studyclix.ie
makes exams easier

Computer Science

How to get an H1 in the Leaving Cert Computer Science Exam

by Aoife Whelan

Aoife Whelan received a H1 in her higher Leaving Cert Computer Science exam. She is now studying in St Patrick's College DCU.

Table of Contents

Introduction

Studying Computer Science for the Leaving Cert has a huge range of benefits. It is a subject that will broaden your knowledge of technology and will boost your skills as a computer programmer.

Computer Science is one of few subjects that is not studied at Junior Cycle, and so it is an entirely new experience for all students who choose it. This course is accessible to people with all capabilities and levels in coding.

The main languages taught are:

- **HTML** (HyperText Markup Language) - a language used for web development.
- **Python** - used for many different coding tasks including analytics
- **JavaScript** - can be used to create interactive elements in web development.

Advantages of Computer Science for Leaving Cert:

As technology is constantly evolving, this subject provides the opportunity to delve into current topics and research real-life examples.

The ALTs (Applied Learning Tasks) which you will prepare for during the two year course, will allow you to work in groups and with fellow peers.

This subject is a great basis for anyone who would like to pursue a career in the technology sector. Equally however, the skills you learn can be of benefit in various careers outside of computer science.

Finally, the coursework project you complete in 6th Year will be worth 30% of your overall grade.

Key Dates

Keep in mind the following dates for LC Computer Science:

- The coursework project is to be completed over a ten-week period (usually between January and March of 6th Year)
- The Leaving Cert Computer Science exam takes place around three weeks in advance of all other subjects. (This is based on the 2021 exam which took place on the 22nd of May).

The Mark Breakdown

The computer science exam can be broken down into 3 distinct sections:

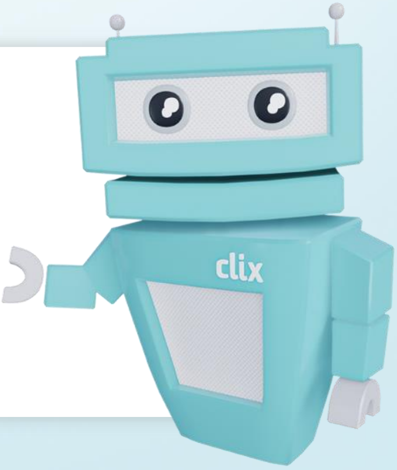
- The written paper (Section A and B)
- The programming component (Section C)
- The coursework component (project)

There are a total of 300 marks available for Computer Science.

- Section A and B together are worth 130 marks (43%)
- Section C is worth 80 marks (27%)
- The coursework project is worth 90 marks (30%)

Tip

Do not rush the short questions as you may miss important parts of the code resulting in an incorrect answer!



Section A: Short Q's

The first section of your written paper is made up of 12 short-answer questions. This section of the paper is worth 60 marks in total.

- Many of the questions in this section will require you **to demonstrate your knowledge** by giving a definition or explaining the role of a particular computer component, protocol, or system.

- Remember these are short answer questions, so you will not need to give a paragraph for every answer. **A concise, correct answer** will get you better marks than a long answer in which the facts are not made clear.

- **Flashcards** are a great way of studying for these types of questions.

- Sometimes you will be given a piece of code (Python, JavaScript or HTML) and you will be asked questions based on it. In these questions, **it is important that you are familiar with simple operators and operations in the different languages.**

- If there is a calculation involved carefully **go through the code on some rough work paper** and work out the output first before doing the question.

- Note the full meanings for any **common abbreviations** that you come across as they may be asked in section A. Some examples of abbreviations are included in the table on the next page.

Abbreviation	Meaning
AI	Artificial Intelligence
ALU	Arithmetic Logic Unit
ASCII	American Standard Code for Information Interchange
BIOS	Basic Input Output System
CPU	Central Processing Unit
CSS	Cascading Style Sheets
GB	Gigabyte
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
IP	Internet Protocol

KB	Kilobyte
LAN	Local Area Network
LED	Light Emitting Diode
MB	Megabyte
OS	Operating System
PC	Personal Computer
PNG	Portable Network Graphics
RAM	Random Access Memory
ROM	Read Only Memory
TCP	Transmission Control Protocol
UI	User Interface
URL	Uniform Resource Locator
USB	Universal Serial Bus
WAN	Wide Area Network

Section B: Long Q's

Section B is worth 70 marks in total and consists of 3 long-answer questions.

- **Do not be thrown off by the fact that these are "long questions"**, as there will be multiple parts to each question, with some only requiring one-word answers.

- In the case of an essay style question, a helpful approach is that of **Point, Example, Explain.**

- This section will often contain questions which will involve **your own opinion** as part of the answer which allows you to interpret the question yourself and draw your own conclusions based on what you have learned.

- In this section of the paper, you may be required to do calculations of your own (e.g Turing Machine/ Sorting algorithm). **Practice examples of these calculations** so that you are prepared on the day of the exam.

- **There are no topics that are guaranteed to come up in this section**, however the core practices and concepts outlined in the curriculum are a good place to start. Here are some of the key topics which tend to appear on the paper:

Key Topics

Computational Thinking	Abstraction Decomposition Algorithms
Computers and Society	Turing Machines Machine Learning + Artificial Intelligence The Internet vs The World Wide Web Ethics
Software Development and Design	Stages of software design User-centred design Accessibility
Computer Systems	The CPU Hardware +Software Networks Network/Communication protocols
Data	Data types (string, integer, float, boolean etc.) Character sets (ASCII, Extended ASCII, Unicode) Binary Hexadecimal

Section C: Programming

The programming component of the exam will consist of one question (which may be split into multiple parts).

This question may be rooted in any of the three main languages (Python, JavaScript or HTML), however, as of 2021 Python is the only language to have been examined in this section so far.

You will open a prepared file on the device you are using to complete the exam, and this is what you will edit in line with the question you are given.

You will receive a Python reference guide (which can be found on examinations.ie), which contains some of the basic commands and operators which you may need to use during the exam.

Remember

Even if exam nerves throw you off, you can always comment out the code that you tried (using the # symbol in Python) as the examiner will look at this during their corrections and you may pick up some marks.

Similarly, if there is a part of the question that you are struggling with, move on and answer what you can and then return to the parts you didn't complete.

Top tips for succeeding in the programming section:

Check with your teacher which application you will be using on the day of the exam to complete the programming task. You can then download the specified platform and become familiar with how it works and use it when practicing. Doing this will prevent any extra stress on the day of the exam, as you will be comfortable with using the software in front of you and you can focus all your attention on the questions being asked.

Be familiar with all basic commands and operators in the language. Below are a few examples (in Python):

Abbreviation	Meaning
*	Multiplication
/	Division
**	Exponentiation
%	Modulus
//	Floor division
==	Equal to
!=	Not equal to

Practice using lists in your programming and make sure you are comfortable with appending values to a list, ordering lists, inserting items in lists and working with index values.

Remember

In a list, the first value always has the index value of 0, not 1.

If, elif and else will be used often in most Python programs you create in Computer Science. Be sure to know how to use them correctly (including when to use elif rather than else and how to use the correct syntax)

Example of If Else In Python

```
x = 22
y = 13
if x > y:
    print(x,"is greater than", y)
elif x == y:
```

The Coursework Brief

The Coursework component of the Computer Science Leaving Certificate exam is worth 30% of your overall grade (90 marks). You will complete this project in 6th year, usually during a 10 week period from January to March.

Within the timeframe you must complete

1. The practical element
2. A comprehensive report of the project
3. A short video explaining your work

It is beneficial to take a deeper look into the 4 ALTs (Applied Learning Tasks) which are a part of the Computer Science Course.

Each year, the brief will outline a problem that you must attempt to solve using your background in computer science. Examples of previous coursework briefs include:

- Climate change
- COVID-19

The project you complete may be based on one or multiple of the ALTs (E.g the 2022 brief is to be focused on ALT 1 and ALT 4).

ALT 1: Interactive Information Systems	Web design (HTML + CSS) Relational databases Interactive elements (JavaScript) User-centred design
ALT 2: Analytics	Collecting data Cleaning and storing raw data (Abstraction) Analysing data Drawing conclusions
ALT 3: Modelling and Simulation	Creating computer model that simulates a problem Modifying the model Analysing emergent behaviours
ALT 4: Embedded Systems	Building an embedded system (Microprocessor) Inputs and Outputs (Digital and Analogue) Measuring data using sensors (Inputs)

Advice for a successful coursework project:

Don't overlook the planning process! Make sure that you hit the brief and create something that satisfies the requirements that are specified.

Keep track of what you are doing every day using some form of documenting system. This can be as simple as a spreadsheet/ word doc or even in a notebook. This will be vital when you come to writing up documents for the brief as you can see when everything was done and how long it took you to deal with certain aspects.

Keep a record of issues/ problems. This is so important as you can show the examiner that you were learning during the process. Every time you come up against an obstacle (particularly in the coding process), screenshot it and make a note. Similarly, when the issue is fixed, make a note of what you did to fix the issue and take a screenshot of the new and improved code.

Manage your time! You may think 10 weeks is plenty of time to complete the entire project, but the time flies and you will have to balance work on the project with all other subjects and homework/assignments. Therefore, make plans and dedicate time to the project every week.

By dedicating **enough time to the project** from the beginning, it will allow you to make further adjustments and improvements to your work later. These changes will boost your grade and impress the examiner.

Organise your files! Keep all your files in one place. This can be on your computer itself or on the cloud. This will prevent panic when trying to find or use files for the project.

Keep a word document with links to any websites or resources you use to help you in the creation of the project or in your research. These can then be copied into the references section of your report.

Timings

In terms of timing for the Computer Science exam, remember:

The written component (Section A and B) is **1 hour 30 minutes** (90 minutes).

- In Section A, you are required to answer 12 short questions for 60 marks.
- In Section B, you will need to answer 3 long questions for 70 marks.

As the amount of marks for both sections are similar, the easiest method of timing would be to dedicate **45 minutes for each section**. However, it might be necessary to give an extra 5 - 10 minutes to the long questions as they are worth an extra 10 marks, but you can see how you are for time in the exam.

Take your time with reading the questions and don't jump straight into answering without understanding what you are being asked.

The computer element (programming) is **1 hour** in length (60 minutes)

This section does not have any specific timing to follow, however, I recommend keeping an eye on the clock so that you are not spending too long on one particular part of the question. You are better off attempting all the parts and not completing the code for them all, rather than leaving certain questions out completely.

Final tips

Get into the habit of practicing programming. The earlier you start to do this, the better!

Don't neglect the theory! In this subject, it can be easy to fall into the trap of dedicating more time to your project and coding, however, remember that the theory is worth over 40% of the marks, so it is essential for those of you looking to get a top grade.

Get the basics right. If you are looking for a high grade in computer science, you need to know the simple stuff. Data types, binary, hexadecimal, CPU, RAM vs ROM and logic gates are all great places to start. Once you are confident with these, you can move onto more challenging concepts such as modelling, artificial intelligence and ethics.

Do your best with your coursework - keep organized and on schedule. This way you already have 30% before even sitting the exam.

Use online resources - you can access past papers as well as topic quizzes on Studyclix which can help you to revise and practice for the programming and written components of the exam.

Best of luck in the exam!
You will be great.
☺