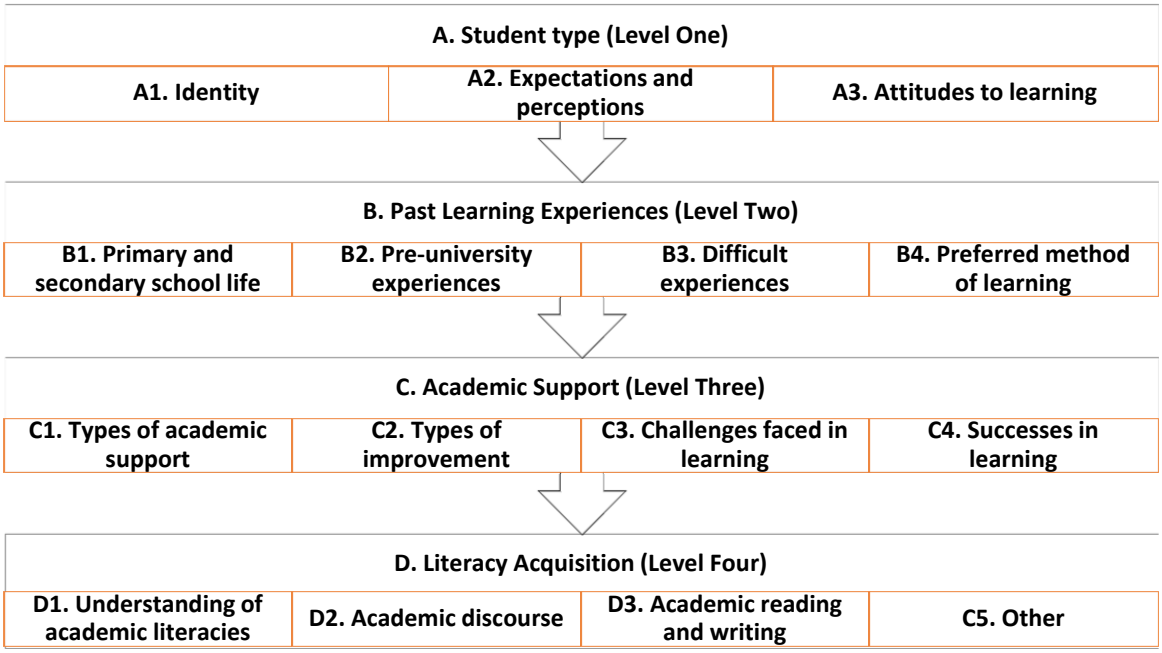
Ethics approval was granted by EdD Virtual Programmeme Research Ethics Committee at the University of Liverpool on the 24 March 2015. To ensure that my participants had comprehensive information regarding my study, such as the aim of the study, and their entitlements and rights including their right to withdraw, I sent a participant information sheet (PIS) to them in the invitation email. It stated the full right of the participant to withdraw at any time of the study for whatever reasons without any consequences to her or him. The PIS also outlined their right to privacy, and confidentiality is guaranteed to the participant. The information that they provided me was de-identified and pseudonyms assigned to them. This occurred throughout the entire process of my research including the de-identification of data when they were stored. They were similarly informed of how data was stored, and when they would be destroyed. The participants were asked to provide informed consent through a consent form.

3.5 Data Analysis

Narrative analysis was used to decode and analyse the narratives obtained through the semi-structured interview and student learning journals. Two coding frames were used to achieve

this. My coding frames were structured according to hierarchical levels where each frame has a main category that forms the first level and subcategories at a lower level. This allowed me to sort themes into a type of order and sequence that would make them easier for management and analysis.

Patterns in the student learning journals and interview transcripts were identified particularly in terms of how certain key ideas were expressed and described, as well as the number of times they occurred. Coding frames such as represented in Figure 1 below provide a mental picture of how data was gathered. The coding frame below depicts the categories of identified themes and categorisations for the data gathered through the student learning journals. The frame consists of clusters and themes on various levels otherwise known as order of abstractions. Coding frames ensured that I was always aware of what data had been derived and at which level. In this manner, such procedures contributed to the reliability and soundness of data and data analysis.



The coding frame above allowed me to deal with data and generate sufficient information for analysis. There were however certain limitations in analysing data in the manner in which I depict above. As with any method of analysis, the selection and categorisations were done with a degree of subjectivity on my part. Of the copious amount of 'thick' data gathered, I chose to examine and highlight themes that were of interest to me and my study. Dealing with copious amounts of data is a challenge when a researcher becomes too familiar with his or her data. In my case, particularly with the student interviews, I found that students often had similar thoughts about their experiences learning in my classes which meant that I had to make sure that I did not overgeneralise their responses and in so doing risk not being able to see beyond their responses to identify outliers in the data.

Indeed, the process of knowing what to do with unexpected themes and surprising findings required significant analysis in order to know what to make of them. It meant sometimes that I had to deal with my own biases as a teacher in the programme. For example, the data had shown that the computer science students in particular felt a major disconnect in being asked to learn academic writing when they opted to learn computer sciences, which they felt were at odds with each other. At first their feedback was quite puzzling to me. I thought surely they could see that all university students would be required to graduate from university with good, if not strong language and writing skills regardless of their discipline. This was an inherent bias that I had to slowly dismantle as I considered their responses over a lengthy period of time. The data that I thought were outliers were in fact was what formed major themes and the opposite was true due to the ways in which I initially chose to classify them. This process has called up my biases as a practitioner-researcher and compelled me to reconsider my deep-seated assumptions about my teaching and my students.

4. Results

This section of the paper presents the results of the case study in three broad categories. Firstly, computer science students have in their learning journals and in interviews expressed their disciplinary learning, and their learning in the literacy classes. Secondly, the accounts of students who narrate their experiences learning in both the disciplinary and literacy classroom and how those experiences are also different adding to their opinion that literacy skills are not needed to learn in the computer sciences. Thirdly, the findings show that not only are there inherent differences in terms of how knowledge is constructed, and how it is taught, differences in assessment also cause computer sciences to feel that it is unnecessary to acquire robust literacy skills to learn in their discipline.

4.1 Disciplinary challenges

One important finding of the study was that first-year computer science students particularly those on a pathway journey to full acceptance into an undergraduate programme, found it difficult to grapple with the content of first-year learning in the discipline and that learning academic writing 'on the side' did not help them with their struggles. The following excerpts from student learning journals and interviews gives us a sense of these difficulties.

Excerpt 1 outlines the experiences of a computer science student in the programme, which comprise many challenges he faced in the first year of study:

> [The] first proper lecture on programmeming, which I already had background knowledge of, but this particular module was different as I had to learn **a new programmeming language** using **a new programmeming application**. The first quiz of the semester was easy, but I have a sense that as time moved on it will become more difficult. The lab session was very difficult at first as the programme we used was **new to me** and the task to be completed was challenging. I needed help and more practice. Algorithms is a **completely new subject** to me, and the **first lecture in algorithms was interesting but difficult to comprehend** (SLJ3).

This journal entry provides us a glimpse of what a student in the computer sciences experiences as a novice, whose foray into a field of study is fraught with newness. The student uses the word "new" four times in that short narration. The use of this word is potentially very significant in light of what computer science students should learn in their first year of study. It raises the question: if they are already struggling with new programmeming languages, is it fair to require them to be able to write academically in English when the valued language is many times the former and not the latter. It may be tempting to write off accounts like this as being quite typical of learning of something new, and that experiences like this are similar across multiple fields of study. The same student wrote in a later week in Except 1a:

> Discrete Math is a bit more **confusing**. Algorithms is **getting difficult** and not a single thing entered my head. Lab session for programmeming was **tough,** but I had help from my peers who helped me complete the task in time. Still, the lectures for algorithms **continue to be a tough** module to understand even with tutorials. I feel demotivated about this module because it **wasn't helping with my understanding at all** (SLJ3).

Except 1b shows his difficulties persisted still through the weeks:

> After the mid semester break and with a new week I decided to start afresh. However the first day back was and I had an algorithms test that I was not looking forward to at all partly because I wasn't well prepared due to the fact that algorithms was such a **difficult concept to grab**. The Discrete Math for this week was **extremely tough** and I could not do it at all. I asked for help from friends and neither knew how to solve it. (SLJ3).

The student's struggles with first year learning in computer sciences are palpably felt throughout the journal. In analysing the journals of other computer science students in the programme, the theme was clear. In a transition setting, these students have a singular goal of completing the pathway programme so that they can progress to year two and beyond of undergraduate study. Their choice was to study the computer sciences and naturally all their focus was centred on learning in their discipline. Therefore, it is not beyond comprehension that these computer science students feel that academic writing has little to nothing to contribute by way of alleviating these learning problems. One computer science student who was interviewed described it simply by stating, "Algorithms is very scary" (CS1).

In a separate interview a student in the programme (CS2) showed how the issue further elevated the problem through his suggestion that the literacy modules built into the programme *disadvantaged* him and his peers rather than help them to cope with disciplinary learning. According to the student:

> I feel like I'm ***disadvantaged*** compared to my peers. Because I'm an [pathway programme] student, I have to take these writing modules that have nothing to do with my learning, but I have no choice. My friends are already ahead because they do two foundational I.T modules. When they get to second year they understand better because they have the basic knowledge needed and they don't struggle as much (CS2).

In another interview (CS3), a student put it very simply by stating:

> Yeah I never saw the point in those units to what I was learning. I wouldn't say that it's a waste of time, but it doesn't do anything for what I'm learning. I prefer to concentrate on what I chose to do (CS3).

And yet another student opines (CS4):

> Only one [literacy] subject is enough. My degree classmates already learn more than me. I don't know if I'll have to write a report or not in my third year. And we have [learned] something similar in a project management class (CS4).

This point in particular is a seemingly strong argument for how the foundational literacy aspect of the programme so-called "disadvantages" its students. There is an extent of truth to the student's argument that other non-pathway students who follow the traditional curricular design would have an edge discipline-wise from undertaking foundational mathematics and programming first year modules rather than academic writing. Despite the logical cogency of such an argument, an alternative perspective is that the pathway students may in fact be better prepared reading and writing-wise for later research-based projects in the latter parts of their undergraduate degree. This study did not aim to address this and is perhaps where further research can be conducted.

4.2 Pedagogical Differences

Thus far the data has centred on disciplinary differences in terms of what it is like for first year computer sciences to encounter new skills and knowledge, and contrast that a different type of knowledge and skill base, that of academic reading and writing. The excerpts have so far shown that the students' sense that there is a strong disconnect between one and the other. This section of the paper furthers this by showing that what happens in the classroom also contributes to the debate on when academic writing should be taught in the computer sciences, if it is taught at all.

A student (CS5) when questioned about how his computer science classes are taught reveals that the teaching and learning strategies to be quite different, and he reflects on the implications of these differences.

> Lectures are not useful [for learning computer sciences] since it's all about doing. All classes are like tutorials where the instructor shows students an example of how to code and shows you how to programme and then you do [them]. (CS5).

The literacy development classes on the other hand, are currently designed around a lecture and tutorial format where students are meant to listen for an hour and then participate in group activities in a two-hour tutorial. The student (CS5) thinks that computer science students are more focused on "hands-on" activities and doing things like programmeming rather than

listening as they do in the academic literacies classes, which he suggests may be "boring" for them.

> The lecturer doesn't give us lectures like in the [literacy] class. He or she usually stands in front of the class and shows us how to write a code. Then, he or she will give us a lab assignment that we have to solve in class. The teacher will go around to check [if it is done correctly] (CS5).

The student (CS5) was asked if academic literacies had any use for him in his learning basic academic reading and writing skills. He thinks that he has benefited to a certain measure by learning those skills and that at some point in the future, he might have to do research project in his final year and may have to write a report of research paper and that is when those skills may be useful.

Being a little more forward-thinking, he also opined that there might be select computer Science students who would like to pursue further studies such as Masters of PhD in the field where academic writing skills would be needed. Otherwise, he thinks that the most writing that they have to do is limited to short answers for tutorial assignments.

4.3 Assessment Differences

An excerpt from a student learning journal (SLJ2) stood out amongst the others for the reason that he expressed what turned out to be a significant issue of the pathway programme, which is its ability to prepare its computer science students to cope with the rigors of second and third-year level study. This account is comparable to the journal entries of other computer science students. From my perspective, the knowledge of this issue warrants serious attention from decision makers in the programme. The student wrote in his journal in the third week of study:

> In academic literacies class we learnt about listening and speaking. The tutorials of this class I always enjoy for some reason. Perhaps because you are allowed to speak more and the tutorial exercises are light and fun. I submitted the first of 10 weekly math assignments yesterday. I'm really happy I did it on my own with a little help from some friends. Unlike last semester where we just used to **copy** [my emphasis] the assignments from other students (Excerpt 3, SLJ2).

The excerpt above illuminates several aspects of the student's experience, one that tells of his enjoyment of the academic literacies class simply because he was allowed to express himself more than his other classes. Secondly, he rejoices in completing one math assignment on his own and admits that previous assessments have been completed through copying and collusion. This brazen admission surprised me, and I look for more information on his claim.

In a subsequent week, the student writes:

> The weekly Maths assignment was submitted again. This time though I copied from Alan (pseudonym) who himself copied the whole thing from another student. I realize and know copying is wrong but sometimes we as students need to do it or we lose marks.

> Although I realize if we study honestly and on time, we don't need to cheat but most of us are not perfect and hence we need to do what is necessary when there is little time (Excerpt 4, SLJ2).

Surprisingly, the student admitted again to "copying", an expression for colluding to submit his work based on his friend's. Similar admissions were made on one or two other student journals. These students reported that the pressure to complete assessments on a weekly basis combined with insufficient time to complete these assignments motivated them to cheat. Like the student in Excerpt 4 above, they understood that collusion to complete their assignments is "wrong", yet their decision to do so was governed by the fear of getting low marks.

These entries were illuminating at the time of reading, but it also caused me to be very concerned about what had been discovered. These students were struggling, and the main question that ran through my mind was: is there more than just lack of time to complete the assessments? I wanted holistic understanding of the assessment structure in the first year of the computer science and so I turned to course documents for possible answers.

The table below illustrates a clue that was uncovered through the analysis of two computer science modules.

Table 1: Learning outcomes and assessment design for two computer science modules

Name of Module	**Module Learning Outcomes**	**Assessment**
Basic Computer Programmeming	1. Develop skills to use diagrams to design solutions for programmeming problems 2. Apply problem-solving strategies and use pseudo-code to design algorithms 3. Design object-oriented solutions to simple problems using multiple user-defined classes 4. Create and test programmeming solutions to problems using JAVA programmeming language […]	1. Assignment 1 – JAVA basics 2. Assignment 2 – Designing a JAVA application 3. Laboratory Work and VILLE Quizzes 4. Final exam
Computer Algorithms	1. Develop knowledge and understanding of the basic ways to structure algorithms, recursion, modular algorithm structures, the equivalence of recursion and iteration, top-down design and bottom-up design, and simple standard patterns for algorithms	1. Assignment 1 - Designing an algorithm to solve problems and 2. Assignment 2 - Understanding different search techniques. 3. Mid-semester test

		4. Final exam

The table above offers a snapshot of the nature of knowledge in computer science-stream of the programme. They also happened to be troublesome modules that these students struggled with the most. Knowledge in computer sciences revolves around computer programme, understanding problems, analysing them and designing solutions to those problems. Therefore, problem-based learning provides a framework for the content of computer science courses and students are required to solve problems across different coursework projects.

The nature of knowledge in computer science is different to the type of knowledge traditionally focused on in literacy classrooms. The two core literacy development modules in the pathway programme however are somewhat misaligned to the academic foci of the computer sciences. Based on Table 1 above, the Computer Algorithms module is designed to develop specialised knowledge in computer algorithms, which is a language different to the use of Standard English and academic writing. Students are required to use computer language for writing codes and not prose for summary or essay writing.

Table 2: Learning objectives of two foundational literacy modules

Learning Outcome No.	**Learning Outcome (LO) Literacy 101**	**Assessment**
LO1.	Demonstrate the ability to access, evaluate, interpret and use information appropriately from a variety of sources, especially within their disciplines.	1. Annotated Bibliography 2. Literature Review 3. Oral Presentation 4. Research Paper
LO2.	Employ skills and strategies for reading a variety of discipline-specific texts: textbooks, reports, research articles and others.	
LO3.	Engage in critical and reflective thinking to respond to and construct academic discourses.	
LO4.	Manage group dynamics and work effectively in teams to solve problems	

	and generate desired outcomes.	
LO5.	Make appropriate choices regarding context, purpose, rhetoric, structure, strategies, and style to communicate effectively for different audiences and academic communities.	
LO6.	Revise and refine work in line with academic conventions, clarity and correctness.	
LO7.	Demonstrate a degree of independence and integration of skills to produce a research paper in their discipline area.	

The assessment structure for the literacy development module is clearly quite different to the first year module on programming and algorithms. Some may argue that this is a given however this given-ness sometimes implies that writing needs equal across all disciplines. To an extent, this is true perhaps in the postgraduate space, however the question remains whether it should be part of their curriculum in year one of undergraduate study.

5. Discussion

First year curriculums in undergraduate degrees are often designed to be in part if not wholly to act as gateways into disciplines. The student narratives above showed that the students felt that a compulsory literacy curriculum stood in the way of their learning in the computer sciences. The literacy component was perceived as a mere hurdle to overcome in order to progress into an undergraduate programme. Overall, they lacked an appreciation for academic writing, more so than students of other disciplines in the programme. The students who spend most of their time with computer codes and mathematical formulae, which are at once troublesome, tacit and alien to them in the first year of university. Therefore they do not see the relevance of those to essay writings, literature reviews and research papers. Their focus is often on computer programming languages and composition of codes rather than written sentences.

The approach taken in the pathway programme above is a generic one where writing is taught outside the discipline, in a literacy classroom that caters to a multidisciplinary student cohort. The fact that it is offered simultaneously to students of other disciplines perhaps signals a lack of coherence between the literacy modules' strong emphasis on writing and the distinctly mathematical and computing nature of many computer science curriculums. When deployed to a multidisciplinary student audience in a generic fashion, computer science students will have

to make sense of how their learning may or may not be enhanced by the items taught in the literacy classroom. Using the conceptual framework of academic literacies, that is precisely what Lea and Street (2006) referred to when theorising that a more dynamic approach to literacy acquisition is the student's ability to make sense of "what counts as knowledge in any particular context" (pp. 227-8). The student will have to look inwards and outwards at how knowledge is constructed in the computer sciences and comparing them to standard academic writing competencies, and reconciling knowledge of one to the other reflexively. As such an embedded approach to academic literacies may be more effective than the current existing structure.

Academics and teachers in the field consider the lack of writing competency in their graduates to be a problem when they enter the workforce. The Association for Computing Machinery (ACM) (2013) Curriculum Guidelines for Undergraduate Programs in Computer Science states that soft skills function crucially in the workplace:

> Indeed, soft skills (such as teamwork, verbal and written communication, time management, problem-solving, and flexibility) and […] play a critical role in the workplace (p. 15)

It is logical that these capacities recommended by the ACM would be important in computing work environments. In fact, other professional and academic organisations including the IEEE, ABET, CSAB and National Association of Colleges and Employers (NACE) have accentuated for many years, the importance of teaching computer science undergraduates writing skills and yet many colleges and universities neglect to do so.

If however studies have shown that teaching instructors in university computer science course are actively promoting writing in their discipline through efforts such as the Writing Across the Curriculum (WAC) then why do its students continue to underestimate the importance of writing to their overall education? (Annewalk, 2003; Michael, 2000; Nelson, 2000). Dansdill, Hoffman and Herscovici (2008) suggest that although the teaching of writing has been regular in some curriculums in the field over the course of a decade, they are proportionately insignificant within the grander body of literature on the subject. They further claim that the WAC movement is nothing but "a few determined voices addressing the field's general historical indifference to national curricular guidelines for the adoption of writing" (p. 25). This seems to resonate with my assessment of the current state of teaching writing in the computer sciences. Accounts in literature of efforts to do this are somewhat sporadic and incoherent. What little writing the students engaged in, did not require a demonstration of persuasive skills, ability to engage in rhetoric or argumentation. That is not to say there were no successes with the computer science students in the literacy classes. There were instances of reported satisfaction associated particularly with the freedom to produce a research paper on a self-selected topic. Some computer science students who were more open to more persuasive forms of academic writing enjoyed the writing process.

The implications for the teaching community in computer science is to consider when writing should be taught to computer science students. If one adopts the stance that writing is a

distraction for first year computer science students then writing it its required forms can be taught at later points of degree study. Instruction by disciplinary experts with knowledge of specific writing requirements in the field that is also of use in their future workplaces may can better meet the needs of students. Teaching of writing can also be embedded into disciplinary curriculums for teachers in computer sciences who require the knowledge and teaching skills of literacy experts (Somerville & Crème, 2005; Monroe, 2003, 2006; Hyland, 2000, 2002); Hyland & Hamp-Lyons, 2002; Klinger & Murray, 2012). A worthwhile strategy would be to show and reinforce to students and staff alike that academic writing is relevant to their future careers in computing.

Academic literacies as a theoretical framework does not pretend to be a structured solution to all disciplinary learning including writing in the computer sciences. Instead, it is a framework that is flexible enough for teachers to use as a pedagogical tool to highlight, explain, describe and show how students themselves can make meanings about learning without consent or guidance from their teachers. It brings to the fore student identities and allows for different academic languages to be spoken and alternate learning cultures to be practiced. Much of it in fact depends on the interactions between students, teachers and texts (Zamel & Speck, 1998). The student can become “an active participant who shares in the responsibility in the process, practices self-evaluation, reflection and collaboration, and conducts a continuous dialogue with his or her coach, tutor or teacher” (Dochy et al. 2007, p. 88). Computer science students in similar transition circumstances can be taught to recognise their own metacognitive abilities, and to use them in meaning making within their own disciplines.

Limitations

Academic literacies key foci are on students’ sensemaking abilities, their identities, and recognition of disciplinary differences particularly in the context of student writing. The findings from this study suggest that computer science students are often not able to make sense of why and how writing skills are relevant to their discipline, which may explain why their motivation to acquire such skills are lacklustre. This study has tried to understand why students in a transition programme are feeling this disconnect, but it does not inform us on what types of writing instruction would be acceptable to them. It also does not tell writing instructors how writing skills can be designed into computer science curriculums, which is an issue that academics in the field have been grappling with for a long time. Although this study did not address the aforementioned gaps, it attempted to understand the limitations in using academic literacies as a conceptual framework to transition first year computer science student to studies in the field. Moving forward, we can ask the question: to what extent can seemingly fixed instructional boundaries within the computer sciences be pushed towards building student writing skills amongst its graduates?

Conclusion

This paper has outlined a case of where literacy acquisition in higher learning by first year students’ accounts were, lacking in its perceived value to the computer sciences. It has shown that students have resisted the literacy modules built into the pathway programme.Their sense was that their day-to-day learning in the computer science classroom did not require the

writing skills taught in the literacy classroom. This study has illustrated how the academic literacies approach in the literacy classroom has limited impact on students of computer science compared to students of other disciplines. It was difficult for them to activate their sense-making abilities as first year students in terms of the writing competencies needed for their present and future learning. The epistemology of the computer sciences rooted in mathematical and algorithmic computation seem not to reconcile with the text-based and persuasive nature of academic writing. The solution that was adopted to resolve this tension was an institutional decision to introduce entry-level computer science units to replace the previously compulsory literacy modules. It remains to be seen if will improve student satisfaction and academic progression in and through the programme. The enduring question of how to build writing into computer science curriculums remains and will require further discussion within the communities of practice in both the computer sciences and literacy studies.

References

ACM. (December, 2013). Computer science curricula 2013: Curriculum guidelines for undergraduate degree programs in computer science. Retrieved 26 November, 2015 from http://www.acm.org/education/CS2013-final-report.pdf

Anewalt, K. (2003). A professional practice component in writing: a simple way to enhance an existing course. *Journal of Computing Sciences in Colleges*, *18*(3), 155-165.

Beatty, S., Collins, A., & Buckingham, M. (2014). Embedding academic socialization within a language support programme: An Australian case study. *The International Journal of in First-year Higher Education*, *5*(1), 9-18.

Dochy, F., Segers, M., Gijbels, D. & Struyven, K. (2007). Assessment engineering: breaking down barriers between teaching and learning, and assessment. In Boud, D. & Falvhikov, N. (Eds.). *Rethinking assessment in higher education: learning for the longer test*. London: Routledge.

Drexel, P. & Andrews, R. (1998). Writing in Computer Science courses: an e-mail dialog. *Plymouth State College Journal of Writing Across the Curriculum*, *9*, 60-68.

Duff, P. A. (2010). Language socialization into academic discourse communities, *Annual Review of Applied Linguistics*, *30*, 169-192. Doi: 10.1017/S0267190510000048

Dugan Jr., & Polanski, V. G. (2006). Writing for computer science: a taxonomy of writing tasks and general advice, *Journal of Computing Sciences in Colleges*, *21*(6), 191-203.

Elder, C. & O' Loughlin, K. (2003). Investigating the relationship between intensive EAP training and band score gains on IELTS. *IELTS Research Reports*, (vol. 4), (pp. 207-254), Canberra: IELTS Australia.

Etherington, S. (2008). Academic writing and the disciplines. In P. Friedrich (Ed.). *Teaching Academic Writing* (pp. 26-58). London: Continuum.

Fee, S. B. & Holland-Minkley, A. M. (2010). Teaching computer science through problems, not solutions, *Computer Science Education*, *20*(2), 129-144.

Garvey, A. (2010, March). Writing in an upper-level CS course. In *Proceedings of the 41st ACM technical symposium in Computer Science education* (pp. 209-213). ACM.

Gee, J. (2008). *Social linguistics and literacies: ideology in discourse* (3rd edn.). Basingstoke: Falmer.

Guba, E. G., & Lincoln, Y. S. (1994). Competing paradigms in qualitative research. *Handbook of qualitative research*, *2*(163-194), 105.

Hyland, K. (2000). *Disciplinary discourses: social interactions in academic writing*. London: Longman.

Hyland, K. (2002). *Specificity revisited: how far should we go now? English for Specific Purposes*, 21.

Hyland, K. & Hamp-Lyons, L. (2002). EAP: issues and directions. *Journal of English for Academic Purposes*, *1*, 1-12.

Hyland, K. & Hamp-Lyons, L. (2002). EAP: issues and directions. *Journal of English for Academic Purposes*, *1*, 1-12.

Hyland, K. (2009). *Academic discourse*. London & New York: Continuum.

Hynninen, N., & Kuteeva, M. (2017). "Good" and "acceptable" English in L2 research writing: Ideals and realities in history and computer science. *Journal of English for Academic Purposes*, *30*, 53-65.

James, M.A. (2006). Transfer of learning from a university content-based EAP course. *TESOL Quarterly*, *40*, 783-806.

James, M. A. (2010). An investigation of learning transfer in English-for-general academic-purposes writing instruction. *Journal of Second Language Writing*, *19*, 183-206. DOI: 10.1016/j.jslw.2010.09.003

Johns, A. M. (1998). The discourse communities dilemma: identifying transferable skills for the academic milieu. *English for Specific Purposes*, *7*(1), 55-60.

Klinger, C. M. & Murray, N. (2012). Tensions in higher education: widening participation, student diversity and the challenge of academic language/literacy. *Widening Participation and Lifelong Learning*, *14*(1), 27-44.

Ladd, B. C. (2003). It's all writing: experience using rewriting to learn in introductory computer science. *Journal of Computing in Small Colleges*, *18*(5), 57-64.

Lea, M. R. & Street, B.V. (June, 1998). Student writing in higher education: an academic literacies approach. *Studies in Higher Education*, *23*(2), 157-172.

Lea, M. R., & Street, B. (1999). Writing as academic literacies: Understanding textual practices in higher education. In C. N. Candlin & K. Hyland (Eds.), Writing: Texts, processes and practices (pp. 62-81). London: Longman.

Lea, M. R. & Stierer, B. (2000). *Student writing in higher education: new contex*ts. Milton Keynes: Open University Press.

Lea, M. R. (2004). Academic literacies: a pedagogy for course design. *Studies in Higher Education*, *29*(6), 739-756. DOI: 10.1080/0307507042000287230

Lea, M. R. & Street, B.V. (Fall, 2006). The "Academic Literacies" model: theory and applications. *Literacies of and for a Diverse Society*, *45*(4), 368-377.

Lee, M. N. N. (1999a). Education in Malaysia: Towards Vision 2020, School Effectiveness and School Improvement. *An International Journal of Research, Policy and Practice*, *10*(1), 879-906. DOI: 10.1108/03090560810891055

Leki, I. (2003). Living through college literacy: nursing in a second language. *Written Communication*, *20*, 81-98.

Leki, I. (2006). Negotiating socioacademic relations: English learners' reception by and reaction to college faculty. *Journal of English for Academic Purposes*, *5*(2), 136-152.

Michael, M. (2000). Fostering and assessing communication skills in the Computer Science context. In *Proceedings of the 31st SIGCSE Technical Symposium*, 52-54.

Monroe, J. (2003). Writing and the disciplines. *Peer Review*, *6*(1), 4-7.

Monroe, J. (Ed.). (2006). *Local knowledges, local practices: writing in the disciplines at Cornell*. Pittsburgh, PA: University of Pittsburgh Press.

Nelson, S. (2000). Teaching collaborative writing and peer review techniques to engineering and technology undergraduates. In *Proceedings of the 30th Annual Frontiers in Education Conference*. (Oct18-21, 2000), 1-5.

Read, J. & Hayes, B. (2003). The impact of the IELTS test on preparation for academic study in New Zealand*. IELTS research reports*, (vol.5) (pp. 237-262). Canberra: IELTS Australia.

Somerville, E. & Crème, P. (2005). 'Asking Pompeii questions': a cooperative approach to writing in the disciplines. *Teaching in Higher Education*, *10*(1), 17-28.

Spack, R. (1997). The acquisition of academic literacy in a second language: a longitudinal case study. *Written Communication*, *14*(1), 3-62.

Street, B. (1983). *Literacy in theory and practice*. Cambridge: Cambridge University Press.

Street, B. (1995). *Social literacies: critical approaches to literacy development*. London: Longman.

Walker, H. M. (1998). Writing within the computer science curriculum. *ACM SIGCSE Bulletin, 30*(2), 24-25.

Wingate, U. (2006). Doing away with 'study skills'. *Teaching in Higher Education*, *11*(4), 457-469.

Zamel, V. & Spack. (Eds.) (1998). *Negotiating academic literacies: teaching and learning across language and cultures*. New York & London: Routledge.

Appendix 1: Interview Protocol

Interview Code No. /Name:

Location:

Date:

Duration of interview:

A. About the interviewee – some background

Question 1

How would you describe yourself as a university student?

Question 2

Prior to entering university or even now what are you expectations or perceptions of university life?

Question 3

One year into your pursuits here, have your expectations or perceptions changed?

Question 4

How would you describe learning at this university?

B. Past learning experiences

Question 5

Can you describe what learning was like at your primary and secondary school?

Question 6

What pre-university programme did you do and what were your experiences like?

Question 7

Was anything particularly difficult in terms of learning at your pre-university (or equivalent) course?

Question 8

What's your preferred method of learning?

C. Academic Support

Question 9

Would you say that you have support from your lecturers, tutors or other learning facilitators here at this university?

Question 10

What kinds of support would you say you've received and from where?

Question 11

Has that support led you to improve in your learning? What are some of your successes in learning so far?

Question 12

What about challenges? Was anything particularly challenging for you? Can you tell me about it/them?

Question 13

Would you say that you've succeeded learning something that you've previously found difficult but overcame that hurdle or difficulty through some measure of support?

D. Academic Literacies

Question 14

Can you tell me what academic literacy is?

Question 15

Would you say you practice academic literacy the way that you learned it in the two modules now in your current learning?

Question 16

How do you practice academic literacy? What do you remember learning?

Question 17

What do recall learning about academic reading in both the academic literacies modules?

Question 18

What about academic writing? What do you recall learning?

Question 19

Have you applied these skills and knowledge to your present learning?

Question 20

How do you think you might apply these skills and knowledge in the future?

Question 21

What other literacies would you say you've acquired or learned from the course?