Energy Consumption

Computer systems consume huge amounts of energy.

Computers are becoming more efficient so use less energy.

We are using computers more than before, meaning higher energy use.

Manufacture

Building uses up natural resources, which are limited in supply.

A computer requires 10x its weight in fossil fuels to make.

Replacement Cycle

How long is a device lifespan before it breaks or becomes too old to be used?

Organisations may set a 3, 5 or 10 year replacement cycle.

Disposal

Computers contain harmful & so must be disposed of carefully.

Often sent to countries with lower disposal standards.

People there will go through waste to find metals they can sell on.

This exposes them to dangerous chemicals .

Ethical & Legal Issues with Personal Data

Many areas are constantly being debated with people having different views.

Computer systems hold large amounts of personal data.

Organisations also collect data such as history of our Internet activity.

Smart devices collect data such as our voice, video & activity.

Governments want access to this data to prevent crime. Is this right?

Should organisations be allowed to collect this much data, and how do they use it?

Issues with AI, Machine Learning & Robotics

These fields are growing & changing rapidly, bringing up new ethical & legal issues. Often, there is not yet a right or wrong answer, but it is important to consider these issues nevertheless.

Accountability

All choices have consequences, sometimes significant in legal, financial or safety terms.

Who is responsible? The person operating the system, the person who produced it, or the system itself.

Legal Liability

Actions taken by systems may have legal consequences.

Who should be accountable in these cases? The system owner, the manufacture or the system itself?

Safety

Ensuring that systems do not cause harm.

Fail safe processes mean the system will default to a safe state.

How should a system act in cases where some harm is unavoidable?

Algorithmic Bias

The design of an algorithm may favour certain groups.

Should a robot choose to save a young person over an older person?

Should a self driving car swerve into one person to avoid hitting four?

Are these things ethical?

Copyright, Designs & Patents Act 1988	Data Protection Act 2018	Computer Misuse Act 1990	Network & Information Security Directive 2016

Data Protection

Data protection law lays out legal rights & responsibilities for data.

Gives rights to those who own data, and places responsibilities on those who use it.

UK law is very robust.

The Data Protection Act 2018 is the UK implementation of the EU GDPR (General Data Protection Regulations)

Before this, the Data Protection act already provided protection, but it has now been updated to be even more robust.

Ownership

Who owns data supplied to a company or organisation?

UK law makes clear that you are the data owner, whilst those who hold the data are data stewards.

Data Stewards have obligations under law to keep data secure, up to date, and delete it when they no longer need it.

Right to erasure, allowing us to tell companies to delete data we don't want them to hold.

There are exceptions to this, such as the Police for crime prevention.

Misuse

Data can be misused by hackers, phishing scams or viruses.

Systems should have processes in place to prevent this.

Consent

Data protection laws require positive consent for data collection.

Data cannot be collected automatically without consent.

It is insufficient to use opt out processes, requiring people to tick a box saying they do not want their data to be used.

Consent must be easy to understand & obvious.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Issues

Malware

Viruses

A program designed to disrupt or damage a computer system.

May cause the system to stop functioning or lose data.

Worms

A computer program which makes copies of itself.

Works by itself rather than attaching to another program.

Sends out the copies to try & infect other systems.

Once installed will damage the system or attempt to steal data.

Trojans

Malicious software hidden in what seems to be a normal program.

Free games or music often contain trojans.

Once installed will damage the system or attempt to steal data.

Ransomware

Encrypts data & demands money to unlock it.

Leaves systems unusable causing huge organisational impact.

Key Loggers

Records all keystrokes & activity.

Capture usernames, passwords, and any other data typed.

Hacking

Unpatched or Outdated Software

Software patches contain fixes for bugs & security flaws.

Software should include features to automatically apply patches.

Programmers may stop updating software, leaving it unsupported.

Hackers take advantage of bugs to gain access to systems.

Hackers look at patches to see what bugs are fixed & try to exploit them.

Out-Of-Date Anti-Malware

Anti-malware requires definitions to be kept up to date to be effective.

Hackers can target systems with outdated anti-malware.

Social Engineering

Targets people rather than systems.

Attempt to manipulate people into handing over data.

Strong policies & user training can help.

Phishing tries to get people to hand over data using messages which look like they have come from a trusted party.

Blagging uses a fake story to try & get a person to hand over money or information.

Shoulder surfing is watching someone enter their password.

Copyright

Protection for works such as music, books or software.

Automatic when work is created & does not need to be registered.

Does not last forever & will expire.

Illegal to share or copy without the owner's permission.

Prevents others from selling copies of the work

Patents

Allows someone to register ownership of an invention or process.

Can apply to different parts of a device or system such as the interface.

Protects the idea & prohibits people from copying it.

Trademarks

Protection for a logo, name or phrase.

Must be registered.

Allows companies to protect what makes their brand distinctive.

Provides protection for software names & logos & prevents someone from attempting to sell their own version.

Also prevents people using names which are too similar & designed to confuse

Licencing

Defines how software may be used.

Prevents people from using the software in a way the owner would not want.

Proprietary licenses are expensive & must be purchased from a company but are often more secure & include software support & updates.

Open source licenses are free & available to anybody, but can be less secure & harder to find support for when something goes wrong.

Anti-Malware

- Scans for, removes, & protects against malicious software.
- Includes anti-virus, anti-phishing, & anti-spyware.
- Can be set to scan manually, at a certain time, or when files are accessed.
- Will attempt to stop malware before it can be installed.
- Scans files against a list of malware (called definitions).
- These definitions must be kept up to date.
- Cannot protect against new malware until definitions are updated.
- Organisations should run this software on their systems & networks.

Backup & Recovery Procedures

- The process used for backing & restoring.
- Backups allow organisations to recover data which may have been lost or damaged.
- Allows people to have all the information to h& when dealing with an incident.
- It is important that recovery is tested.
- Backups will contain sensitive data, so it is important that they are kept secure

Encryption

- Changes data so it can't be read by anyone other than the intended recipient.
- Encrypted with an encryption algorithm & decrypted with a decryption algorithm.
- An encryption key is a string of characters used to encrypt data.
- Encrypting data before transmission helps security, as the message cannot be read without the key.
- Data can also be encrypted when stored meaning if the device is stolen the data cannot be read.
- Organisations may be required to use encryption by policies, laws, or contracts.

Acceptable Use Policy (AUP)

- Rules for how systems & networks may be used.
- Users should read & agree before using the system.
- Discourages users from actions which may damage the system.
- Allows the organisation to discipline those who use systems inappropriately.
- Provides clear guidance to users on what they can & cannot do.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Issues

The Fetch-Decode-Execute Cycle

1. The memory address held in the program counter is copied into the MAR.
2. The address in the program counter is then *incremented* (increased) by one. The program counter now holds the address of the next instruction to be fetched.
3. The processor sends a signal containing the address of the instruction to be fetched along the address bus to the computer's memory.
4. The instruction held in that memory address is sent along the data bus to the MDR.
5. The instruction held in the MDR is copied into the CIR.
6. The instruction held in the CIR is decoded and then executed. The results of processing are stored in the ACC.
7. The cycle then returns to step one.

Von Neumann Architecture

- Data and instructions are stored in binary.
- Data and instructions are stored together in RAM.
- Instructions are fetched from RAM one at a time in order.
- The CPU decodes and executes an instruction, before fetching the next instruction.
- The cycle continues until no more instructions are available.

A CPU using Von Neumann architecture have five special registers:

- Program counter - holds the memory address of the next instruction to be fetched.
- Memory address register (MAR) - holds the address of the current instruction.
- Memory data register (MDR) - holds the content at the address held in the MAR.
- Current instruction register (CIR) - holds the instruction that is currently being decoded and executed.
- Accumulator (ACC) - holds the results of processing.

Control Unit (CU)

Fetches, decodes, and manages the execution of instructions Issues control signals to control hardware Moves data around the system

Arithmetic Logic Unit (ALU)

Performs arithmetic and logical operations. Where calculations are done and where decisions are made.

Registers

Small amounts of high speed memory in the CPU. Used to store small amounts of data that are needed during processing.

Cache

A small amount of high speed memory In the CPU. Used to temporarily hold data the CPU will reuse. Allows for faster processing since as the CPU need not wait for data to be fetched from RAM.

Clock

Used to coordinate all the computer's components. Sends out a regular electrical pulse to do this. The frequency of the pulses = clock speed, measured in hertz. Higher clock speed = greater number of instructions which can be performed at a time.

Buses

High speed internal connections. Used to send control signals and data between the processor and other components.

- Address bus - carries memory addresses from the CPU to other components.
- Data bus - carries data between the CPU and other components.
- Control bus - carries control signals from the CPU to other components

Secondary Storage

Used to store programs/data when the computer is switched off.

Non-volatile data is retained with the computer is switched off.

Not all computers require secondary storage.

Embedded computers do not need to store data when power is turned off.

Solid State Devices

Use flash memory to store data indefinitely.

Have faster access times than other devices

Because they have no moving parts, are more durable.

More expensive so tend to be smaller in capacity.

Require little power, so used where battery life is a consideration.

Portable due to their small size and durability.

Optical Devices

Use a laser to scan the surface of a spinning disc.

The disc surface is divided into tracks, with each track containing lands and pits.

Lands reflects the laser light back; pits do not.

ROM (Read Only Media) cannot be overwritten.

Read (R) media is blank, can only be written to once, but read many times.

Read/write (RW) media can be

Magnetic Devices

Use magnetic fields to magnetise individual sections of a spinning disc.

Each section represents one bit.

A read/write head moves across its surface.

Fairly cheap, high in capacity and durable.

Susceptible to damage if dropped.

Vulnerable to magnetic fields.

Embedded Systems

A small computer which includes hardware and software, designed to control a specific device.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Computers

Abstraction

Using symbols and variables to represent a real-world problem using a computer program and removing unnecessary elements

Advantages:

- Allows the creation of a general idea of how to solve the problem.
- Provides focus on what actually needs to be done.
- Provides a simple view of the problem

Sequencing

Breaking down complex tasks into simple steps. The order of steps matter Step by step progress through a program

Advantages:

- Each line follows the next.
- Can create simple programs very quickly.
- Easy to follow for a small program.

Disadvantages:

- Not very efficient.
- Difficult to follow with large programs. Hard to maintain.

Decomposition

Breaking down large problems into a set of smaller parts. There are several different approaches, and not one single right way to do this.

Advantages:

- Smaller problems are easier to solve
- Each part can be solved independently
- Each part can be tested independently
- The parts are combined to produce the full problem.
- Allows each smaller problem to be examined in more detail

Flowcharts

Created to represent an algorithm.

Show the data that is input, and output.

Show processes that take place.

Show any decisions and repetitions that take place.

Lines show flow through the chart.

Shapes represent different function

Pseudocode

Uses short English words and statements to describe an algorithm.

Generally, looks a little more structured than normal English sentences.

Flexible.

Less precise than a programming language.

Sub Programs

Small programs which form part of a larger program.

Procedures are sets of instructions stored under a single name (identifier).

Functions are like procedures but will always return a value to the main program.

Parameters are values passed into a sub program. These are referred to as arguments when calling the sub program.

Advantages:

- Used to save time and simplify code.
- Allows the same code to be used several times without having to write it out each time.
- Usually small in size, so easier to write and test.
- Easy for someone else to understand.
- Can be saved separately as modules and used again in other programs.
- Saves time because code that has already been written and tested can be reused.

Evaluating Fitness for Purpose and Efficiency

Fit for Purpose - meets the original purpose and requirements the code was designed for. Provides the expected outputs. Test tables help to examine the values at each stage and check code is working as expected.

Efficient – the amount of time and resources needed to run a particular program.

Steps which improve efficiency:

- Using repetition (loops) to reduce the amount of code
- Using arrays instead of declaring many individual variables
- Using selection statements which only make comparisons until a solution is reached

Programming Constructs

Variables

A single location in memory in which data may be stored.

Different types e.g. string, decimal, etc.

Allows the program to store data such as an input for later use.

Constants

A fixed value used by the program such as pi.

Allows easy use of fixed values without having to store them in the program.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Computational Thinking

Arithmetic Operators

- Addition +
- Subtraction -
- Multiplication *
- Division /
- MOD Modulus (the remainder, e.g. 12 MOD 5 gives 2)
- DIV Quotient (integer division, e.g. 21 DIV 5 gives 4)
- Exponentiation (to the power of, e.g. 3^3 gives 27)

Comparison Operators

- == Equal to
- != Not equal to
- < Less than
- <= Less than or equal to
- > Greater than
- >= Greater than or equal to

Boolean Operators

- AND - two conditions must be met for the statement to be true
- OR - at least one condition must be met for the statement to be true
- NOT – inverts the result, e.g. NOT(A AND B) will only be false when both A and B are true

Selection

Allows the program to make decisions.

Uses conditions to change the flow of the program.

Selection statements may be nested one inside another.

Selection statements perform comparisons sequentially, so the order is important.

SELECT CASE has less typing but is less flexible.

```
num = 30
if num > 10:
    print("True")
else:
    print("False")
```

Arrays

An ordered collection of related data.

Each element in the array has a unique index, usually starting at 0.

All elements must be the same type of data.

Arrays are usually a fixed size.

1 Dimensional arrays are like a simple list, each element needs a single index number. **Fruits[1] references element 1 in the 1D Fruits array.**

2 Dimensional arrays are like tables, with each element needing two index numbers.

2 Dimensional arrays are usually used to store properties of objects, with objects in rows and properties in columns. **Tools[0,2] references element 0,2 in the Tools array.**

```
names = ['jack', 'mary', 'robert', 'owen', 'maggie', 'timothy']
#           0       1       2         3        4         5

print(names[1])
#prints the index 1 - which is mary
```

Count controlled:

- Repeats the same code a set number of times.
- Uses a variable to track how many times the code has been run.
- This variable can be used in the loop.
- At the end of each iteration the variable is checked to see if the code should be run again.
- FOR sets how many times the code should be repeated.
- NEXT tells the code to return to the start of the loop.
- STEP sets how the variable should increment.

```
for i in range(3,30):
    print("Iteration:", i)
```

Condition Controlled Iteration

- Uses a condition to determine how many times code should be repeated.
- While loops will run whilst a condition is met and use the statements WHILE and ENDWHILE.
- Repeat loops will run until a condition is met and use the statements REPEAT and UNTIL.

```
x = 1
while x <= 5:
    print (x * 5)
    x = x + 1
print("done")
```

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Python Programming

Measuring Storage

Size	Binary Unit
8 bits (b)	**1 byte**
1024 bytes (B)	**1 kibibyte**
1024 kibibytes (KiB)	**1 mebibyte**
1024 mebibytes (MiB)	**1 gibibyte**
1024 gibibytes (GiB)	**1 tebibyte**
1024 tebibytes (TiB)	**1 pebibyte**

Hexadecimal

A number system made up of 16 symbols, 0-9 and the letters A-F.

- Also known as Base 16
- Useful because large numbers can be represented using fewer digits.
- Easier to understand, write and check than binary.
- Commonly used for colour values and MAC addresses.

Binary

A number system made up of 0 and 1 used by computers to store and represent data such as numbers, sound, and graphics.

- Also known as Base 2.
- Computers use binary because the CPU contains transistors, which are either on or /off.

Binary States

The number of states which can be represented in a given number of bits using binary.

- To calculate the number of states use the formula 2 n where n is the length of the pattern
- Limits on the number of bits available affect how much data can be stored

Converting Between Bases

Binary to Denary

1) Draw your conversion table.

128	64	32	16	8	4	2	1
1	1	0	0	1	1	0	0

2) Write the binary number in the conversion table.

3) Add together all numbers with a 1 beneath them

128
64
\+ 8
4
204

11001100 in binary is 204 in denary

Denary to Binary

1) Draw your conversion table.
2) Is the number higher than the first column in the table?
   a) If so, put a 1 in that column and work out the difference.
   b) If not, put a 0 in that column.
3) Repeat the step above with the difference.
4) Keep going until the difference is 0, put a 0 in any empty columns.
5) Read the number from the bottom row of the table.

128	64	32	16	8	4	2	1
1	1	0	0	1	0	0	0

200 – 128 = **72**

72 – 64 = **8**

8 – 8 = **0**

200 in denary is 11001000 in binary

Denary to Hex

1) Divide the denary number by 16 and write down both the answer and the remainder.
2) Divide the answer by 16 again. Write down both the answer and the remainder.
3) Keep going until you reach an answer of 0.
4) Read the remainders from bottom to top.
5) Convert each remainder to hex.

62 ÷ 16 = 3 R 14

3 ÷ 16 = 0 R 3

3 14

3E

62 in binary is 3E in hexadecimal

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Binary Maths

Addition

- 0 + 0 = 0
- 1 + 0 = 1
- 1 + 1 = 10 (binary for denary 2)
- 1 + 1 + 1 = 11 (binary for denary 3)

Multiplication (using binary shifts)

- Move the digits to the left and fill the gaps after the shift with 0.
- Move 1 place for X2, 2 places for X4 etc.

Division (using binary shifts)

- Move the digits to the right and fill the gaps after the shift with 0.
- Move 1 place for X2, 2 places for X4 etc.

Binary to Hex

1) Draw two separate conversion tables.
2) Write the binary number across both tables.
3) For each table, add up the numbers which have a 1 beneath them.
4) Convert each number to hexadecimal.

8	4	2	1
0	1	1	0

4 + 2 = 6

6

8	4	2	1
1	1	0	1

8 + 4 +1 = 13

D

01101101 in binary is 6D in hexadecimal

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Data

Types of Network

LAN - Local Area Network

Confined to a single location.

Owned and maintained by a single organisation.

Used by organisation such as schools and small businesses.

Connected by cables or wireless.

WAN – Wide Area Network

Covers a wide geographical area.

Used by organisations with several different sites such as banks or universities.

Allows all the sites to communicate and share data.

Uses national or international long distance media.

The Internet

A vast WAN covering the entire world.

An Internet Service Provider (ISP) provides access to the Internet.

Routers provide an interface between the Internet and the customer via the ISP.

Mesh Network

No central connection point with each device connecting directly to others. Full mesh networks have every device connected to every other device. Partial mesh networks have each device connected to several others but not necessarily every other device.

Advantages:

Messages can be received more quickly.

Messages have many possible routes they can take.

Multiple connections mean that no device should be isolated

Each device can talk to more than one node at the same time.

Devices can be added without interruption.

Disadvantages:

Can be impractical and expensive to setup.

Require a lot of maintenance

Advantages and Disadvantages of Networks

Advantages	Disadvantages
Software and files can be shared.	Cost, additional equipment is needed.
Hardware such as printers can be shared	Additional management by specialist staff.
Users can communicate via email, chat, etc.	Spread of malware.
Centralised maintenance and updates.	Potential for hacking.
Centralised security.	
User monitoring.	
Different users can be given different access rights or permissions.	

Wired v Wireless Networks

Wired Networks

Using fibre or copper cable to connect devices in the network together. Fibre cable provides a faster connection and can cover longer distances.

Advantages:

- Faster data transfer
- Less likely to suffer from interference
- More difficult for unauthorised users to intercept data

Disadvantages:

- Expensive to install or reconfigure
- Harder to move devices so less flexible

Wireless Networks

Using radio signals or infrared light to connect devices in a network together.

Advantages

- Devices can easily be added
- Users can move around freely and stay connected

Disadvantages:

- Signals have a limited range.
- Can suffer from electromagnetic interference from other devices.
- Signals can also be blocked by walls or other objects.
- Each wireless access point (WAP) only has so much bandwidth.

Star Network

All nodes are connected to one or more central switches. Often used with wireless networks.

Advantages:

Every device has its own connection so failure of one node will not affect others.

New devices can be added by simply connecting them to the switch.

Usually have higher performance as a message is passed only to its intended recipient.

Disadvantages

If the switch fails it takes out the whole network.

Requires a lot of cable so can be expensive.

Bus Network

All devices are connected to a single cable (called the bus) with a terminator is at each end of the cable.

Advantages:

Easy to install extra devices.

Cheap to install as it doesn't require much cable.

Disadvantages

If the cable fails or is damaged the whole network will fail.

Performance becomes slower as additional devices are connected due to data collisions.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Networks

Computational Thinking	Data	Computers	Networks	Issues
Abstraction	Amplitude	Arithmetic & Logic Unit	Bandwidth	Artificial Intelligence
Constants	ASCII	Cache	Copper Cable	Copyright
Decomposition	Base	Control Unit	Ethernet	Data Protection
Flowchart	Binary	CPU	Fibre Optic Cable	Energy Consumption
Functions	Capacity	Embedded Systems	Internet	Machine Learning
Operators	Colour Depth	FDE Cycle	Latency	Malware
Parameters	Compression	Main Memory	Local Area Network	Manufacture
Procedures	Hexadecimal	Registers	Topologies	Personal Data
Sequencing	Resolution	Secondary Storage	Wide Area Network	Robotics
Variables	Sample Rate	Von Neumann Architecture	Wi-Fi	Social Engineering

Read over the keywords and try to define what they mean.

Python Key Terminology			
Algorithm	Data types	Integers	Robustness
Annotations	Efficiency	Iteration	Runtime Errors
Arithmetic	Fitness for purpose	Layout	Selection
Arrays	Flowchart	Logic Errors	Strings
Assignment	Global/Local Variables	Logical Operators	Subprograms
Authentication	Identifier	Modulus	Syntax Errors
Boolean	Indentation	Output	Test Data
Condition-controlled	Input	Pseudocode	Validation
Count-controlled	Inputs	Real	Verification
Data Structures	Integer Division	Relational Operators	White Space

Amend (1-6 marks)

Requires changes or additions to code, or deletions or rearrangement of code.

Calculate (2-4 marks)

Obtain a numerical answer, showing relevant working.

If the answer has a unit, **this must be included.**

Complete (2-6 marks)

Requires the completion of a table/ diagram/algorithm (in any notation).

Convert (2-6 marks)

Requires changing information from one symbolic representation to another representation.

Requires amending to provide new functionality/facility.

Draw (2-6 marks)

Produce a diagram/image either using a ruler or freehand.

Requires labelling/annotation to express meaning.

Used when symbolic representations need to be manipulated.

Write (1-6 marks)

Requires creation/manipulation of an artefact using a subject-specific notation.

State/Give/Name (1 mark)

Requires recall of one or more pieces of information.

Construct (2-4 marks)

Requires creation of an artefact using subject-specific symbolic representation, rules and syntax.

Unit 1 Command Words

Amend (5-15 marks)

Requires changes or additions to code, or deletions or rearrangement of code.

Write (5-15 marks)

Requires creation/manipulation of a program using a high-level programming language

Unit 2 Command Words

These here are the more common command words and should be looked at more often!

Define (1-2 marks)

When the meaning of a term is expected but there are different ways of how this can be described.

Describe (2-4 marks)

To give an account of something.

Statements in the response need to be developed, as they are often linked, but do not need to include a justification or reason.

Discuss (6 marks)

Identify the issue/situation/problem/ argument that is being assessed within the question.

Explore all aspects of an issue/ situation/problem/argument.

Investigate the issue/situation by reasoning or argument.

Explain (2-4 marks)

An explanation requires a justification/exemplification of a point.

The answer must contain some element of reasoning/justification, this can include mathematical/ logical explanations.

The mark scheme will have marking points which are linked.

Identify (1 mark)

Requires key information to be selected from a given stimulus/ resource/set of options.

Kettlethorpe HIGH SCHOOL

Computer Science

Year 11| Command Words