Make the Grade

Ashby School
Empowered to Learn

Name:

Target Grade:

Recent Grades:

Exam Dates **Exam Board: OCR**

Wednesday 15th May—Paper 1—Computer Systems

Tuesday 21st May—Paper 2—Computational thinking, Algorithms, and Programming

Maths Faculty

Computer Science

Additional Notes and Resources Available

Lessons In the student shared area > de-sign_technology < computing

Free Craig 'n' Dave videos on YouTube

Bespoke revision sessions:

Each student has their own copy of their revision timetable.

(Mondays and Wednesdays every week in M10: 3:05pm — 4:30pm)

Exam Content

Paper 1 (50%):

1.1—Systems Architecture

1.2—Memory and Storage

1.3 Computers networks, connections and protocols

1.4—Network Security

1.5—Systems Software

1.6—Ethical, Legal, cultural and environmental impacts of technology

Paper 2 (50%):

2.1 - Algorithms

2.2—Programming Fundamentals

2.3—Producing Robust Programs

2.4—Boolean Logic

2.5—Programming Languages and Integrated Development Environments

Notes:

20-30 minutes of revision every couple of days is plenty!

Remember to revise key terminology, there's lots of it.

Make the Grade

Ashby School

Empowered to Learn

Revision Programme 2024

Week start	Topic	*Revision Tasks*
5th Feb	1.1 Architecture of the CPU	Draw a diagram of the CPU labelling all the parts of the CPU Create a poster of factors affecting CPU performance
12th Feb	1.2.1 Primary Storage	Create flashcards for each of the 4 primary storage components labelling their function and why they're needed. Create a comparison table of the characteristics of primary storage compared to secondary storage
19th Feb	11.2.2 Secondary Storage	Create a comparison table of the characteristics of primary storage compared to secondary storage Create flashcards for the 3 main types of secondary storage Read through notes about when each type of secondary storage is best for different scenarios
26th Feb	1.2.3 Units	Practice converting between bits, bytes, kilobytes, megabytes, Gigabytes, Terabytes Practice converting between binary and denary, binary and hexadecimal and denary to hexadecimal. Read through notes about why hexadecimal is useful over binary
4th March	1.2.4 Data Storage	Create 3 posters, one for each type of data being stored: text, sound and images. On the posters label how the data gets stored Learn and revise the key terminology for this topic
11th March	1.2.5 Compression	Revise what compression is and why it's useful Create flashcards about lossy and lossless compression and how they differ
18th March	2.1.1 Computational Thinking	Learn the key terminology: abstraction, decomposition, algorithmic thinking
25th March	2.1.2 Designing Algorithms	Practice programming challenges to keep programming skills fresh on replit or trinket
1st April	2.1.3 Searching and Sorting Algorithms	Create a poster for each type of algorithm: binary search, linear search, bubble sort, merge sort, and insertion sort Learn how each one searches/sorts data
8th April	1.3.1 Networks and Topologies	Revise key terminology, there is a lot of key terms in this topic. Create a poster about each type of topology and what it looks like Create flashcards about the hardware involved in computer networks: switches, routers, ethernet cable etc.

Make the Grade

Week start	Topic	*Revision Tasks*
15th April	1.3.2 Wired and wireless networks, protocols and layers.	Create flashcards for each of the protocols and learn what each one is involved in Learn key terminology
22nd April	1.4.1 & 1.4.2 Threats and vulnerabilities computer systems and networks	Learn key terminology and create flashcards about each type of attack. Create flashcards about how to prevent each kind of attack.
29th April	1.5 & 1.6 Systems software and ethical, legal problems	Create flashcards on the different legislation for technology Do some light reading of issues in technology on BBC https://www.bbc.co.uk/news/technology this will help to raise awareness of issues in technology.
6th May	2.2 Programming fundamentals	Practice programming challenges using replit or trinket. Practice using loops (for and while), if else statements, generating random numbers, using arrays (1D and 2D), creating variables, reading and writing to files. Look at what SELECT, FROM, WHERE and what they do in a database
13th May **Paper 1**	2.3 & 2.4 Producing robust programs and Boolean logic	Create a poster about how you can create robust programs Practice combining logic gate symbols together and create simple diagrams
20th May **Paper 2**	2.5 Programming languages and IDE's	Create flashcards contrasting high-level and low-level languages Revise key terminology