An Chomhairle Náisiúnta Curaclaim agus Measúnachta
National Council for Curriculum and Assessment

Report on the early enactment review of Leaving Certificate Computer Science

May 2023

Contents

1. Introduction

The National Council for Curriculum and Assessment (NCCA) convened the Leaving Certificate Computer Science (LCCS) Development Group in March 2017 in response to a request from Minister Richard Bruton, then Minister for Education, to introduce computer science as a new senior cycle subject. Towards the end of 2017, the Department of Education (DE) invited all post-primary schools to express their interest in introducing LCCS in Phase 1 of its introduction in September 2018. There were 138 expressions of interest from which 40 schools were selected for Phase 1. Selection was based on criteria including the teachers' previous skills and knowledge in the area of CS, school type, school size and geographical location to ensure the schools in Phase 1 represented the range of post-primary schools in Ireland.

On 5 February 2018, Minister Richard Bruton joined teachers, students and school leaders from the Phase 1 schools to launch the new Leaving Certificate Computer Science (LCCS) curriculum. As part of this event, NCCA and PDST provided initial professional development to the schools in the afternoon session. It signalled the beginning of a significant commitment from the teachers in terms of upskilling and professional learning.

The first cohort of students in Phase 1 schools were due to sit the first examination of the subject in May 2020, however, all state examinations were cancelled in 2020 due to the Covid-19 pandemic. The national rollout phase of the subject began in September 2020. Based on data from the Post Primary Online Database (PPOD), there are 145 schools offering LCCS for the 2022-23 academic year, and approximately 4,200 students studying the course in senior cycle.

An early enactment review was scheduled to be undertaken by NCCA when the first cohort of students had completed the course but this plan was suspended in March 2020 due to the Covid-19 pandemic. It was not possible to commence the review until the 2022-23 school year, when students had completed the curriculum and assessment arrangements as set out in the curriculum specification.

This early enactment review of LCCS was completed in Q4, 2022 to gather feedback and explore:

- how the curriculum is being enacted and how it could be improved
- the student experience of learning and assessment
- how teachers are working with the curriculum.

This report presents an overview of the consultation conducted as part of the review, followed by a summary of the findings. Based on the findings from the review, the report concludes with key insights and recommendations for modifications to the specification.

In considering the feedback outlined in this report, it is important to be mindful of the extraordinary effort made by LCCS teachers to introduce this new subject to the senior cycle curriculum.

2. Background information

This section provides a brief overview of the curriculum specification, its intended assessment arrangements, and the assessment adjustments introduced during the Covid-19 pandemic, all of which is important contextual information when considering the findings below.

Overview of the curriculum

There are three strands in the Computer Science specification (DE, 2018): Practices and principles, Core concepts and Computer science in practice. Strand 3 comprises four Applied Learning Tasks (ALTs). All three strands are interwoven, to be studied concurrently at different stages of the course.

Table 1: Overview of the strands in the LCCS specification

Strand 1 Practices and principles	Strand 2 Core concepts	Strand 3 Computer Science in practice
Computational thinking	Abstraction	ALT1 Interactive information systems
Computers in society	Algorithms	ALT2 Analytics
Designing and developing	Computer systems	ALT3 Modelling and simulation
	Data	ALT4 Embedded systems
	Evaluation and testing	

The course is designed for students to experience the learning outcomes through the lens of the ALTs, and to have multiple opportunities to engage with the learning outcomes in a wide variety of contexts. This pedagogical approach is encouraged in the specification and reinforced through the design of the Continuing Professional Development (CPD) offered to all teachers.

Overview of the intended assessment arrangements

There are two components to the assessment of LCCS: a coursework component and a final examination. The coursework is based on all strands with strand 3 being particularly relevant. It is noted in the specification that timeframe for the coursework is anticipated to be 6-8 weeks from the first week in January of sixth year and is based on a common level brief issued by the State Examinations Commission (SEC). However, in recognition of the feedback from phase one schools, the timeframe for completion of the coursework has always been at least 10 weeks. NCCA also published guidelines for completing the coursework component.

The format established by the SEC for the final examination is composed of 3 sections. Section A and B make up the written examination with section A comprising 12 short questions and section B comprising 3 long questions. Section C is a computer-based examination of programming skills. There is no choice in the questions on the final examination, and students can opt to sit the examination at Ordinary level or Higher level. The (high-level) programming languages for assessment purposes are Python and Javascript.

Adjustments to assessment arrangements introduced during the Covid-19 pandemic

On March 12, 2020, the national response to the Covid-19 pandemic began a process of substantial emergency measures in education. In the case of Phase 1 of the rollout of LCCS, students were not required to submit coursework in 2020, nor did students sit a final examination that year following the introduction of a system of Calculated Grades.

For the 2021 final assessment, students could choose SEC Accredited Grades or participate under revised assessment arrangements. Revised LCCS coursework arrangements, published by the DE in December 2020, included an extended period of 12 weeks, beginning in mid-December with the release of the brief, with teachers given advance notice of which two ALTs formed the primary focus of the brief.

Adjustments were made to the final examination to increase choice and reduce the number of questions to be completed, and Python was identified as the only high-level programming language that would assessed under the arrangement for the final examination. In 2022, there were also adjustments to the final examination but there was not an option of SEC Accredited Grades.

3. Overview of the Review Process

The review comprised the following four modes of engagement:

- A survey issued to teachers in post-primary schools where LCCS has been offered for more than one year
- Written submissions from, and/or bilateral meetings with, stakeholders who were directly involved in the introduction of LCCS to schools since 2018 : Professional Development Service for Teachers (PDST), SEC, Department of Education (DE) and the Computers in Education Society of Ireland (CESI) who are the recognised Subject Association for LCCS.
- School visits to capture the perspectives and experiences of sixth year students, teachers and school leadership
- Online focus groups with students who had completed the LCCS course.

Methodological approach

School visits

The schools selected for the visits were chosen from those schools in the initial Phase 1 cohort of LCCS schools who responded to an open call to participate in the review. Phase 1 schools were the only schools to have enacted the LCCS course before the emergency measures in response to the Covid-19 pandemic. For the school visits, a stratified sample of seven schools was selected, with the breakdown shown in Table 2. The school visits comprised three focus groups involving three to six sixth year students, the LCCS teacher(s), the majority being out-of-field teachers, and a member of the senior management team.

Table 2: Diversity of school type for school visits

ETB	Voluntary Secondary	Community and Comprehensive
3 schools DEIS : 2; non-DEIS: 1 Co-educational: 3	3 schools DEIS: 0; non-DEIS: 3 Co-educational: 1; Boys: 1; Girls: 1	1 school Co-educational and non-DEIS

Written submissions/bilateral meetings

The written submissions and/or the bilateral meetings were by invitation to the four agencies directly involved in the enactment. Three written submissions were received (PDST, DE, CESI) and there were two bilateral meetings (SEC, PDST).

Teacher survey

A self-selecting approach was used for the teacher survey, drawn from the teachers in the schools with more than one year's experience of the course. There were 34 responses to the survey. Table 3 shows the breakdown by school type.

Table 3: Teacher survey demographic by school type

ETB	Voluntary Secondary Schools	Community and Comprehensive	Other
8	15	6	5

The breakdown of the schools that identified as 'Other' was Private (2), Fee-paying (2) and CEIST (1).

Online student focus groups

The focus groups with students who had completed the LCCS course were chosen with the assistance of two Phase 1 schools. These students were now studying third level or Further Education and Training (FET) courses. Some of the students were studying computer science and others studying courses not related to computer science. There were five participants in the online focus groups with students who had completed LCCS.

Students aged 18 years and over consented to their participation in the consultation. Parental consent and student assent were sought for school visit participants under the age of 18. Data gathered through the school visits and online interviews was anonymised and transcribed, and all data from the consultation was stored as digital files in line with the NCCA's Data Protection Policy (2020). The privacy of all participants has been maintained through anonymisation, except where an organisation has given explicit permission to be identified as contributing to the consultation through invited written submissions.

A thematic approach was used for data analysis, framed by a set of guiding themes used throughout the review. This helped to identify and analyse themes within the data gathered, with the assistance of coding. The findings of this analysis are presented in the next section of this report.

4. Feedback from the review

This section presents an overview of the enactment of the LCCS specification, based on the analysis of the feedback from the review. All modes of engagement in the consultation were shaped around similar guiding themes which informed the analysis of the feedback. The analysis is presented under the following headings:

- Achieving aims and objectives
- Strands and learning outcomes in the specification
- Enacting ALTs
- Planning with the specification
- The coursework component
- Final examination.

Achieving aims and objectives

There was a very strong and broad consensus across all modes of engagement that the aims and objectives of the specification are being realised. Participants spoke of the students' enhanced skills of creativity and collaboration, while also improving their ability to work independently. In addition, dispositions and values changed positively in areas such as perseverance, embracing mistakes, confidence, helping fellow students in the classroom and increased motivation for lifelong learning. The increased motivation to learn was reported by students who had completed LCCS:

> 'Computer Science not only is it practical, but again it's because of that relevance ... there's more of an incentive to learn'. (Online focus groups with students who had completed the LCCS course)

In some cases, the teachers linked the facilitation of genuine collaboration during teamwork to the observed changes in dispositions and values.

The structure of the course, and the core pedagogical approach at the heart of the CPD for teachers, is the application of the practices and concepts of LCCS through the ALTs. Across all modes of engagement, the ALTs were seen as central to the success of enacting the aims and objectives. The centrality of the ALTs was reported as particularly enhanced when students engage with the ALTs in a collaborative manner, and teachers plan to developmentally embrace learning outcomes within different contexts and across the three strands. In this context, it was consistently reported that students themselves become a key resource for one another as knowledge and skills are shared through natural collaborative interactions in what one teacher described as a 'socialisation' process.

> 'In the lessons observed, teachers were not overly prescriptive and students were generally being given scope to choose their own topics and design individual technical solutions for practical projects. These effective teaching

practices observed, motivated students to be creative.' (DE, written submission)

'like other people, be struggling in certain areas, you'd go over and help them, and then they'll come and help you ... in a different scenario as well. So we were all looking at learning and developing together. So like, that was like really nice and something probably I'd say I took away from the classroom.' (Focus group of students who had completed LCCS)

Teachers and students reported this collaborative, socialisation phenomenon in the LCCS classroom occurring more regularly than when other, more traditional, pedagogical practices are implemented. Students often linked their enjoyment of the course to the pedagogical approach where ALTs are at the centre of the learning. They also reported how their agency in carrying out the ALTs further enables creativity and self-directed learning. When teachers spoke of adopting this pedagogical practice, they described the initial challenge of 'letting go' of the traditional role of the teacher as central to the knowledge flow and dynamic of the classroom. Adapting this approach was reported by teachers as needing courage and patience, from them and from their students.

'I had to learn to take a step back in the classroom and be comfortable with knowing I'm not the expert in the room.' (Respondent to teacher survey)

There was also some evidence that this approach is not universal.

'In a minority of lessons; there was a tendency to rely heavily on the use of the text book coupled with presenting slides. Such lessons were quite didactic and closer to a lecture format.' (DE, written submission)

Schools that had tracked the post-school pathways of students who studied LCCS, found significant uptake of computer science related courses or courses related to technology, compared to previous years. They also reported an emerging culture of students entering more national STEM competitions since LCCS was introduced and in particular where there was a strong curricular culture in junior cycle and in Transition Year in the area of coding, computer science and STEM generally. Students reported having a more deeply grounded sense of what is entailed in LCCS in schools that had built a wider culture around computer science and technology. Students who had completed the LCCS course spoke of the influence of the experience on the next steps they had chosen and its continued influence in the pathway they had chosen.

'But like, now that I'm in college, the first thing I made sure that I did was make friends with people from other courses. So I can take the knowledge that they learn and like art and film and try to see if I can apply [it]. And they're asking me questions. .... it's being able to take knowledge from other places and improve your own work with it and then share that knowledge with others.' (Online focus groups with students who had completed the LCCS course)

There were some reservations around achieving the aims and objectives related to the breadth of the course in proportion to the time allocated to the subject. It was also reported that extensive time and commitment need to be invested by the teacher, particularly many out-of-field teachers, to ensure they are sufficiently skilled or aware of the latest issues and developments in the world of digital technology.

Strands and learning outcomes in the specification

Participants across all modes of engagement were consulted on the articulation of knowledge and skills in the specification and also the opportunities and challenges of working with the learning outcomes in the specification. In addition, during school visit focus groups, teachers were asked about where they felt additional clarity would support the teaching and learning and also how the strands, designed to be interwoven and interdependent, worked in their classroom practice.

Participants generally reported the specification as being clear on the articulation of knowledge and skills to be acquired by students. Teachers, during school visits and to a lesser extent through the survey, identified areas of the three strands where they felt greater clarity within topics would benefit both classroom practice and the realisation of aims and objectives, while potentially enhancing the extent to which the three strands are interwoven.

The survey asked teachers to speak to the clarity of the articulation of the learning in the specification. While there were mixed views on the articulation, the participants were given space to elaborate. However, this provided limited information around which areas of the specification were clearly articulated or not clearly articulated. Some participants referenced the CPD training as helpful in this regard while some others reported the textbook as their reference point. Others also expressed how difficult it can be to know when a topic is covered or what is examinable.

> 'Very hard at times to wonder what depth to take a topic. Whilst I'm not a fan of strict recipes; I think a limit could be imposed e.g., history, turing, half-adder, database in particular etc.' (Respondent to teacher survey)

Throughout the consultation there was a sense that understanding of the curriculum increases over time as the teachers gain experience and where there is greater clarity in relation to the nature of the examination of LCCS. Those who agreed that the articulation of learning was clear also tended to emphasise the interwoven nature of the learning in the specification as a key element of the student experience.

> 'They acquire knowledge though learning the core concepts in Strand 2 and putting this knowledge to practical use in the ALTs in strand 3. By engaging with societal issues from strand 1 allows for positive dispositions to be nurtured in students as subject is made meaningful for them.' (Respondent to teacher survey)

The overall consensus from these participants was that the specification clearly articulates the knowledge and skills though some areas could benefit from greater clarification. Participants also reported how the knowledge and skills articulated in each strand naturally connect with the knowledge and skills developed in other strands. Using the ALTs in strand 3 as the lens through which the course is mediated was consistently reported as central to the success of connecting the learning across all three strands. The purpose of this section of the consultation was to gather feedback on each of the strands. The suggestions and insights are summarised below.

Strand 1: Practices and principles

There are 3 sections in this strand:

- Computational thinking
- Computers and society
- Designing and developing.

The titles of these three sections are in bold in the specification. Given that text in bold indicates higher level only learning, it was suggested that the titles of all sections in each strand be unemboldened to avoid potential confusion and provide clarity.

Computational thinking

The *Students learn about* section currently has some of the skills associated with computational thinking. Some participants suggested further elaboration of the skills would be beneficial by including decomposition, abstraction, pattern recognition and evaluation.

Computers and society

The *Students learn about* section refers to the Turing Machine without any further elaboration. Participants felt that some clarity around its origins and operation for example would be beneficial for both learning and assessment purposes.

There were also suggestions that the learning outcomes around Artificial Intelligence (AI) and machine learning, in addition to the principles of universal design, could be modified to make them less broad in scope.

It was also suggested that some of the areas designated as higher level should be accessible for students studying ordinary level, given that many of them will do this through the ALTs.
Some participants suggested that the experience of the course could benefit from asking the students to describe not only adaptive technology but also assistive technology.

Designing and developing

Some participants suggested that it should be specified that students learn about the stages of software development, and the roles and responsibilities associated with the stages. Also, in terms of the processes of designing and developing, it would be beneficial to specify agile and waterfall approaches and the life cycles of software development.

It was suggested that in addition to read, write, test and modify computer programs, it could be enhanced to reflect what students do every day when programming by including design as an action verb. It would also create greater consistency with the title of this section.

Strand 2: Core concepts

There are 5 sections in this strand:

- Abstraction
- Algorithms
- Computer Systems
- Data
- Evaluation and testing.

Abstraction
There was a suggestion from teachers on a school visit that when the specification talks of a range of methods for pattern identification and abstraction, that the range could be more clearly specified.

Algorithms
In the sections dealing with sorting and searching algorithms, the sorting algorithms that students learn about include a simple sort which participants found unclear and should be removed. Inclusion of selection sort was suggested by participants and one suggestion was to further categorise the algorithms into in-place and stable sorting. It was also suggested that some basic algorithms could be included such as the Fibonacci sequence, filtering or detecting prime numbers, numbers and dates. In relation to planning and outlining the functionality of an algorithm, the use of pseudo code is currently specified, but the use of flowcharts could also be specified. The complexity of algorithms, which is higher level learning, is generally measured by the use of Big O notation and some participants suggested more clarification would be helpful.

Computer Systems
There was a strong consensus that the components of a computer in the *Students learn about* column could be opened up more to ordinary level students and the separate components could be described more clearly. A common suggestion was to use the basic von Neumann architecture and its operation, such as the fetch-execute cycle, and primary/secondary storage. Similarly for computer system layers and web infrastructure, it was reported the clarity in the learning could be enhanced with some changes in terminology or simply movement of text.

A common suggestion was that the concept of communication protocols could be opened up to ordinary level students and the basic electronics outlined in this section would be more aligned to ALT4 in strand 3. There was also a suggestion that the building of logic gates need not exceed a half-adder in terms of complexity.

Data
Common feedback on this section related to simply referring to the Unicode and ASCII character sets, without naming any subsets, and including compound data types at this point to align better with the learning specified in the algorithms section. Also, in the discussion of strand 3, it was very commonly reported that ALT1 would benefit by removing the explicit requirement to interact with databases as part of the task itself. A smaller number of participants suggested that a more theoretical knowledge of databases, possibly including relational databases, could be included in this section, in addition to more references to data representation, validation, quality, security, cloud storage and big data.

Evaluation and testing

It was noted that the emphasis on testing tends to be from the programmer's view and some explicit inclusion of end-user acceptance testing, such as beta testing, could connect this section even more with user-centred design.

Strand 3: Computer Science in Practice

There are four ALTs in this strand:

- Interactive Information Systems
- Analytics
- Modelling and simulation
- Embedded systems.

The overall consensus was this strand plays an important and significant role in the successful enactment of the overall specification. The impact of the ALTs on achieving the aims and objectives has already been discussed. A more detailed analysis of the feedback on the enactment of ALTs in the classroom is discussed at the end of this section.

The consensus on strand 3 was that the learning is clearly articulated and that the ALTs are highly effective, they can be interwoven and are appropriate for senior cycle students. It was described by teachers in one school as 'high fives are regular' between students and teachers, mainly due to the team atmosphere and the satisfaction that came with creating a digital artefact. There were suggestions regarding a re-focus of the purpose of ALT1 that could potentially enhance the overall student experience of ALTs.

ALT1: Interactive Information Systems

The strong consensus was that the task specified in this ALT was difficult to achieve within the context of the entire course. The task requires students to create a database and design a website in such a way that the website updates in an interactive manner from the database.

At higher level, students are expected to create a relational database. It was felt strongly that this learning outcome obscured students from some key learning. Often in the students' attempts to develop a workable relational database, through the use of Google firebase or mySQL for example, the focus on web design to meet the user's needs tended to be lost. It was described by teachers as unintentionally encouraging 'black-box thinking in students'. It was the only significant area of learning in the strands where participants felt a re-focus would be beneficial and could also present an opportunity to interweave learning outcomes associated with user-centred design and development.

The removal of the higher level, emboldened aspects in both columns would also have the added effect of making the entire ALT accessible to all learners at levels appropriate to their abilities. Such a change would also need to be reflected in overviews of the ALTs that occur in two sections of the specification.

ALT2: Analytics

Participants reported the learning in this ALT as clearly articulated. Some participants felt projects and artefacts from this ALT linked very closely with ALT1 and ALT3, most notably when it came to visualising and representing their data.

ALT3: Modelling and simulation

The general consensus was that this ALT was clearly articulated and plays a key role in visualisation and modelling of data. Participants also noted that the incorporation of the learning outcomes on agent-based modelling and emergent behaviours, while fascinating, was often an outlier in the context of the purpose of the practical aspect of strand 3. The learning about, and assessment of, agent-based modelling would perhaps be better served in the modelling section of strand 1.

ALT4: Embedded systems

This ALT was generally the first task to be completed in strand 3. The strong consensus was the learning is clearly articulated and that building an embedded system, with a microprocessor or a micro:bit and accompanying electronic sensors and controls, tended to engage and excite students.

It was felt that because of the nature of this ALT, the basic electronics description in the *Students learn about* column of the computer systems section in strand 2, would be more appropriate in this ALT. Additionally, the specification refers to robotic systems on several occasions, where embedded systems would be more aligned to the specification.

Enacting ALTs

The experience reported by students during the school visits, and of students who have completed the course, spoke to all four ALTs. Students spoke in very positive terms of the benefits and enjoyment of the ALTs, and particularly of the collaborative nature of ALTs. Across all schools, and in common with the students who had completed the course, what they found particularly powerful was the relevance of the topics, the cross-curricular dimension, the application of existing skills and the development of new skills.

> 'ALTs are more real…. I can see the practical value of them.' (School visit, student focus group)

Across the schools, student agency was encouraged through projects of their own choosing or choosing from a list of suggested briefs. Students reported that this agency, where it was encouraged and enabled by the teacher, tended to enable self-direction, ownership and a sense of originality and creativity in their work. This was not to underplay the steep learning curve associated with learning how to program, problem-solve and to think computationally while working on various ALT briefs.

Most students found the structured collaborative approach to ALTs, with assigned roles, was enriching. It allowed knowledge and skills acquisition to be shared, though some reported the group dynamic in this kind of collaboration could be problematic around shared responsibilities and an uneven balance of design and development skills within the group. They also spoke of how other skills can be developed through the multi-media presentation of their projects, meeting deadlines and maintaining reflection journals. In some cases, notably where they had experience of Junior Cycle Coding, students spoke of perhaps some element of designing applications (apps) or games in the ALTs could enhance the learning in addition to encouraging an agile development approach to project management.

Students reported that the ALTs felt interconnected. Although students who had completed the course tended to see more interconnectedness, and felt the coursework brought the work of the ALTs into focus and gave it more meaning and relevance. Knowing that work from ALTs could be re-used and re-purposed for the coursework was a further incentive for most students to learn and engage with the ALT briefs. They also reported that while the collaborative learning has continued to positively impact their approach to learning in their current pathway, they felt the ALTs also allowed sufficiently for 'individual pursuits' and personal competence. This was seen in action during inspections carried out by the DE, in addition to the centrality of the ALTs in the pedagogical approach:

> 'In terms of inspection in effective or highly effective lessons when working on ALTs there is strong evidence of the prime aim of the specification being enacted: to develop and foster the learner's creativity and problem solving, along with their ability to work both independently and collaboratively' (Written submission, DE)

A student who completed the LCCS course spoke about the parallels between how they engage with ALTs and the approach to knowledge transfer in their FET course:

> 'It's like constantly improving that knowledge, and then when you're in another module you can be like oh well I learned this in this one. Let me bring it into this one as well because I can do this.' (Online focus groups with students who had completed the LCCS course).

Teachers who participated in the consultation spoke of the positive impact of ALTs on teaching and learning and how the pedagogical approach to ALTs, encouraged through the specification and CPD, has become the 'window' to realising many of the other learning outcomes.

> 'Very positive as allows hands on learning opportunities and development of collaborative skills. Enables teacher to ensure strands are interwoven.' (Respondent to teacher survey)

There was also a learning curve for teachers around managing the integration of a broader set of relevant learning outcomes into the classroom experience of the ALT: in other words, weaving the theory into the practice of LCCS. Where a teacher encouraged ALT processes that involved some elements of students reporting to peers, creating video summaries, using the iterative design process and reflecting in a structured fashion, they found it tended to deepen student understanding and engagement with the course. It also tended to enhance student motivation, in particular towards completing the coursework component.

> 'Projects encourage a collaborative culture.' (Teacher, focus group)

'The peer collaboration and cooperation encourages perseverance.' (ibid)

Some teachers spoke during the school visits of pivoting to individual ALTs, during the period of restrictions due to Covid-19, with the intention of switching back to a collaborative approach. Where there was a practice of ALTs being more individualised, students spoke of wanting more collaboration and communication with peers. Teachers across all modes spoke of ALTs 1, 2 and 3 in particular allowing higher programming concepts of Python to be explored in practical meaningful ways, and that ALT1 in particular gave students opportunities to use Javascript in real-life and practical contexts. Developing skills in both Python and Javascript was generally considered by participants to be necessary to enable students to explore the full range of ALTs and to allow some students to further highlight their creative skills.

Planning lessons with the specification

This section of the consultation did not involve consulting students directly but focused on teachers and stakeholders directly involved in the enactment of LCCS.

Teachers tended to observe that learning outcomes in the specification provided opportunities to work to their strengths as practitioners while encouraging creativity in students. The practice that had the most impact on planning was the enactment of the specification through the prism of the ALTs in strand 3, using the ALTs to interweave learning outcomes from across all three strands, while allowing students to choose how they engage with the brief of each ALT.

'The move [to] learning outcomes has led to a change in thinking within our department. The cross-linking of strands is one that has seen topics not being dealt with in isolation but in fact linked to other areas of the course.'
(Respondent to teacher survey)

Planning for the assessment of programming skills was reported as complex and time-consuming, though online Integrated Development Environments (IDE) such as Replit can help. There was some discussion of the steep learning curve and significant cognitive load for most students to become proficient with the technologies needed to program, such as an IDE, or technologies associated with embedded systems (Raspberry Pi or micro:bit). What emerged through the review was the level of self-assessment enabled by the technology itself, which was generally embraced by the students. Students and teachers often reported the learning and assessment of problem-solving, computational thinking and debugging skills as intertwined within the same process.

Teachers also reported the use of ALTs for assessment purposes, not only for the technical competence and computational thinking skills of the students, but also as a means of assessing their communication skills. For example, some teachers use ALTs as opportunities for student teams to present their artefact to an external person.

Confidence and competence in using the specification to plan lessons tended to be linked to the level of experience in teaching the course. Teachers who have been through the initial first cycles of the course often felt that the concept of teaching to the exam runs counter to their approach, while acknowledging that the content of the final examination does have an influence on how they plan. The familiarisation of teachers, and in some cases students, with the learning outcomes was very high.

When participants were of the view that the specification should be more prescriptive, there tended to be a greater importance placed on the final examination in terms of setting boundaries for the learning. Resources from PDST, NCCA and external sources including the textbook, were also referenced in this regard as useful. While some found the textbook useful for planning, other participants found it better to plan directly from the specification.

Where there were two teachers in a department, the benefits extended beyond collaboration and increased capacity into the generation of more ideas, greater motivation and a wider exposure of the subject within the school. One teacher spoke of their transition from two teachers of LCCS to one teacher and the effect seemed to be most acutely felt in the reduced spectrum of pedagogical approaches and the loss of more diverse thinking that can come from working with fellow practitioners. Also, where numbers are growing in schools, it means there is a wider range of student abilities and interests which is in turn changing how teachers plan and differentiate in order to achieve the aims and objectives for all students. The challenges were seen not only in the nature of the lessons but also in the appropriate physical layout of rooms to ensure the pedagogical approaches needed for successful enactment could be accommodated.

There was almost universal recognition of the quality of the support provided by the PDST to assist teachers with the practical aspects of engaging with learning outcomes.

> 'The PDST Continual CPD was invaluable to help me plan for learning outcomes and put some structure into what is expected to be taught.'
> (Respondent to teacher survey)

The resources on compsci.ie, the clustered communities of practice and the online community offered through the digital communication tool Slack, were highly valued, though some teachers reported that the resources could be misaligned to the specification and online conversations could often veer in unhelpful directions.

During most school visits, teachers were asked an additional question regarding how they would plan differently if they knew only the aims and objectives. Typical replies were that they would not change their pedagogy nor the approach of using the ALTs as the main driver of the learning, given their experience over the previous years of teaching the course. Even with the added complication of students needing time to adjust to the somewhat counter-cultural learning environment of the LCCS classroom, teachers would advocate planning for the approaches described in the specification, encouraged by the support services and reinforced by the coursework and the final examination.

Coursework component

The specification describes the assessment of the coursework, stating that the *coursework will be based on all learning outcomes, with those of strand 3 being particularly relevant* (DES, 2018, p.11). Also, during the consultation, participants often moved from ALTs into discussing the coursework, and vice versa.

The students who had completed the coursework spoke of the relevance of the brief and the close alignment of their learning experience to the experience of completing the coursework. Given this experience, it is perhaps unsurprising that they viewed the 30% allocated to the coursework as too low. They welcomed the changes to the assessment arrangements in response to the Covid-19 pandemic as very helpful.

There was a strong consensus around the high degree of alignment between the coursework component and the classroom experience of ALTs, and the positive impact of one upon the other. There was some concern expressed that the coursework submission focused more on the research and video presentation of the artefact than the quality of the artefact itself, and embedding the whole submission in a HTML format was not rewarded in the marking scheme.

It was also observed that artefacts submitted could be enhanced by greater levels of testing by the students. In terms of the logistics around the coursework, there was also a strong consensus that overall the current operation of this component was working very well, notwithstanding some issues reported around the transmission of the report and the time afforded to students in the specification. Teachers reported that the increased time of 12 weeks assigned to the students, as a result of the Covid-19 arrangements, was beneficial to the design and development of the students' coursework. Linked to the longer timeframe, the release date of the brief in December of the second year of the course was cited by almost all participants as crucial for the engagement and wellbeing of the students.

A further arrangement put in place during the Covid-19 emergency response was SEC signalling to schools, early in the second year of the course, which two ALTs would be the primary focus of the coursework. It was reported by many teachers that this measure was helpful in their planning and reducing the workload of the students to prepare for the coursework, though other participants felt it limited the scope of creativity for the students without necessarily reducing their workload.

Final examination

The final examination is a single examination composed of three sections. Section A and B are the written parts of the paper and section C is the computer-based programming part of the examination.

The final examination takes place towards the end of May of sixth year. The specification states that the final examination will be a *computer-based assessment of learning* outcomes (DES, 2018, p. 25). This is not the established practice of the current form of final assessment and some participants suggested that this should be changed to reflect the current reality of a partially written and computer-based examination.

There was strong consensus across the various modes of engagement that there is a very high level of alignment between the specification and the final assessment. This manifested itself most explicitly in the assessment of programming skills, knowledge of the core concepts and the ability to apply the practices and principles of LCCS. It was also reflected in how the assessment generally aligned to the pedagogies in the classroom and the CPD delivery, and to how inclusivity of students was enhanced through the variety in the modes of assessment.

Sections A and B: The written part of the examination

The students in the school visits had not yet experienced the final examination though they did speak positively of the variety in the modes of assessment and some students referenced the benefits of choice in the final examination as a result of the assessment arrangements in response to the Covid-19 pandemic. The students who had completed the course spoke of how the effective reduction in course content, resulting from the greater choice in the final examination, helped them during a period of wide-spread uncertainty and severe disruption to their classroom experience. They reported how the practical nature of the course embeds the theory more deeply than learning the theory in isolation or simply committing it to memory without a context or without an application.

> 'It is not a memory game like most other subjects …. I don't remember anything from school just because it's memory like, how are you supposed to remember it afterwards when you're not continuously remembering it, whereas computer science, I still remember the stuff because of its application.' (Online focus groups with students who had completed the LCCS course)

Throughout the review, teachers felt strongly that the choice offered in the final written examination, in response to the Covid-19 emergency, was needed to reduce the time pressures on students, ensure they could more reasonably answer the required number of questions and answer them to the best of their ability. The choice also made the classroom experience more focused on the learning in the specification than the assessment of the entire specification and therefore less stressful for teachers and students. During school visits, teachers generally viewed these sections of the examination as very well aligned with the specification and with the teaching and learning experience in the classroom. There was a high level of agreement, particularly among teachers who reported that the specification clearly stated the learning that was required for LCCS, and that the examination was aligned to the specification.

On the content of the written paper, teachers felt that it was positive that the paper did not hinge on the level of programming competence of the student and students could pivot to questions on, for example, computational thinking, ethics, computer systems, binary and hexadecimal, computers in society and still 'flourish' in the examination.

Across the various modes of engagement of the consultation, there was a strong consensus that assessment of programming skills in the final examination should be confined to Python, as opposed to Python and Javascript as stated in the specification. Some of the reasons for not assessing both languages included: the danger of steering the classroom experience too much towards learning programming languages, the difficulty in setting appropriate, authentic questions in Javascript mainly due its less forgiving syntax and its relationship to the web, and the sense among participants that the programming concepts required by the specification are learned to a sufficiently high standard through Python without the need for students to program in another language. A further discussion of this issue of one or two high-level languages in the final examination is given below in the analysis of section C of the final examination.

Section C: The computer-based part of the examination

Most of the students consulted on the school visits reported a positive disposition towards the prospect of a 'live' examination of their programming skills citing the close alignment between how they learn to program and how they were due to be assessed. Generally students felt confident about their ability to program and this was evident in how they viewed this section of the final examination. In addition, programming allows for many different solutions to a problem and so being assessed on their programming skills means students can find ways around a problem, which students found appealing and different to other more traditional examinations. The absence of the use of a browser in the examination was seen as misaligned to how programming is done in practice, however the SEC's Python Reference Guide was seen as helpful in this regard.

Students generally found Python a better language for assessment due mainly to its readability, ease of debugging and the more forgiving nature of its syntax compared to Javascript. Some students expressed apprehension about this section of the final examination, often citing the fact there was no time provided in advance of the one hour examination to plan and think about their solution and also a fear of being caught up in debugging a problem.

Students who had completed the LCCS examination reported a wholly positive experience, and this was especially true for this section of the examination. The relevance of the subject and the application of their knowledge were significant for these students. The assessment of the course was not seen as 'memory game' but due to the strong alignment of the various modes of assessment with the learning articulated in the strands, they tended to view assessment more as part of the continuum of the course rather than a discrete and separate entity. The experience reported by teachers and agencies who participated was not at odds with that of the students.

'[section C] adds something to subject. It gives students who may not be strong on theory an opportunity to showcase their skills.' (Teacher, focus group)

'The coding exam went well for the students that had maintained good coding practice.' (Respondent to teacher survey)

'[section C has] a positive backwash into the course' (Teacher, focus group)

Apart from a similar consensus on the strong alignment of the various modes of assessment, the assessment of students' application and understanding of programming concepts through Python was seen as a significant positive for several reasons. As mentioned previously by the students, Python is seen as more readable and more forgiving in terms of, for example, syntax and debugging. It is more accessible and inclusive as a high level programming language and embraces a wider diversity of programming abilities, including students who are new to programming at the beginning of the course. It was reported broadly that extending the same expectations to Javascript for all students could increase the risk of the course being less inclusive, and inadvertently changing the focus to 'expert coding'. The IDEs for learning Python, most commonly Thonny but also IDLE, in turn accommodate a practical assessment of programming competence in the language. Such IDEs preserve the integrity of the assessment process without the need for students to familiarise themselves with a new assessment platform or to access another development environment to test their code. Some less common concerns were expressed around pressure on students who have weaker programming skills using an IDE in a 'live' setting worth almost one third of overall marks.

The logistics of running the final examination in May was almost universally well received by students, teachers and school leaders. Students across all modes of engagement welcomed having one subject fully assessed before June though some concern was expressed at school leader level about the timing of the examination coinciding with preparation for graduation from school, and the associated distractions.

There were also some concerns raised around the heavy load for the LCCS teacher in the run-up to the examination and also on the transmission of the student work through USB sticks. Some participants suggested that the final examination could be held during the same period as other practical assessments are held, and in particular in schools where there was more than one class, access to the requisite number of computers in an exam setting would be more easily facilitated during this period. The one hour allocated to section C was considered onerous by some participants and a suggestion was made again, in a different school setting to the students previously mentioned, that perhaps some design time could be given in advance of the one hour of programming. Other participants felt the one hour was appropriate for the standard of programming expected and the evidence is that a relatively high percentage of students achieve full marks in this section. Apart from some clarity around file handling as a viable topic in a practical examination, there was very strong agreement on the alignment and efficacy of this section of the final examination.

5. Insights, recommendations and next steps

Insights and recommendations

NCCA would like to acknowledge and thank the teachers, students and other stakeholders who contributed to this review. Their feedback was deeply insightful into how the LCCS specification has been enacted and experienced since its introduction in 2018. As discussed in the previous chapter, the review revealed a spectrum of responses and experiences, and an analysis of the feedback indicated that:

- The aims and objectives of the specification are being achieved, and learning through the prism of the ALTs is a significant factor in achieving the aims and objectives.
- The overall structure of the specification enables teachers to exercise professional autonomy and is sufficiently open to facilitate and encourage agency, creativity and ownership of learning in the students.
- The general articulation of knowledge, skills and dispositions in the strands is clear and appropriate to senior cycle students, though some modifications and additional clarity would be beneficial to future enactment.
- The enactment of the specification in the classroom, in the delivery of CPD and through the modes of assessment for certification shows a high degree of alignment.
- The differentiation articulated in the specification is clear and inclusive, though some modifications could benefit the wider range of learners beginning to engage with the course.

A range of insightful and practical suggestions are presented in this report which can now help inform adjustments to the specification to build on and improve the experience of LCCS. Table 4 below sets out recommended responses to key insights that can be acted upon in the short-term. Table 5 sets out recommended responses to key insights that will need to be considered over a longer period. Further details on the exact modifications to the specification that would arise from the recommended responses are summarised in the appendices.

Table 4: Key insights and recommended responses for introduction in September 2023

Key insight	Recommended response
1. The current modes of assessment, and the percentage of marks distributed for each mode of assessment in the specification, play a significantly positive role in the successful enactment of the specification. The assessment of one high level programming language in the final examination seems to have been sufficient to the development of the	Python should be the language assessed in the end-of-course assessment and Python and Javascript assessed in the coursework assessment. In addition, the timing of the release of the coursework brief should be moved towards the end of term 1 of the second year of the course. These measures would further facilitate the opportunities for students to be as creative as

practical programming skills and to the understanding of key programming concepts.	possible and to enhance, and test, their artefacts to the best of their abilities. (See Appendix 1)

Table 5: Key insights and recommended responses for introduction as part of senior cycle redevelopment

Key insight	Recommended response
1. The aims and objectives of the specification are being achieved, and learning through the prism of the ALTs is a significant factor in achieving the aims and objectives.	The inclusivity and universality of the ALT1 in particular, and to a lesser extent ALT3, could be further enhanced, without compromising the levels of theoretical knowledge articulated in the specification. (See Appendix 2)
2. The general articulation of knowledge and skills in the strands is clear and appropriate for students studying at Leaving Certificate level.	Some modifications would be beneficial to future enactment, and additional clarity would improve the articulation of learning. (See Appendix 2)
3. The differentiation articulated in the specification is clear and generally inclusive of the wider range of learners beginning to engage with the course.	There are some minor modifications within a few sections that could make the specification more inclusive of a broader range of learners. (See Appendix 2)

The modifications to the specification to enact these recommendations are shown in the relevant appendices, with potential changes to text underlined in the third column.

Next steps

Based on the insights and recommendations presented above, and consideration of the ongoing redevelopment of senior cycle, the following two-step approach is proposed.

Step one

The proposed modifications set out under Appendix 1 are made to the specification and are introduced to schools offering LCCS with effect from September 2023.

Step two

In order to ensure that the redevelopment of LCCS is informed by ongoing research on assessment in senior cycle, and in particular, the research and deliberations on the technical form of curriculum specifications, a Development Group should be convened. This development group could then consider the recommendations set out under Appendix 2.

It is proposed that the work of this group could commence within one of the next tranches of subjects to be revised as part of the redevelopment of senior cycle. This will ensure that a revised LCCS specification, informed by this review, will support the realisation of the purpose and vision for a redeveloped senior cycle.

Appendix 1: Proposed modifications with effect from September 2023

Proposed modifications to the LCCS specification, based on the first set of recommended responses in Table 4

Section of the current specification	Current text	Proposed text (*actions such as remove, move, etc are in italics*) Some inclusions of additional text are underlined.
Structure of Assessment for Certification (p.25)	Computer-based assessment of learning outcomes	Written and computer-based assessment of learning outcomes
Coursework Assessment (p.27)	In January of the second year of the course ..... A period of 6-8 weeks is anticipated	Towards the end of term 1 of the second year of the course ..... A period of 10 weeks is anticipated ....
Assessment Programming Languages (p.28)	Leaving Certificate Computer Science does not require a specific language. However, for the initial years of the subject, Python and JavaScript will be the languages used in the end-of-course assessment and the coursework assessment; this will be reviewed on an ongoing basis.	Leaving Certificate Computer Science does not require a specific language. However, following the initial years of the subject, Python will be the language assessed in the end-of-course assessment and Python and Javascript in the coursework assessment. This will continue to be reviewed on an ongoing basis.

Appendix 2: Proposed modifications for consideration as part of senior cycle redevelopment

Proposed modifications to the LCCS specification, based on the second set of recommended responses in Table 5

Section of the current specification	Current text	Proposed text (*actions such as remove, move, etc are in italics*) Some inclusions of additional text are underlined.
p.2, p.10, p.15, p.22	robotic systems	embedded systems
Applied Learning Tasks (p.15)	1. Create an artefact or website that can display information from a database.	1. Create an <u>interactive website / application to meet specific user needs</u>
Strand 1: Computational Thinking (p.18) *Students learn about*	S1: Computational Thinking Problem Solving, Logical Thinking, Algorithmic Thinking	S1: Computational Thinking <u>Decomposition, Pattern Recognition,</u> Problem Solving, <u>Abstraction,</u> Logical Thinking, Algorithmic Thinking, <u>Modelling, Evaluation</u>
Strand 1: Computers and Society (p.19) *Students learn about*	**Turing Machines**	Turing Machines: description **and operation**
Students should be able to	1.14 explain when and what machine learning and AI algorithms might be used in certain contexts 1.15 consider the quality of the user experience when interacting with computers and list the principles of universal	1.14 explain when machine learning and AI algorithms might be used in certain contexts 1.15 consider the quality of the user experience when interacting with computers, including the role of a user interface and the factors that contribute to its usability

	design, including the role of a user interface and the factors that contribute to its usability **1.16 compare two different user interfaces and identify different design decisions that shape the user experience** 1.17 describe the role that adaptive technology can play in the lives of people with special needs	1.16 compare two different user interfaces and identify different design decisions that shape the user experience 1.17 describe the role that adaptive and assistive technology can play in the lives of people with special needs
Strand 1: Designing and Developing (p.19) *Students learn about* *Students should be able to*	Software development and management 1.22 read, write, test, and modify computer programs	Software development: approaches (agile and waterfall), life cycles and stages 1.22 read, design, write, test, and modify computer programs
Strand 2: Algorithms (p.20) *Students learn about* *Students should be able to*	Sorting: Simple sort, Insert sort, Bubble sort, **Quicksort** Algorithmic complexity 2.5 use pseudo code to outline the functionality of an algorithm	Sorting: Selection sort, Insert sort, Bubble sort, **Quicksort** **Algorithmic complexity: Big O notation, equivalence classes** 2.5 use pseudo code and flowcharts to outline the functionality of an algorithm
Strand 2 : Computer Systems (p.20) *Students learn about*	CPU: ALU, Registers, Program counter, Memory Basic electronics: voltage, current, resistors, capacitors, transistors Operating system layers: Hardware, OS, Application, User	Basic von Neumann architecture **and its operation** *Move text to the Students learn about column of ALT4, in strand 3* *Include* Units of logic gates: from basic to half-adder Computer system layers: Hardware, OS, Application, User

	Web infrastructure - Computer Network Protocols: HTTP, TCP, IP, VOIP	Web infrastructure - including the client server model, communication protocols (**layers: application, transport, network and physical**), internet hardware components, and cloud computing
Students should be able to	2.13 describe the rationale for using the binary number system in digital computing and how to convert between binary, hexadecimal and decimal 2.15 explain what is meant by the World Wide Web (WWW) and the Internet, including the client server model, hardware components **and communication protocols**	2.13 describe the rationale for using the binary number system in digital computing and <u>perform basic binary arithmetic</u>, including conversion between binary, decimal and hexadecimal numbers 2.15 explain what is meant by the World Wide Web (WWW) and the Internet, and <u>describe the web infrastructure</u>
Strand 2: Data (p.21) *Students learn about*	Boolean, integer, real, char, string, date, array 8-bit ASCII, Non-Roman character sets Unicode: UTF-8, Emojis	Boolean, integer, real, character, string, date, array, <u>compound data types</u> ASCII and Unicode
Strand 2: Evaluation and testing (p.21) *Students learn about* *Students should be able to*	Testing: Unit test, **Function test, System test** 2.22 explain the different stages in software testing	Testing: Unit, **Function, System (alpha and beta)** 2.22 <u>outline</u> the different stages in software testing
Strand 3: ALT1 Interactive Information Systems (p.22)	In this applied learning task, students will develop an interactive website that can display information (either local or remote data) from a database to meet a set of user needs.	In this applied learning task, students will develop an interactive website/application that can display information to meet a set of user needs.

Students learn about *Students should be able to*	File systems and relational databases 3.2 create a basic relational database to store and retrieve a variety of forms of data types	*Remove text. Include* Graphical Use Interfaces (GUIs); HTML/CSS 3.2 design a user interface taking the quality of the user experience into account *(see LO 2.18 below for knowledge of data storage)*
Students learn about	Information Systems	Data Storage: cloud storage, **Big Data** and traditional database management systems
Students should be able to	2.18 collect, store and sort both continuous and discrete data 3.3 use appropriate programming languages to develop an interactive website that can display information from a database that meets a set of users' needs	2.18 collect, store and sort both continuous and discrete data, and describe different approaches to data storage 3.3 use appropriate programming languages to develop an interactive website/application that can display information to meet a set of users' needs
Strand 3 : ALT3 Modelling and Simulation (p.23) *Students should be able to*	3.9 analyse and interpret the outcome of simulations **both before and after modifications have been made** **3.10 explain the benefits of using agent-based modelling and how it can be used to demonstrate emergent behaviours**	3.9 analyse and interpret the outcome of computer models and simulations both before and after modifications have been made *Remove **LO 3.10** and incorporate the modelling aspect into LO 1.9* 1.9 use modelling and simulation in relevant situations **and explain the benefits of using agent-based modelling**

An Chomhairle Náisiúnta
Curaclaim agus Measúnachta
National Council for
Curriculum and Assessment