Janet Allison
Martin McKinney
Adrian Moore

CCEA | AS

DIGITAL TECHNOLOGY

ANSWERS

These answers are for use in conjunction with the book
Digital Technology for CCEA AS Level,
ISBN: 978 1 78073 116 2

COLOURPOINT
EDUCATIONAL

Publisher's Note: These answers have been written to help students preparing for the AS Digital Technology specification from CCEA. While Colourpoint Educational and the author have taken every care in its production, we are not able to guarantee that it is completely error-free. Additionally, while the mark scheme has been written to closely match the CCEA specification, it is the responsibility of each candidate to satisfy themselves that they have fully met the requirements of the CCEA specification prior to sitting an exam set by that body. For this reason, and because specifications change with time, we strongly advise every candidate to avail of a qualified teacher and to check the contents of the most recent specification for themselves prior to the exam. Colourpoint Creative Ltd therefore cannot be held responsible for any errors or omissions in this mark scheme or any consequences thereof.

Answers

Chapter 1: Systems Development

1. Any system has the following three characteristics: a purpose; a need; be made up of various components. In an educational organisation, for example a school, examples of these three characteristics are as follows. Purpose: to record pupil attendance and provide staff with personal information and results of pupils. Need: to monitor pupil attendance and provide regular progress reports to parents. Components: electronic register to store attendance, and a database to store student records.

2. Reasons for the software crisis: any four of the following.
   - People had little historical experience of similar work, so time and cost estimates were often inaccurate.
   - The tools used to develop software were not very sophisticated and as a consequence software took much longer to produce.
   - No tools were available to support the early identification of errors which slowed down the detection and correction of errors.
   - Systems did not meet user requirements.
   - Specialist testing staff were not part of the software development process, so there was no formal process for testing of software for correctness.

   The software industry's response was as follows.
   - A systematic approach to the development of software was required.
   - To develop ways to develop software systems on time and within budget.
   - To develop methods to ensure that software met user requirements.
   - Advances in technology (hardware and software) needed to be utilised.

3. Software engineering is the application of a structured approach to the design, development and maintenance of software systems.

4. A computer system must be able to (any three of the following):
   - receive data (raw facts),
   - process the data received,
   - store the data received and any data as a result of processing,
   - present the processed data (information),
   - communicate data between related sub-systems.

5. (a) Data: This is the raw facts which are provided by the user to the computer system

   User Interface: This is how the user interacts with the computer system. It allows the user to enter data (raw facts) into the system.

   Processes: These are the tasks carried out by the computer system. A process takes the raw facts and converts it into information.

   (b) (i) Examples of information a supermarket needs:
   - Details of each item it sells.
   - Quantity in stock of each item.
   - When it is necessary to re-order an item.
   - Reports detailing best/worst selling items.
   - Reports detailing items which may soon reach their "sell-by" or "best before" dates.

   (ii) Data:
   - ID/name of every item in the supermarket.
   - Quantity of item in stock.

   (iii) Processes:
   - Add new items and delete items from the computer system.
   - Change the quantity of items in stock (when new stock arrives and when items are sold).
   - Produce reports.

   (iv) User interfaces
   - Present details of stock items
   - Edit stock levels
   - Permit the user to enter details of new stock items & delete stock items

6. • Systems Analyst. Role: to identify the user and data processing requirements. Duties: to carry out a feasibility study and produce a feasibility report; liaise with the client (user); undertake some fact-finding (using interviews, observation, questionnaires or document sampling); define the system specification.
   • Programmer:. Role: to develop and test the system. Duties: write or amend the source code (using a high-level programming language) based on module specifications; document the code to aid future maintenance; debug the code to identify any errors that may have been found during testing; ensure integration of existing systems (if they exist); test the new system using a test plan; write operational documents; work closely with the systems analyst and project manager to ensure the user requirements are met within the expected time frame.
   • Project Manager: Role: to plan, manage and co-ordinate the system; Duties: plan the project's schedule; split the project into suitable tasks/sub-tasks; allocate resources such as personnel, hardware and software to each task/sub-task; identify risks to the project; monitor progress of the project; ensure the project is delivered on-time and within budget; report back to the client.

Chapter 2: Analysis

1. (a) Direct data source – a data source designed and used for a specific purpose. Indirect data source – a data source that is already available and is used for a purpose for which it was not originally intended.
   (b) Benefit – it is designed to acquire specific information. Drawback – it can be expensive.
   (c) Benefits – it can be very cost effective as the data has already been collected/processed/paid for. It should be immediately available as the data has already been collected using the data source. Drawbacks – the source is usually designed for a different purpose so it may not provide exactly the data required and additional filtering or processing may be required, for example it may be difficult to find out how many places sell newspapers apart from through news agencies.
   (d) (i) Indirect (ii) Direct (iii) Indirect (iv) Indirect

2. (a) Purpose of fact finding is to:
   • understand the existing system in terms of inputs, processes and outputs,
   • identify strengths and weaknesses of the system,
   • identify opportunities for improvement or addition of new features.
   (b) **Interviews**: main features:
   • Analyst asks users questions to confirm their understanding of the system.
   • Fills in gaps in the analyst's basic understanding.
   • Allows the analyst to delve deeper into certain aspects of the system in which the interviewee is an expert by using follow-up questions.
   • Questions may be asked on a one-to-one basis or in groups.
   • Questions can be structured or unstructured.
   Benefits:
   • The interview can be fine-tuned for the particular user.
   • Follow up questions can be asked.
   Drawbacks:
   • Users may 'act up' under the pressure of an interview.
   • It may be difficult to select the users to interview.
   • Interviews can be very time consuming.

   Questionnaires: main features:
   • The same set of questions are asked to everyone.
   • Questions may be structured (using radio buttons, checkboxes etc) or free text response.
   • Completed anonymously.
   Benefits:
   • The same set of questions can be given to a large cross section of users at the same time.
   • The anonymous process may elicit honest answers.
   Drawbacks:
   • The questions are set beforehand and a question cannot be changed in the light of the user's response.
   • There tends to be a low return rate.

   Observation: main features:
   • Representative sample of employees are selected.
   • Analyst "shadows" the users as they carry out their daily tasks.
   • Analyst can see the time required for and constraints for each process.

- Identifies data required, processes involved and data produced.

Benefits:

- The system's working can be studied first hand.

Drawbacks:

- Users may 'act up' under observation.
- It may be difficult to select the users to observe.

Document Sampling: main features:

- Analyst reads through samples of essential documentation, for example invoices, sales receipts, orders, reports.
- Identifies key users of the system and the volume of data being processed.

Benefits:

- It can identify key users of the system.
- It can identify where a particular system fits into the entire organisation.

Drawbacks:

- The organisation may not have up-to-date/complete documentation for the system.
- Can be very time consuming.

(c) A functional requirement is something 'the system must do' while a non-functional requirement describes 'additional conditions which the system should meet'. Possible examples of functional requirements of the payroll system:

- Add/edit/remove employees.
- Enter the number of hours worked per week.
- Edit the hourly pay rate.
- Backup payroll details.

Possible non-functional requirements of the payroll system:

- Security issues/access control.
- Reasonable response time.
- Reliable.

(d) Purpose of a DFD:

- To provide a visual/graphical presentation.
- To provide different levels of the system.
- To show how a system interacts with external entities.
- To identify processes/data stores.

(e) Context diagram for placing orders:

3. A: External Entity; B: Data Flow; C: Process; D: Data Store.

4. Context diagram:

Level 1 DFD:

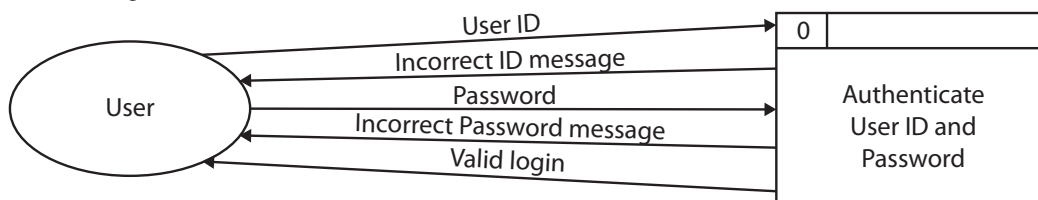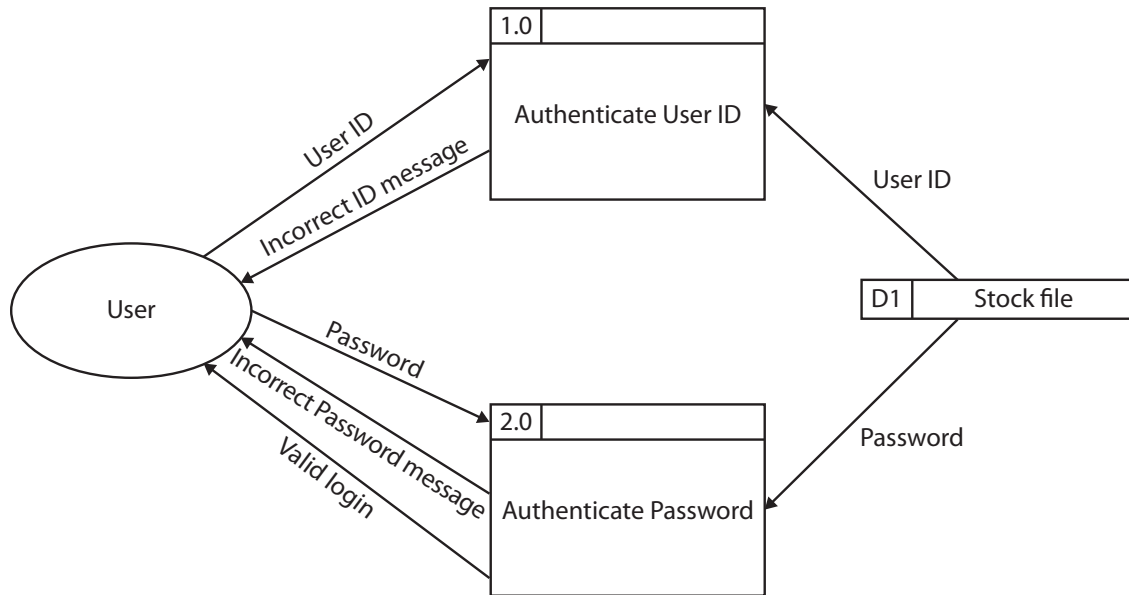5. A Level 1 DFD shows the system in more detail than the context diagram. The context diagram has been exploded into more detail describing the sub-systems that make up the initial system. Data stores are often not shown in the context diagram, but are identified in a Level 1 diagram. A Level DFD 1 is created from a context diagram by sub-dividing the system into a number of processes while maintaining consistency by showing the same data flows into and from the system as the context diagram.

6. (a) To establish precisely what the client needs, and to find out the good/bad points of the current system.
   (b) Any two of: questionnaires; observation; examine the documentation available for the current system.
   (c) System Specification.

Chapter 3: System Design, Development and Testing

1. Any four of:
   - Design of the user interfaces.
   - Specification of data structures.
   - Identification of data validation or verification required.
   - A test plan.
   - Design of any reports to be produced by the system.

2. (a) Storyboarding is useful to assist discussions with clients about the proposed user interface. The following steps are carried out:
   - Diagrams are produced showing the contents of each page/screen of the user interface.
   - All inputs and outputs are identified.
   - Details of navigation through the system are shown.
   (b) Advantages, two of:
   - Storyboards can be created quickly.
   - They promote early communication between the designer and client.
   - They can be used to obtain feedback quickly.
   - Storyboards do not require specialist knowledge to understand.
   - Changes can be implemented early in the development process.
   Disadvantages, two of:
   - Storyboards do not provide any interactive ability.
   - Only limited detail is provided.
   - Aspects such as speed, system responsiveness or usability cannot be accurately conveyed.

3. (a) Prototyping is the creation of early versions of software applications so that the client can get a "look and feel" of the system. In terms of the user interface:
   - Prototyping shows the arrangement of menus and colour schemes so that prospective clients can give

feedback.
- The prototype can be used to check with the client that the proposed system meets the user requirements.

In terms of the data model:
- Prototyping can be used to focus on input data required and what outputs the system needs to produce.
- It can help identify data tables and relationships.

(b) Advantages, two of:
- Prototyping can improve the quality of the user requirements and specifications provided to developers.
- Problems, refinements and enhancements can be identified earlier in the development of the system.
- Requires user involvement, so feedback is provided.
- Reduces possibility of misunderstandings between client and developer.
- Users are involved in the user interface which results in an enhanced product.

Disadvantages, two of:
- Prototyping can be expensive.
- Can lead to insufficient reflection on possible extensions or enhancements.
- Users can be swayed by the aesthetics of the prototype and lose sight of more crucial elements such as accuracy and security.
- Seeing a prototype can lead to unrealistic expectations of how quickly the finished system can be implemented.

4. (a) Testing is important to ensure user requirements are met and the system is free from errors.
    (b) (i) Unit Testing
        - Performed in-house by the developer.
        - Tests that individual components of the system work under all circumstances.
    (ii) Integration Testing
        - Performed in-house by the developer.
        - Bringing together individual modules that have already been tested and making sure they work together as one unit.
    (iii) System Testing
        - Performed in-house by the developer.
        - The software is tested against the user requirements.
        - Test plans are followed using valid, invalid and extreme data.
    (iv) Alpha Testing
        - Performed in-house by the developer.
        - It includes unit testing, black box testing, white box testing, system testing and integration testing.
        - Final testing before the software is handed over to the organisation.
        - It uses either simulated or actual data.
    (v) Beta Testing
        - Beta testing comes after alpha testing.
        - Can be considered a form of external user acceptance testing.
        - Different versions of the software, known as pre-release versions, are often released to potential users to test in a realistic environment.
    (vi) Acceptance Testing
        - Performed when the software is ready to be released or handed over to the client.
        - Intended to give the end users the confidence that the software meets their requirements.
        - A group of potential end users tests the application in a "live" environment using real world volumes of data.
        - The users provide feedback on any problems.
        - Eventually, the users sign off the software.

5. (a) Components of a test plan:
    - The part of the system to be tested.
    - What is being tested (for example, valid data, boundary data).
    - The test data to be used.
    - The expected outcome.
    (b) (i) 0 or 1
        (ii) Any integer in the range 2-17
        (iii) Any integer in the range 18 – 64
        (iv) Any integer in the range 64 – 100 (reasonable upper limit)
        (v) 0 or 1

(vi) 2 or 17

(vii) 18 or 64

(viii) 65

(ix) Any real number or text, for example 34.5, "ABC"

6. The systems analyst could provide test data – particularly to test the boundaries. They could validate test results against the specification.

Chapter 4: System Implementation

1. **Parallel running**:
   - The old and new systems run side by side until the organisation is happy with the new system.
   - Each transaction is performed twice – once for the old system and once for the new system.
   - The two systems are compared to make sure the new system is producing the correct result.
   - Once the new system has been found to work reliably, the organisation can stop using the old system.

   Benefit, one of:
   - Very safe method of changeover as the old system is still available if the new system fails
   - Changeover can take place over a period of time allowing staff to familiarise themselves with the new system.

   Drawback, one of:
   - It is expensive as both systems have to be resourced
   - Time required to check the two systems against each other
   - Two sets of data must be maintained.

   Direct changeover:
   - The old system is switched off on a particular date and the new system starts
   - No parallel systems

   Benefit, one of:
   - No duplication of resources
   - Everyone moves to the new system at the same time.

   Drawback, one of:
   - Very risky. If the new system fails, it may be very difficult to go back to the old system
   - There may be a period between the old system being switched off and the new system being operational during which no system is available.

   Pilot changeover:
   - The new system is tested in one department or section of the organisation
   - Once users are convinced that the system is fully operational, it can be rolled out to the rest of the organisation

   Benefit, one of:
   - Only one section of the organisation is affected – the rest continues working on the old system until the new system is proven to work
   - Users from the pilot section can help to train new users
   - Any problems identified in the new system only affect one section of the organisation.

   Drawback, one of:
   - Extra resources are required as two systems are running concurrently in different sections of the organisation
   - It may be difficult to share data from the new system with sections running the old system.

   Phased changeover:
   - The new system is gradually introduced by activating different components, one at a time.

   Benefit, one of:
   - Staff are introduced to changes in stages
   - If a problem occurs, it should only affect one section of the organisation.

   Drawback, one of:
   - Changeover can take a long time
   - One section of an organisation may ask for changes to the system after it has been introduced, which may slow down implementation in other departments.

2. (a) Pilot – one outlet is tested with the new menu before rolling it out to other restaurants.

   (b) Direct – the bookshop moves everything from one location to another at one time.

   (c) Parallel – both manual and electronic systems are running in parallel.

   (d) Pilot – one episode of the new format is tried out before deciding to change all future episodes.

(e) Parallel – everything is available on both computers until the person is happy that everything works on the new computer, before getting rid of the old laptop.

(f) Phased – the layout of a different store is changed at regular intervals (each week), so that eventually all stores will have the same layout.

(g) Direct – the new system replaces the old system immediately.

(h) Parallel – both systems are running concurrently until the bank is satisfied that the new system is working.

(i) Phased – the decoration of one room is completed before moving on to decorate the next room.

3. It is necessary to produce documentation at each stage:
   - to provide a starting point for the next stage of the development process,
   - to provide a record of work done at each stage,
   - to aid maintenance,
   - to track changes of personnel in the development team.

4. (a) User documentation, three from:
   - It contains an overview of/introduction to system.
   - The hardware and software installation instructions.
   - A user guide.
   - Troubleshooting section.
   - FAQ section.
   - Training materials.
   - Backup procedures/maintenance procedures.

   (b) Technical Documentation, three from:
   - System/user requirements.
   - Data modes, data flow diagrams (DFDs), entity-relationship diagrams (ER diagrams).
   - Input/Output design.
   - Report specifications.
   - Query designs.
   - Code listings.
   - Test plans with data used and results.
   - Detailed hardware and software configuration.

5. (a) Data conversion is the task of converting data under the old system into the correct format for the new system. Jane's florist business currently holds all records in paper-based format (diary, invoice book and receipt book). In an electronic system the data will be stored digitally, so the paper-based format will need to be converted into tables to hold customer details, order details, cost of raw materials etc.

   (b) Phased changeover would allow Jane to gradually move over to an electronic system, perhaps by first creating a database table for customer details, then creating the remaining tables over a period of time.

6. (a) Corrective – correcting errors or weaknesses in the system when it is in use
   (b) Adaptive – a change that is forced by some internal or external influence
   (c) Perfective – a change aimed at improving the operation or performance of the system after initial delivery.

7. (a) Adaptive – the current system works at present but needs to change to deal with changes brought about by an internal influence.
   (b) Corrective – the system does not work as it cannot deal with account numbers of 6 digits, so this must be corrected.
   (c) Perfective – the current system works but the bank is trying to improve the time it takes to process a cheque.
   (d) Perfective – the current system works but the bank is trying to improve the user interface.
   (e) Corrective – the system cannot deal with transfers greater than £65,535. This must be corrected.

Chapter 5: Alternative Development Approaches

1. (a) • Consists of a sequence of distinct stages – Analysis, Design, Implementation, Testing and Maintenance.
   - Each stage must be completed before the next stage begins.
   - Deliverables are produced at the end of each stage.
   - An earlier stage may need to be revisited if an error is found at a later stage of development.

   (b) Analysis:
   - During this stage the systems analyst establishes the user requirements using different methods of fact-finding (such as interviews, questionnaires, observation and document sampling).
   - The end product of the analysis stage is a set of well-documented, clearly defined and unambiguous user

requirements, known as the system specification.

Design:
- Starts with the system specification from which the inputs, processes and outputs of the system are identified.
- The software units, interaction between them, and the identification of hardware needs are included.
- The user interfaces are designed (often using storyboarding)
- The end product of this stage is a design document.

Implementation:
- The software developers write the code, often involving a team of programmers, each of whom is assigned particular tasks.

Testing:
- The system is tested to ensure that it meets the system specification using a test plan developed as part of the system specification.

Maintenance:
- This may involve the migration of existing data from the old to the new system.
- Improvements or refinements may be required immediately following installation.
- Maintenance also involves changes to the system over an extended period of time which may be a result of internal (changing business requirements) or external (legislation) influences.

(c) The user is only involved at the analysis stage when the systems analyst is gathering the user requirements and at the final stages of testing to ensure the system is working in the way in which it is intended.

(d) Advantages:
- There are distinct stages with clearly-defined deliverables at the end of each stage.
- Each stage is likely to require staff with different skillsets, so staff are experts in one particular area and work on the same stage of a range of different projects.

Disadvantages:
- Having staff focus on a particular stage means that they are limited in their skillset.
- As the waterfall model has a critical handover interface between stages, any misunderstanding introduced may travel through the system for quite some distance before being detected.

2. (a) The Rapid Application Development model:
- Produces applications by dividing the project into a series of builds.
- Each build undergoes its own development.
- Requirements are ranked in priority order, with "desirable add-ons" sacrificed if time runs out.
- Has four phases: requirements, user design, construction and cutover phases.

(b) Requirements:
- During this phase the systems analyst establishes the user requirements using different methods of fact-finding (such as interviews, questionnaires, observation and document sampling).

User Design:
- This phase involves the regular interaction between users and systems analysts.
- Prototypes are used to show the user what the system might look like – useful for testing out a variety of alternative user interfaces.

Construction:
- In this phase the software developers write the code.
- The software is continually refined, adapted and enhanced in response to further user input.
- Testing and integration of various software modules takes place in this phase.

Cutover:
- This is the transition from the old system to the new system.
- Further testing is necessary to make sure the new system is working as intended.

(c) The user is involved throughout development, with prototypes being produced frequently and delivered to the users for feedback.

(d) Advantages:
- RAD can shorten development time, so the new system can be built, delivered and operational much sooner.
- Continual user involvement means that misunderstanding and/or errors can be identified at an early stage.

Disadvantages:
- RAD can usually only be undertaken by skilled practitioners trained in the methodology.
- Often used when time is of the essence and can therefore be expensive.

3. (a) In the Agile methodology:
- The project is undertaken in a series of iterations called sprints.

- A product backlog is produced at the start of development.
- As features on the product backlog are selected for implementation, these are moved to the release backlog.
- The release backlog is used to provide early deliverables.
- The Scrum master co-ordinates members of the Scrum team to develop the project.

(b) The user is continually consulted throughout development.

(c) Advantages, any two of the following:
- The project can start quickly, with important requirements being given highest priority, and then evolve.
- Deliverables are created at an early stage.
- Ongoing communication with the customer means it is less likely that there will be misunderstandings.
- It is flexible because the team members are multi-skilled and can be moved around in order to support work that is taking longer than estimated.

Disadvantages, any two of the following:
- It can prove difficult to accurately estimate the time needed to complete activities and subsequent allocation of the team members to undertake the work.
- The accuracy of the time estimates are often dependent on the skill and experience of the Scrum master.
- Documentation may be of poorer quality or less detailed when compared to systems developed using more traditional techniques.
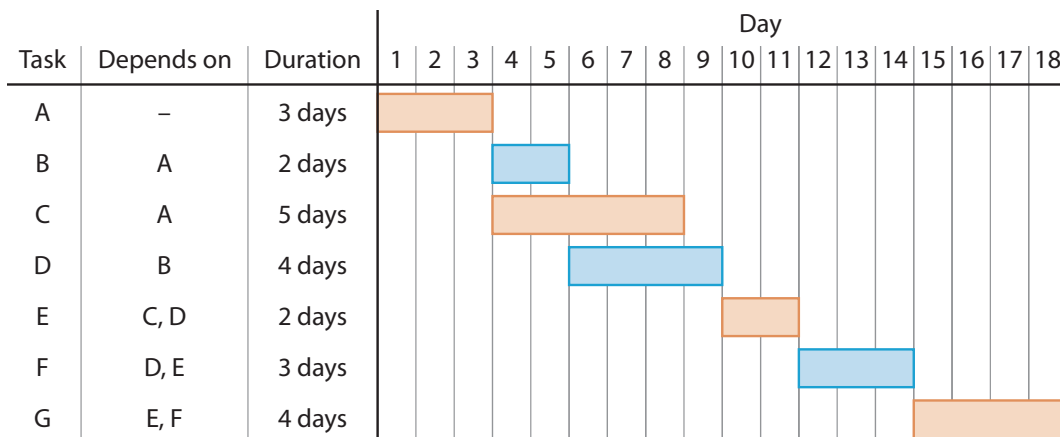
Chapter 6: Software Projects

1. (a) Resources – A resource can be human (for example, programmer, graphic designer), technical (for example, processing power, network bandwidth), financial (for example, available budget) or physical (for example, available accommodation).

   (b) Constraints – A restriction which is placed on the project, for example:
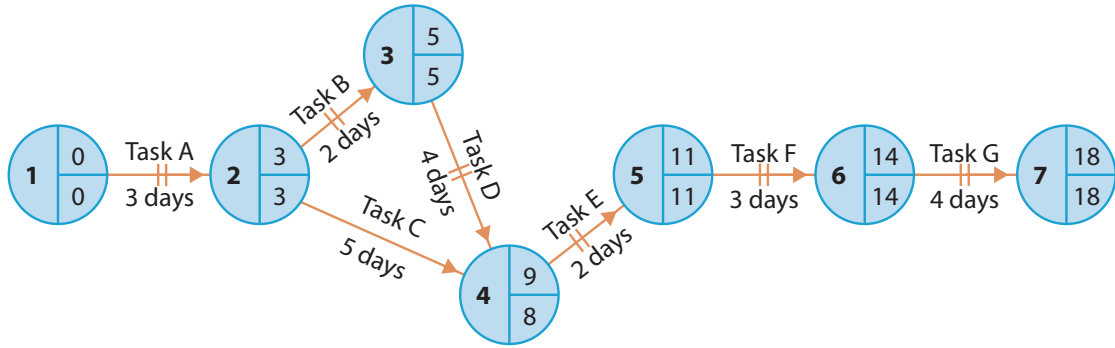   - Scope – the characteristics of the project may be limited by the client.
   - Time – there is usually a fixed deadline by which the final system must be delivered.
   - Resources – there is generally a limit to the resources available for a project, such as cost.

   (c) Risks – Anything which could have a detrimental effect on the project or cause it to fail, for example,
   - Time – scheduling problems resulting in the project not being completed by the agreed date.
   - Requirements shift – what the developer has produced is not what the client requires, often caused by the client not being involved enough in the analysis stage (incomplete user requirements).
   - Staff turnover – changes in personnel in the development team.
   - Productivity – the project manager needs to guard against the risk of complacency and keep the project on track from the start.

2.
   - Specific – clearly defined objectives with no possibility for misunderstanding.
   - Measurable – each objective should have a clear, quantifiable metric that can be used to assess success, for example, time, money, size.
   - Agreed – the objectives should be signed off by both the client and developer before development begins.
   - Realistic – the project should have objectives that the team believes they can meet with the resources available.
   - Timed – each objective should have a target date by which it will be achieved.

3. (a)

Task	Depends on	Duration	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
A	–	3 days	███	███	███															
B	A	2 days				███	███													
C	A	5 days				███	███	███	███	███										
D	B	4 days						███	███	███	███									
E	C, D	2 days										███	███							
F	D, E	3 days												███	███	███				
G	E, F	4 days															███	███	███	███

(b)

4. (a) A Gantt chart is a form of bar chart which tracks tasks against time and allows the manager to keep track of tasks, milestones and interdependencies. The manager can easily see:
   - what each task is,
   - when each task is scheduled to begin and end,
   - how long each task is scheduled to last,
   - the degree of overlap between tasks,
   - dependencies between tasks.

   Critical path analysis allows a software project manager to analyse all the project tasks so that delays can be minimised and any potential conflicts resolved. A project is represented by a circle and arrow diagram where:
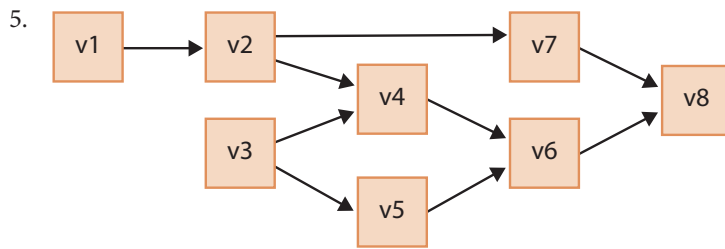   - circles (nodes) represent the events within the project,
   - arrows represent the tasks,
   - the earliest start time (EST) and latest finish time (LFT) for each event is shown.

   (b) Advantages of Gantt charts and critical path analysis:
   - They help the project manager to identify which events can be completed in parallel, thus minimising the overall project duration.
   - They help with resource planning and allocation.
   - CPA can help identify bottlenecks and resolve conflicts.

   (c) Version management allows a team of developers to keep track of all changes made to their program code files over time. Version management only works successfully if:
   - there is a consistent method of tracking changes, recording all version numbers,
   - it is possible to reverse changes,
   - they help with resource planning and allocation,
   - all members of the development team work from the same version of the code file when making changes.

5.

Chapter 7: Security Issues

1. A restore point is a record of what a system looked like at the moment in time it was backed up.

2. **Full backup**: A copy is made of all data on the system.
   Advantages:
   - Everything is backed up at the same time.
   - The complete system can be restored from the one backup.
   Disadvantages:
   - It can take a long time to create the backup and a long time to restore the complete system.
   - In most systems, only small amounts of data change from one day to the next, which leads to a lot of unnecessary work and wasted storage.

   Differential backup: A copy is made of all data on the system that has changed since the last full backup.

Advantages:
- Only files which have changed since the last full backup are copied, so it is faster to create than a full backup.
- Faster to restore than incremental backups as there are at most two copies of each file.

Disadvantages:
- It takes longer to restore than a full backup as there are two possible locations for each file – the full backup or the most recent differential backup.
- Requires more storage space than a full backup as there may be two versions of a file.

Incremental backup: A copy is made of all data on the system that has changes since the last backup (full or incremental).

Advantages:
- It only copies files which have changed since the last backup of any type, so it is the fastest backup method.
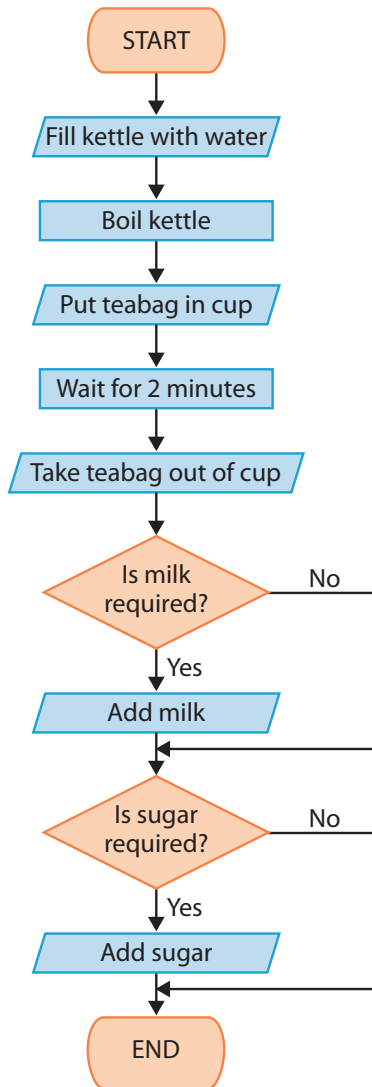- Requires the least amount of storage space.

Disadvantages:
- It takes longer to restore than any other type of backup as multiple copies of each file may have been copied since the last full backup.
- Problems can occur if any of the incremental backups have been corrupted as every incremental backup created since the last full backup is required to restore the system.
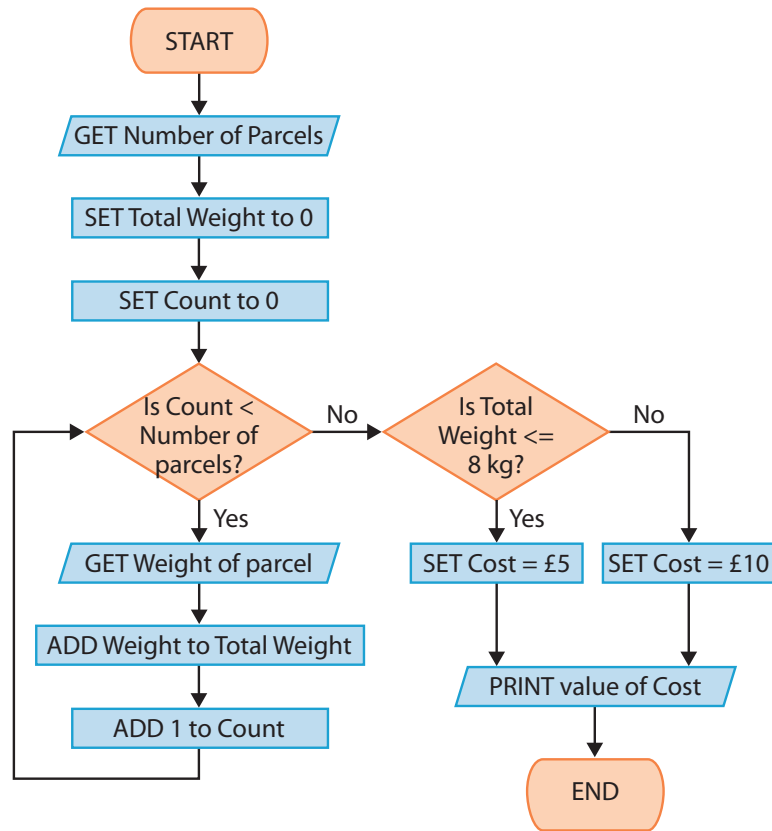
3. (a) Data archiving is the process of moving inactive data to safe and secure storage. The supermarket must be able to retrieve the data at some point in the future for reference or regulatory compliance.
   (b) Advantages of data archiving, two of:
   - Speeds up everyday processing as fewer records need to be accessed to generate reports.
   - Speeds up the backup process as fewer records are on the system.
   - Allows the supermarket to keep data of inactive customers for future reference or regulatory compliance.
   - Records of inactive customers will be indexed to allow easy location and retrieval.
   - Inactive records will be stored on cheaper storage devices, thus saving the supermarket money.
   (c) Disadvantages of data archiving:
   - All data is not immediately available. If a customer starts using their loyalty card after five years, time must be spent searching the archived data for their record.
   - Data access times for archived data can be slow.
   (d) Data recovery:
   - A full backup takes place every Sunday at 6:00am where all data is copied to another location.
   - An incremental backup takes place each day at 10:00pm where all data which has changed since the last backup (full or incremental) is copied to another location.
   - When the hardware problem occurs at 3:20pm on Wednesday the data is recovered as follows:
     - The data from the previous Sunday's full backup is restored first.
     - Data from each of the incremental backups made on Sunday, Monday and Tuesday are restored in this order.
   - After this data is restored, the system will be back to where it was at 10:00pm on Tuesday evening. Each transaction that has taken place from 10:00pm on Tuesday is usually held in a separate transaction file and these can be applied to the system to bring it up to date.

4. (a) Disaster recovery is the way in which an organisation recovers from a catastrophic event – accidental (for example, fires), natural (for example, earthquakes) or deliberate (for example, malicious hacking).
   (b) A Disaster Recovery Plan should identify:
   - critical data,
   - key personnel,
   - backup and recovery process,
   - an alternative site from which the organisation can operate.
   (c) The recovery point objectives (RPO) is a list of the parts of the system that will need to be restored
      The recovery time objectives (RTO) is the time limit for achieving the recovery of each part of the system.
   (d) Any two of the following techniques:
   - Selection of an appropriate backup scheme, which might include storing regular backups off-site.
   - Use of private online storage for instant restoration of data in the event of a local disaster.
   - Full online deployment with servers and databases hosted by a third party service provider.

Chapter 8 – Algorithms

1. START
       Fill kettle with water
       Boil kettle
       Put teabag in cup and wait for 2 minutes
       Take teabag out of cup
       If person wants milk, add milk
       If person wants sugar, add sugar
   END

```
                        START
                          |
                          v
                Fill kettle with water
                          |
                          v
                     Boil kettle
                          |
                          v
                  Put teabag in cup
                          |
                          v
                  Wait for 2 minutes
                          |
                          v
                Take teabag out of cup
                          |
                          v
                    Is milk          No
                    required?  ------------
                          | Yes          |
                          v              |
                     Add milk            |
                          |<-------------
                          v
                    Is sugar         No
                    required?  ------------
                          | Yes          |
                          v              |
                     Add sugar           |
                          |<-------------
                          v
                        END
```

2. START
       GET Number of Parcels
       SET Total Weight to 0
       SET Count to 0
       REPEAT Number of Parcels times
           GET Weight of parcel
           ADD Weight to Total Weight
           ADD 1 to Count
       END REPEAT
       IF Total Weight ≤ 8 kg, Cost = £5, otherwise Cost = £10
       PRINT value of Cost
   END

Chapter 9 – Fundamentals of Programming

1. Any three from:
    Editor – to write, edit and save code. Most editors provide additional facilities:
    - Smart editors – using colour, indentation, code outlining (where regions of code can be collapsed/expanded), line numbering.
    - Syntax error identification – use of wavy lines to indicate incorrect code.
    - IntelliSense – suggestions of available variables, functions etc. appear in a popup window.
    Preferences – ability to set your own code layout:
    - A team of programmers can decide how to lay out the code so that when each team member's contribution is brought together it is laid out consistently as agreed.
    Compiling – ability to convert the code into a form that the computer can understand.
    - IDE should also allow the programmer to run the program once it has been successfully compiled.
    Debugging – ability to correct errors in a program:
    - Provide the ability to add breakpoints in the execution of the program to allow the programmer to inspect the contents of variables at those points.
    - Ability to step through the code, one line at a time.

2. (a) Translation is the process of converting a computer program from one computer language to another. Programs are written in a high-level language designed for humans to understand (for example, C#) and need to be translated to a low-level language (such as assembly language) so that the computer can execute the code.
    (b) **Assemblers**:
    - Low-level programming language.
    - There is a one-to-one mapping between the programmer instructions and the machine-code instructions.
    - The instructions are closely related to the hardware being programmed.

Compilers:
- Translates a program written in a high-level programming language into a common intermediate language or bytecode which is processed during execution.
- Translates the entire program before executing it.
- An error report is produced at the end of compilation of the complete program.
- An executable version of the program is produced which can be run without reference to the original program.

Interpreters:
- Operates one instruction at a time, translating the instruction and then executing it before translating the next.
- Errors can be debugged as they are detected immediately.
- No object code version is produced.

3. (a) A variable is a named location in the computer's memory capable of holding a single value while a program is running. It will normally be assigned a specific data type. Each variable is given a name (identifier) so that the programmer can refer to it.

   (b) (i) age (integer)
       (ii) address (string)
       (iii) price (decimal)
       (iv) license (boolean)
       (v) initial (character)

4. (a) START
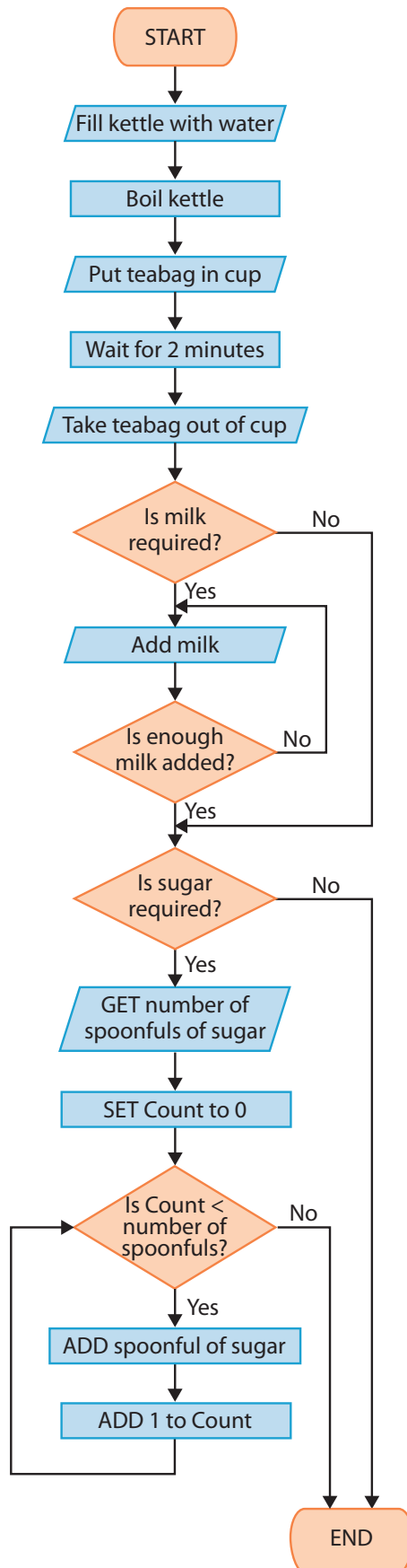   Fill kettle with water
           Boil kettle
           Put teabag in cup and wait for 2 minutes
           Take teabag out of cup
           IF person wants milk
               REPEAT
                   ADD milk
               UNTIL required amount of milk is added
           ENDIF
           IF person wants sugar
               GET number of spoonfuls
               REPEAT Number of Spoonfuls times
                   ADD spoonful of sugar
               END REPEAT
           ENDIF
       END
       NOTE: Adding milk is a condition-controlled loop whereas adding sugar is a count-controlled loop.

(b)

START

Fill kettle with water

Boil kettle

Put teabag in cup

Wait for 2 minutes

Take teabag out of cup

Is milk required? — No

Yes

Add milk

Is enough milk added? — No

Yes

Is sugar required? — No

Yes

GET number of spoonfuls of sugar

SET Count to 0

Is Count < number of spoonfuls? — No

Yes

ADD spoonful of sugar

ADD 1 to Count

END

5. `age >= 18 && age <= 30`

6.
```
if (size < 12)
{
    Console.WriteLine("\nComponent size too short");
}
else
{
   if (size > 12.5)
   {
        Console.WriteLine("\nComponent size too long");
   }//if
}//if
```

7. (a) START
    READ and STORE person's Name
    SET Total Hours to 0
    REPEAT 5 times
        READ and STORE Overtime Hours
        ADD Overtime Hours to Total Hours
    END REPEAT
    PRINT Name and Total Hours
    END

(b)
```
static void Main(string[] args)
{
    string name;
    int hours, totalHours = 0;

    Console.Write("\nPlease enter your name: ");
    name = Console.ReadLine();

    //Use a for loop to read in and add up overtime hours for each day
    for (int count = 1; count <= 5; count++)
    {
       Console.Write("\nHow many overtime hours for day " + count + ": ");
       hours = Convert.ToInt32(Console.ReadLine());
       totalHours = totalHours + hours;
    }//for

    Console.WriteLine("\n" + name  + ", you have worked " + totalHours +
       " overtime hours this week");
    Console.ReadKey();
}//Main
```

8. (a)
```
static void Main(string[] args)
{
    int noOfParcels, cost;
    double weight, totalWeight = 0;

    Console.Write("\nHow many parcels have you? ");
    noOfParcels = Convert.ToInt16(Console.ReadLine());
    for (int count = 0; count < noOfParcels; count++)
    {
        Console.Write("\nPlease enter the weight of parcel " + (count + 1) + " (in kg): ");
        weight = Convert.ToDouble(Console.ReadLine());
        totalWeight = totalWeight + weight;
    }//for

    if (totalWeight <= 8)
    {
        cost = 5;
    }//if
    else
    {
        cost = 10;
    }//else
    Console.Write("\nCost of postage:\t\t£");
    Console.WriteLine("{0:0.00}", cost);
    Console.ReadKey();
}//Main
```

(b) START
    REPEAT
    GET Number of Parcels
    IF (Number of Parcels < 1 OR Number of Parcels > 10)
       PRINT warning message
    ENDIF
    WHILE (Number of Parcels < 1 OR Number of Parcels > 10)
    SET Total Weight to 0
    REPEAT Number of Parcels times
       REPEAT
         GET Weight of parcel
         IF (Weight < 1 OR Weight > 5)
           PRINT warning message
         ENDIF
       WHILE (Weight < 1 OR Weight > 5)
       ADD Weight to Total Weight
    END REPEAT
    IF Total Weight <= 8 kg
       SET Cost = £5
    OTHERWISE
       IF Total Weight < 20 kg
         SET Cost = £10
       OTHERWISE
         SET Cost = £20
       ENDIF
    ENDIF
    PRINT value of Number of Parcels
    PRINT value of Total Weight
    PRINT value of Cost
  END

(c) Count-controlled loops are used when we know in advance the number of times an action needs to be performed. Example:

REPEAT Number of Parcels times

    …

    ADD Weight to Total Weight

END REPEAT

This for…loop is an example of a count-controlled loop as the Number of Parcels is entered before the loop starts.

Condition-controlled loops are used when an action is performed while a specified condition is true. Example:

REPEAT

    GET Number of Parcels

    IF (Number of Parcels < 1 OR Number of Parcels > 10)

        PRINT warning message

    ENDIF

WHILE (Number of Parcels < 1 OR Number of Parcels > 10)

This do… while loop is an example of a condition-controlled loop as it repeats until a valid number of parcels is entered.

Or, this example:

REPEAT

    GET Weight of parcel

    IF (Weight < 1 OR Weight > 5)

        PRINT warning message

    ENDIF

WHILE (Weight < 1 OR Weight > 5)

This do… while loop is an example of a condition-controlled loop as it repeats until a valid weight of parcel is entered.

(d)
```csharp
static void Main(string[] args)
{
    int noOfParcels, cost;
    double weight, totalWeight = 0;

    do
    {
        Console.Write("\nHow many parcels have you? ");
        noOfParcels = Convert.ToInt16(Console.ReadLine());
        if (noOfParcels < 1 || noOfParcels > 10)
        {
            Console.WriteLine("\nYou must enter a value between 1 and 10");
        }//if
    } while (noOfParcels < 1 || noOfParcels > 10);

    for (int count = 0; count < noOfParcels; count++)
    {
        do
        {
            Console.Write("\nPlease enter the weight of parcel " + (count+1) +
                " (in kg): ");
            weight = Convert.ToDouble(Console.ReadLine());
            if (weight < 1 || weight > 5)
            {
                Console.WriteLine("You must enter a value between 1 and 5");
            }//if
        } while (weight < 1 || weight > 5);
        totalWeight = totalWeight + weight;
    }//for
```

```
   if (totalWeight <= 8)
   {
      cost = 5;
   }
   else
   {
      if (totalWeight < 20)
      {
         cost = 10;
      }
      else
      {
         cost = 20;
      }//if
   }//if

   Console.WriteLine("\nNumber of parcels:\t\t" + noOfParcels);
   Console.Write("Total weight of parcels:\t");
   Console.Write("{0:0.00}", totalWeight);
   Console.WriteLine("kg");
   Console.Write("Cost of postage:\t\t£");
   Console.WriteLine("{0:0.00}", cost);

   Console.ReadKey();
}//Main
```
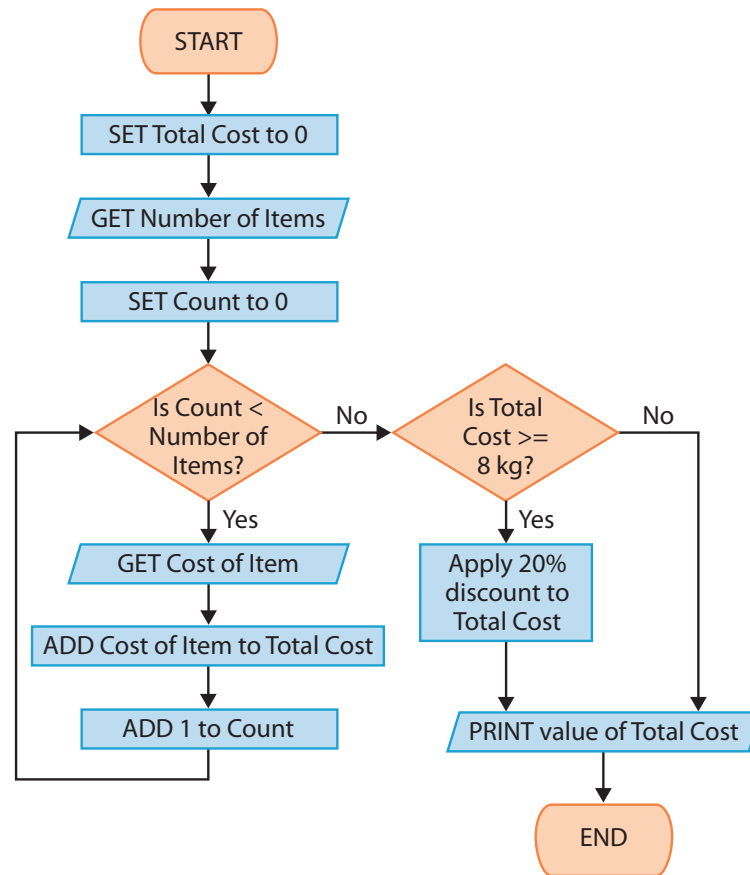
9. (a) (i) Sequence is the process of carrying out a set of instructions in the order in which they are listed. In the example given, the following steps will be executed one after the other:
- Set total cost to zero.
- Cost of each item will be entered.
- Discount will be applied (if appropriate).
- Total cost of items will be printed.

(ii) Selection is when one of two sets of statements is executed depending on a particular condition being true or false. In the example given, the discount of 20% will only be applied if the total cost of all items is greater than 500.

(iii) Iteration is the repetition of the same sequence of instructions until or while some condition is met. There are two types of iteration:
- Count-controlled loops, when we know in advance the number of times an action needs to be performed.
- Condition-controlled loops, when an action is performed while a specified condition is true.

In the above exampl,e a count-controlled loop is used to read in and store the cost of each item and then add the cost to the total.

(b)

(c)
```
static void Main(string[] args)
{
    // Declare a variable for the Total Cost of the order and set to 0
    double totalCost = 0;
    // Declare variable for Number of Items
    int noOfItems;
    // Declare variable for the Cost of each Item
    double itemCost;
    // Declare variable to store the discount
    double discount;
    // Enter the Number of Items in the order
    Console.Write("\nHow many items are there? ");
    noOfItems = Convert.ToInt32(Console.ReadLine());
    // Enter the cost of each item
    for (int count = 1; count <= noOfItems; count++)
    {
        Console.Write("\nPrice of item " + count + ": ");
        itemCost = Convert.ToDouble(Console.ReadLine());
        totalCost = totalCost + itemCost;
    }//for
    // Calculate and apply discount, if applicable
    if (totalCost >= 500)
    {
        discount = (totalCost * 0.2);
        totalCost = totalCost - discount;
    }//if
    //Output total price
    Console.WriteLine(String.Format("\nTotal Cost:    {0:C}", totalCost));
    Console.ReadKey();
}//Main
```

10. (a) **Syntax error**: The syntax of a programming language is the set of rules which define the format of each statement in the language. A syntax error is any error which does not conform to these rules, for example a missing semicolon.

    Runtime error: This occurs while the program is running. A common runtime error is entering the wrong type of data, for example entering a decimal number when an integer is expected.

    Logical error: A logical error occurs when the code compiles and produces an answer, but the output is incorrect because there is some flaw in its logic. For example, a program reads in two numbers (num1 and num2) and outputs their product, but the programming instruction to calculate the product is (num1 + num2) rather than (num1 × num2).

(b) (i) The variable netPay is used but has not been declared in line 3.

    The semicolon is missing at the end of line 5.

    The opening bracket { is missing after the first 'if' statement.

    (ii) In line 8, the condition:
```
if (grossPay == 10000)
```
should read:
```
if (grossPay > 10000)
```

(c) (i) Replace the section 'Calculate tax and net pay' with the following:

```
// Calculate tax and net pay
if (grossPay > 40000)
{
   tax = ((grossPay - 40000) * 0.4) + 6000;
}//if
else
{
   if (grossPay > 10000)
   {
      tax = (grossPay - 10000) * 0.2;
   }//if
   else
   {
      tax = 0;
   }//else
}//else
netPay = grossPay - tax;
```

Note: The number `6000` in the code above could be replaced with: `(30000 * 0.2)`.

(ii)

Test	What part of system being tested	What is being tested	Data to be used for the test	Expected Outcome	Actual Outcome	PASS/ FAIL
5	Initial Data Input	Valid data – high taxable rate	Any value above £40,000, for example 60000	Gross Pay £60000 Tax: £14000 Net Pay: £46000		
6	Initial Data Input	Boundary data – between low and high taxable rates	40000	Gross Pay £40000 Tax: £6000 Net Pay: £34000		

Chapter 10 – Object-Oriented Programming

1. (a) (i)   Class: A class is the basic building block of object-oriented programming. It specifies the attributes (properties) together with the behaviours (methods) which can be performed on those attributes. In the above code the class is 'Square'.

   (ii)   Object: An object is a member (instance) of a class. In the above code 'square1' and 'square2' are objects. They are members of the 'Square' class.

   (iii)   Attribute: An attribute is a characteristic (property) of an object: 'description' and 'lengthOfSide' are attributes of each object of the 'Square' class.

   (iv)   Behaviour: The behaviour of an object is defined by its methods, i.e. the actions that can be performed on an object. 'setDescription', 'setLengthOfSide', 'getDescription', 'getLengthOfSide' and 'calculateArea' are all examples of behaviours of each object of the 'Square' class.

   (b)   Output:

   ```
   The area of Small Square is 9 units squared
   The area of Large Square is 625 units squared
   ```

2. Class: A class is the basic building block of object-oriented programming. It specifies the attributes (properties) together with the bahaviours (methods) which can be performed on those attributes.
   Object: An object is a member (instance) of a class.
   An example of a class could be Car with car1, car2, car3 being three different objects.

3. Encapsulation is where the technical details of attributes and behaviours are hidden within the object so that data can only be changed if an appropriate method has been provided. To enforce this, the attributes of an object are defined as private. A method is defined to inspect the current value of a specific attribute of an object (a 'get' method) and a method is defined to amend the value of the attribute (a 'set' method).

4. Inheritance is the mechanism whereby the attributes and behaviours of one class can be inherited by a new, more specialised, class derived from it. The original, general, class is referred to as the superclass (or base class). New subclasses (or derived classes) based on this one, but more specialised, can be created.

5. Advantages of the object-oriented approach to programming include the following:
   - More accurate modelling – the object-oriented approach models the real world more realistically than traditional methods which tended to focus on data and processing.
   - Improved productivity due to code reusability – when a new object is created, it will automatically inherit the attributes and behaviours of the superclass from which it is derived.
   - Improved productivity due to less testing required – only the new methods developed for the derived class need to be tested as those of the superclass will already have been tested.

   Disadvantages of the object-oriented approach to programming include the following:
   - It is not suitable for all applications. Some programs are very abstract and do not readily correspond to real world 'objects'.
   - It has been widely adopted in computer-aided design (CAD), computer-aided manufacturing (CAM) and engineering design systems. However, it has not been as successful for either corporate systems or many 'run of the mill' information systems applications, such as payroll and accounting.

6. (a) and (b)

```
class Student
{
    private string studentNumber;
    private string surname;
    private string firstName;
    private string course;
    private int yearOfCourse;

    //Constructor to create a student
    public Student(string newStudentNumber, string newSurname, string newFirstName,
        string newCourse, int newYear)
    {
        this.studentNumber = newStudentNumber;
        this.surname = newSurname;
        this.firstName = newFirstName;
        this.course = newCourse;
        this.yearOfCourse = newYear;
    }
```

```csharp
    public string getStudentNumber()
    {
        return this.studentNumber;
    }

    public string getSurname()
    {
        return this.surname;
    }

    public string getFirstName()
    {
        return this.firstName;
    }

    public string getCourse()
    {
        return this.course;
    }

    public int getYearOfCourse()
    {
        return this.yearOfCourse;
    }
    public void setSurname(string newSurname)
    {
        this.surname = newSurname;
    }

    public void setFirstName(string newFirstName)
    {
        this.firstName = newFirstName;
    }

    public void setCourse(string newCourse)
    {
        this.course = newCourse;
    }

    public void setYearOfCOurse(int newYear)
    {
        this.yearOfCourse = newYear;
    }

    public void printAllDetails()
    {
        Console.WriteLine("\nStudent Details");
        Console.WriteLine("***************");
        Console.WriteLine("Student Number:      " + this.studentNumber);
        Console.WriteLine("Name:          " + this.firstName + " " + this.surname);
        Console.WriteLine("Course Enrolled on:  " + this.course);
        Console.WriteLine("Year of Course:      " + this.yearOfCourse);
    }
}//Student
```

(c)
```
static void Main(string[] args)
{
    Student student1 = new Student("MUR384733", "Murphy", "James", "A Levels", 2);
    Student student2 = new Student("JON300940", "Jones", "Rachel", "Level 3 IT", 1);
    Student student3 = new Student("ROB544494", "Robinson", "Hannah", "A Levels", 2);

    //Print out the attributes of each house
    student1.printAllDetails();
    student2.printAllDetails();
    student3.printAllDetails();

    Console.ReadKey();
}//Main
```

(d) Output from this code is as follows:

```
Student Details
***************
Student Number:    MUR384733
Name:              James Murphy
Course Enrolled on: A Levels
Year of Course:    2

Student Details
***************
Student Number:    JON300940
Name:              Rachel Jones
Course Enrolled on: Level 3 IT
Year of Course:    1

Student Details
***************
Student Number:    ROB544494
Name:              Hannah Robinson
Course Enrolled on: A Levels
Year of Course:    2
```

Chapter 11 – Data Representation

1. (a) bit – a binary digit (stores either a 0 or a 1)
    (b) byte – a group of 8 bits, for example 10100011
    (c) kilobyte (KB) – 2^{10} bytes
    (d) megabyte (MB) – 2^{20} bytes
    (e) gigabyte (GB) – 2^{30} bytes
    (f) terabyte (TB) – 2^{40} bytes

2. 16 777 216 numbers need to be represented = 2^{24}. Therefore 24 bits are required.

3. (a) $(1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$
        $= (1 \times 8) + (0 \times 4) + (1 \times 2) + (1 \times 1)$
        $= 8 + 0 + 2 + 1$
        $= 11$
    (b) $(1 \times 2^5) + (0 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$
        $= (1 \times 32) + (0 \times 16) + (0 \times 8) + (1 \times 4) + (1 \times 2) + (1 \times 1)$
        $= 32 + 0 + 0 + 4 + 2 + 1$
        $= 39$
    (c) $(1 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (0 \times 2^0)$
        $= (1 \times 128) + (0 \times 64) + (0 \times 32) + (0 \times 16) + (1 \times 8) + (0 \times 4) + (0 \times 2) + (0 \times 1)$
        $= 128 + 0 + 0 + 0 + 8 + 0 + 0 + 0$
        $= 136$

4. (a) $45 \div 2 = 22$ remainder 1
$22 \div 2 = 11$ remainder 0
$11 \div 2 = 5$ remainder 1
$5 \div 2 = 2$ remainder 1
$2 \div 2 = 1$ remainder 0
$1 \div 2 = 0$ remainder 1
Reversing the remainders (i.e. reading from bottom to top) gives 101101.
Check:
$(1 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0)$
$= (1 \times 32) + (0 \times 16) + (1 \times 8) + (1 \times 4) + (0 \times 2) + (1 \times 1)$
$= 32 + 0 + 8 + 4 + 0 + 1$
$= 45$

(b) $99 \div 2 = 49$ remainder 1
$49 \div 2 = 24$ remainder 1
$24 \div 2 = 12$ remainder 0
$12 \div 2 = 6$ remainder 0
$6 \div 2 = 3$ remainder 0
$3 \div 2 = 1$ remainder 1
$1 \div 2 = 0$ remainder 1
Reversing the remainders (i.e. reading from bottom to top) gives 1100011.
Check:
$(1 \times 2^6) + (1 \times 2^5) + (0 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$
$= (1 \times 64) + (1 \times 32) + (0 \times 16) + (0 \times 8) + (0 \times 4) + (1 \times 2) + (1 \times 1)$
$= 64 + 32 + 0 + 0 + 0 + 1 + 1$
$= 99$

(c) $120 \div 2 = 60$ remainder 0
$60 \div 2 = 30$ remainder 0
$30 \div 2 = 15$ remainder 0
$15 \div 2 = 7$ remainder 1
$7 \div 2 = 3$ remainder 1
$3 \div 2 = 1$ remainder 1
$1 \div 2 = 0$ remainder 1
Reversing the remainders (i.e. reading from bottom to top) gives 1111000.
Check:
$(1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (0 \times 2^0)$
$= (1 \times 64) + (1 \times 32) + (1 \times 16) + (1 \times 8) + (0 \times 4) + (0 \times 2) + (0 \times 1)$
$= 64 + 32 + 16 + 8 + 0 + 0 + 0$
$= 120$

5. (a) Add leading zeros: 01011010
Invert: 10100101
Add 1:   10100110

(b) Add leading zeros: 00100111
Invert: 11011000
Add 1:   11011001

(c) Add leading zeros: 10001000
Invert: 01110111
Add 1:   01111000

6. (a) Convert 30 to binary: 11110
Add leading zeros: 00011110
Invert: 11100001
Add 1: 11100010_2
Check:
$(-1 \times 2^7) + (1 \times 2^6) + (1 \times 2^5) + (0 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$
$= (-1 \times 128) + (1 \times 64) + (1 \times 32) + (1 \times 16) + (1 \times 8) + (0 \times 4) + (0 \times 2) + (0 \times 1)$
$= -128 + 64 + 32 + 0 + 0 + 0 + 2 + 0$
$= -30$

(b) Convert 88 to binary: 1011000

Add leading zeros: 01011000

Invert: 10100111

Add 1: 10101000_2

Check:

$(-1 \times 2^7) + (0 \times 2^6) + (1 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (0 \times 2^0)$

$= (-1 \times 128) + (0 \times 64) + (1 \times 32) + (0 \times 16) + (1 \times 8) + (0 \times 4) + (0 \times 2) + (0 \times 1)$

$= -128 + 0 + 32 + 0 + 8 + 0 + 0 + 0$

$= -88$

(c) Convert 121 to binary: 1011000

Add leading zeros: 01111001

Invert: 10000110

Add 1: 10000111_2

Check:

$(-1 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (0 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$

$= (-1 \times 128) + (0 \times 64) + (0 \times 32) + (0 \times 16) + (0 \times 8) + (1 \times 4) + (1 \times 2) + (1 \times 1)$

$= -128 + 0 + 0 + 0 + 0 + 4 + 2 + 1$

$= -121$

7. (a) $(0 \times 2^7) + (0 \times 2^6) + (1 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$

$= (0 \times 128) + (0 \times 64) + (1 \times 32) + (0 \times 16) + (1 \times 8) + (0 \times 4) + (1 \times 2) + (0 \times 1)$

$= 0 + 0 + 32 + 0 + 8 + 0 + 2 + 0$

$= 42$

(b) $(-1 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (0 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$

$= (-1 \times 128) + (0 \times 64) + (0 \times 32) + (0 \times 16) + (0 \times 8) + (0 \times 4) + (1 \times 2) + (0 \times 1)$

$= -128 + 0 + 0 + 0 + 0 + 0 + 2 + 0$

$= -126$

(c) $(-1 \times 2^7) + (1 \times 2^6) + (1 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$

$= (-1 \times 128) + (1 \times 64) + (1 \times 32) + (0 \times 16) + (1 \times 8) + (0 \times 4) + (1 \times 2) + (0 \times 1)$

$= -128 + 64 + 32 + 0 + 8 + 0 + 2 + 0$

$= -22$

8. Unicode, because ASCII can only represent 256 characters which would be insufficient to represent the 2000 different Chinese characters.

Chapter 12 – Data and Information

1. Data is the raw figures, for example 12.6. Information is the processed data, for example 12.6 seconds to complete 100 m. Knowledge is the application of a rule to the information, for example that Sam completed the 100 m in the fastest time (12.3 seconds).

2. Data is the raw figures, for example 32, 40. Information is the processed data, for example 32 kg. Knowledge is the application of a rule to the information, for example the sale of apples and bananas did not meet the target in this week while that of pears did.

3. Any three from: accurate, relevant, up-to-date, complete, well-presented, reliable.

4. (a) Data which is not required, for example height, weight.
   (b) Insufficient data, for example only the first name is recorded, but there could be more than one person with the same first name.
   (c) Data which changes over time, for example age.

5. Data validation is a check to ensure that the data entered is sensible and reasonable within acceptable boundaries. Data verification ensures that all copies of the data exactly match the original source.

6. (a) 6017, calculated as follows:

Weighting	4	3	2
Digit	6	0	1
Weighting × Digit	24	0	2

Checksum = 24 + 0 + 2 = 26   Remainder = 26 ÷ 11 = 4   Check digit = 11 – 4 = 7

(b) 95850, calculated as follows:

Weighting	5	4	3	2
Digit	9	5	8	5
Weighting × Digit	45	20	24	10

Checksum = 45 + 20 + 24 + 10 = 99   Remainder = 99 ÷ 11 = 0   Check digit = 0

(c) 415065, calculated as follows:

Weighting	6	5	4	3	2
Digit	4	1	5	0	6
Weighting × Digit	24	5	20	0	12

Checksum = 24 + 5 + 20 + 0 + 12 = 61   Remainder = 61 ÷ 11 = 6   Check digit = 11 – 6 = 5

(d) 801X, calculated as follows:

Weighting	4	3	2
Digit	8	0	1
Weighting × Digit	32	0	2

Checksum = 32 + 0 + 2 = 34   Remainder = 34 ÷ 11 = 1   Check digit = 11 – 1 = 10 (X)

7. (a) Double entry is one method of checking that the data matches the original source. When a computer system asks for a new password the user is often asked to type it twice. The computer will check if the same string of characters has been typed each time and notify ther user if there is a difference. Entering the data twice should identify the majority of mistakes but it effectively doubles the workload and therefore can be very costly for large amounts of data.

   (b) Proofreading is a method of verification that can be used where both copies of the data are available. The original data is visually checked against the data entered into the computer, either on screen or from a printout. This can be time consuming and expensive if there is a lot of data, and mistakes can be overlooked if the proofreading is performed too quickly. Ideally the proofreading should be done by a different person from the one who typed in the data.

8. (a) Two range checks – that the number before the : is in the range >= 0 AND <= 23, and that the number after the : is in the range >= 0 AND <= 59.

   (b) Type check – integer rather than text.

   (c) Length check – exactly 6 characters.

   (d) Format check – exactly 2 letters followed by 4 digits.

   (e) Presence check .

9. Batch, control and hash totals are validation checks which are used as a control precaution to ensure data from each document in a batch is entered exactly once.

   Before any data is entered the user counts how many documents (in this case scorecards) there are. This is the **batch total**. As the data from each document is typed in, the computer maintains a count of how many documents data has been entered for. After all the data has been entered the computer generated total is compared to the manual total. If there is a discrepancy then either data from one document has been missed or some data has been entered twice. A batch total would not give an error if the data from one document was missed out but data from another document was entered twice as these two errors would cancel each other out.

   In this example, to calculate the batch total, one would need to manually count the number of scorecards and also have a cell in the spreadsheet to calculate the number of total scores entered. In this case there are 20 scorecards, so the number of total scorecards entered into the computer should also be 20.
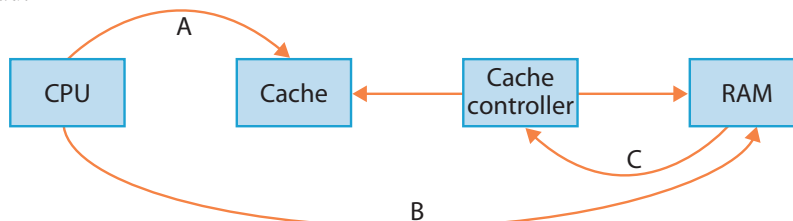
   Control or **hash totals** are types of batch totals which try to spot this type of error. A batch total which has some meaning is known simply as a control total, whereas a batch total that has no meaning is known as a hash total. In this example, the competitor IDs could be used as a hash total, as the total sum has no meaning. All 20 competitor IDs could be added together (manually and electronically). If a card has been omitted, the difference in the manual and electronic total would help to identify the ID of the competitor whose scorecard data has been missed.

   In this example, the total score obtained by each competitor could be used as a control total. All 20 competitor **total** scores could be added together (both manually and electronically). If a card has been omitted, or one card omitted and another entered twice, the difference between the manual and electronic totals would indicate that there was an error or inaccuracy in the data entry.

10. (a) Information is processed data, for example 03/09/15 represents a date in DD/MM/YY format. Knowledge is the application of a rule to the information, for example profit is £2.00 (£5.99 – £3.99).
    (b) Relevance – making sure the content is linked to the topic, for example that the product name contains the appropriate information. Completeness – all information is present, for example that the selling price is included.
    (c) Data validation ensures the data is sensible, reasonable, in the correct format, within set boundaries etc.
    (d) Any two data validation methods from:
        - Range check, for example 1 <= month <= 12.
        - Length check – tha the plant identifier is 6 characters long.
        - Format check, for example the date format of the date of purchase.
        - Lookup list could be used for product name.
        - Presence check – that something is entered into every field.
    (e) Data verification:
        - Manual verification (proof reading) where the source is read and checked against the data stored looking for differences.
        - Computer verification (double entry) where data is entered twice and differences are highlighted by the computer.

Chapter 13 – Architecture

1. (a) Control Unit:
        - It extracts instructions from memory and decodes and executes them, calling on the ALU when necessary (the fetch-execute cycle).
        - It controls and monitors the hardware attached to the system to make sure that any commands sent to it by the application software are used.
        - It controls the input and output of data.
    (b) Arithmetic Logic Unit:
        - Loads data from the input registers, receives instructions from the control unit on what operation to perform and stores the result in an output register .
        - Performs two types of operation:
            - arithmetic computations such as addition, subtraction, multiplication and division ,
            - logical operations which deal with logic and comparisons using <, >, = etc.
    (c) Registers:
        - Temporary, high-speed storage areas which are used for data or instructions needed quickly by the control unit.
        - Each register is a single storage location within the CPU.
        - All data must be stored in a register before it can be processed.

2. Read-only memory (ROM):
    - Used to store fixed programs such as basic instructions to start up the computer and launch the operating system.
    - The content of ROM is permanent – it cannot be altered by the user.
    - Memory is non-volatile, so the content of ROM is preserved when the computer is turned off.
    Random access memory (RAM):
    - Used to temporarily store data and programs that are currently in use by the processor.
    - Memory is volatile, so the contents are lost when the computer is switched off.

3. Cache memory performs two tasks:
    - to store frequently used instructions, and
    - to 'read ahead'.

A device called a cache controller transfers the data from RAM into the cache. If the CPU needs to obtain information from memory, it goes through the following process:
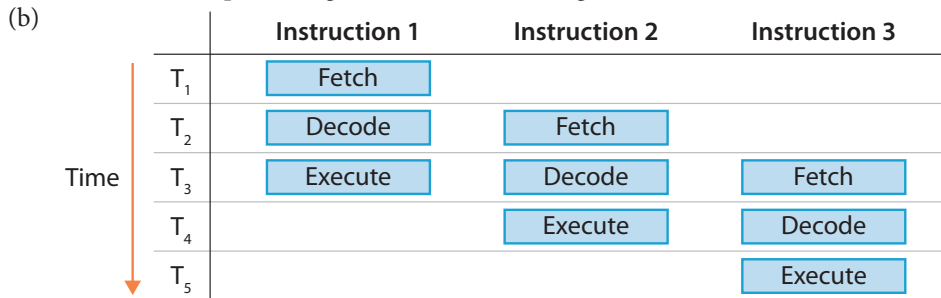1. Checks whether the information is located inside the cache (A).
2. If the cache controller has been successful in predicting that the CPU would need this data, then the required

information is available from a previous data reading.

3. If not, the CPU will get the information from RAM (B) and place it in the cache (C) before processing it. The cache controller may then read further data in response to this event.

4. (a) An external device, for example a printer.
    (b) The three parts are:
    - I/O port which allows the exchange of data between the CPU and the peripheral.
    - Electronics which interface the controller to the system bus.
    - Electronics capable of sending signals to the device connected to the computer.
    (c) External devices differ in speed, operate at different voltages and have different levels of complexity.

5. (a) Clock speed does not consider the architecture of the processor.
    (b) RAM (bits) = $2^{(\text{address bus width})}$ × (data bus width) bits

    $64 \times 2^{30} \times 8 = 2^{(\text{address bus width})} \times 128$ bits

    $2^6 \times 2^{30} \times 2^3 = 2^{(\text{address bus width})} \times 2^7$ bits

    $2^{39} \div 2^7 = 2^{(\text{address bus width})}$

    $2^{32} = 2^{(\text{address bus width})}$

    Address bus width = 32 bits
    (c) The width of the address bus determines the number of addresses the processor can talk to, i.e. how much main memory it can directly access.
    (d) The drive is a CD-ROM, therefore read only. It is not possible to write data using a CD-ROM drive.

6. (a) The address bus carries the memory address. The data bus carries data from a memory location to the processor. The control bus sends signals such as read/write.
    (b) RAM (bits) = $2^{(\text{address bus width})}$ × (data bus width) bits

    $= 2^{32} \times 64$ bits

    $= 2^{32} \times 2^6$ bits

    $= 2^{38}$ bits

    $= 2^{35}$ bytes

    $= 2^5$ Gigabtyes

    $= 32$ GB
    (c) Cache memory is faster and more expensive (per megabyte) than main memory. It is located on (or immediately adjacent to) the processor.
    (d) Reasons:
    - RAM is expensive.
    - Most programs do not require maximum RAM to be installed.
    - Some addresses are allocated to input/output ports.
    (e) The maximum addressable memory size would be doubled from 32 GB to 64 GB.
    (f) Using clock speed or benchmark tests giving arithmetic capability in MIPS (million instructions per seconds) or FLOPS (floating point operations per second).

7. 2. Data bus is set up with data to be written.
    4. Data from the data bus is placed into the specified memory location.

8. RAM (bits) = $2^{32} \times$ (data bus width) bits

    $64 \times 2^{30} \times 8 = 2^{32} \times$ (data bus width) bits

    $2^6 \times 2^{30} \times 2^3 = 2^{32} \times$ (data bus width) bits

    $2^{39} \div 2^{32} =$ (data bus width) bits

    $2^7 =$ (data bus width) bits

    Data bus width = $2^7 = 128$ bits

9. | | | |
    |---|---|---|
    | 1 – clock | 2 – processor | 3 – main memory |
    | 4 – mouse controller | 5 – screen controller | 6 – keyboard controller |
    | 7 – mouse | 8 – screen | 9 – keyboard |
    | 10 – control bus | 11 – data bus | 12 – address bus |

10. (a) **Fetch**: This will fetch the next instruction to be processed by placing the instruction's memory address on the address bus and retrieving the memory contents from the data bus. The precise sequence of events is as follows:
    1. The address of the first instruction to be read is loaded into the program counter (PC).
    2. The contents of the PC are placed in the memory address register (MAR).
    3. The PC is updated to point to the location of the next instruction which must be executed.
    4. The MAR places the address on the address bus.

5. A 'read' signal from the MAR loads the instruction into the memory data register (MDR) via the data bus.

6. The instruction is copied into the current instruction register (CIR).

Decode: The second step is to decode the instruction that has been placed in the CIR. The CPU is designed to understand a specific set of commands (known as the 'instruction set'). Different makes of CPU have different instruction sets. The control unit decodes the instruction by splitting it into an operation ('opcodes') and data ('operands'). It also prepares various registers for the execution of the command.

Execute: This third step is where the actual data processing takes place. If the instruction involves some arithmetic or logic, the ALU will be used. The opcode identifies the type of the instruction (add, subtract etc). The result of the processing is stored in another register – the accumulator (ACC).

(b)

	Instruction 1	Instruction 2	Instruction 3
T_1	Fetch		
T_2	Decode	Fetch	
T_3	Execute	Decode	Fetch
T_4		Execute	Decode
T_5			Execute

Time →

11. (a) Benefits of secondary storage:
- capacity – it has a greater storage capacity than RAM;
- portability – data can be saved and moved from one place to another;
- security – important data can be copied and stored safely (backups).

(b) Three from:
- speed – how quickly the user needs to be able to access the data from the storage device;
- capacity – the amount of data to be stored;
- durability – how long the storage media is expected to last;
- portability – how far the data has to be moved (e.g. to another continent or just an adjacent room);
- reliability – assurance that the data can always be accessed reliably and in the format in which it was saved.

(c)
- Access speed – internal storage devices are connected to the data bus which allows much faster access than data on an external device connected through a hardware interface such as a USB port.
- Portability – external storage devices are very portable so data can be moved easily from one location to another.

(d) When a memory stick is inserted into a USB socket the appropriate USB driver is loaded into memory to provide the computer with code on how to read and write from the USB device. Different USB devices have different drivers. Data is read from a memory stick as follows:
1. The USB device is read, giving information on the file and folder structure (the file allocation table, or FAT) to the computer.
2. When the user attempts to read a file on the device, the computer sends the address required to the USB port.
3. The USB device returns the data from the required the location.

(e) The steps involved to write data to the hard disk are:
1. The platters spin around the spindle.
2. Data is sent to the hard disk using the disk controller.
3. The actuator arm moves the write head to the track that will be written to.
4. Once the data sector that is required has spun around and is under the write head, data is written to the platter.

12. (a) Two from:
- requires less power;
- no moving parts;
- fast access times.

(b) Advantages:
- It is smaller than a magnetic hard drive, which makes it particularly suitable for tablet computers that have insufficient space for a traditional hard drive.
- It provides faster data access than a traditional hard drive due to the absence of moving parts.
- It requires less power, which is a very important consideration for portable computers.
- It is very reliable: because there are no mechanical parts, there is less chance of something being damaged by an impact.

Chapter 14 – The User Interface and Data Compression

1. Four from:
   - simple, with a clear, concise interface design which is easy to learn and use;
   - intuitive and consistent, allowing the user to navigate through the system easily and apply previously-learned knowledge to new tasks;
   - responsive, so that the user feels that the interface is 'keeping up' with their mouse clicking or input from the keyboard;
   - forgiving, so that when the user makes mistakes the actions are easily reversible;
   - attractive, providing an environment that is aesthetically appealing both in terms of typography and colour scheme. Visual appeal enhances the user experience.

2. In each case, two from the list.
   (a) Advantages of a WIMP interface:
      - It is intuitive to use.
      - It is easy to navigate, as the user does not have to learn complicated commands.
      - It is easy to exchange data between different software applications.
      - Help guides are available for novices, and shortcut keys for experts.

      Disadvantages of a WIMP interface:
      - It requires a lot of memory (RAM).
      - It uses much more hard disk space than other types of user interface.
      - A powerful processor is required.
      - It takes more time for programmers to develop applications for a GUI.

   (b) Advantages of a command line interface:
      - Little processing power and CPU processing time is required, compared to other user interfaces.
      - It uses much less memory (RAM) than other user interfaces.
      - A cheap, low resolution monitor can be used.
      - If the user knows the correct commands, then this type of interface can be faster than any other type of user interface.

      Disadvantages of a command line interface:
      - Commands need to be typed precisely. An error will result in the command not being executed correctly and, in the worst case, could result in the loss of a lot of data.
      - Can be very confusing for someone who has never used a command line interface.
      - Not very intuitive, since there are a large number of commands to learn.
      - It is unsuitable for novice users.

   (c) Advantages of a forms interface:
      - It is easy to use; there is no need to learn a lot of commands.
      - It is ideal for beginners as everything is displayed in a logical manner.
      - Little processing power or memory is required.
      - Data validation can be performed on the data as it is being entered.

      Disadvantages of a Forms interface:
      - Poorly designed forms may be slow to use.
      - Only a limited number of options can be made available.
      - Can be tedious for experts.

3. An advanced programmer would be likely to know the correct commands to perform operations. A command line interface might be easier and faster for them to use than a graphical user interface.

4. - To reduce file size to save storage space.
   - To reduce transmission (upload and/or download) time.

5. (a) Compression makes the file size smaller.
   (b) • to reduce file size to save storage space;
      • to reduce transmission (upload and/or download) time.
   (c) Lossless compression algorithms do not lose any quality.

6. (a) 2^{24} = 16 777 216 colours
   (b) Size $= (3 \times 150) \times (4 \times 150) \times 3$ bytes
              $= 810\ 000$ bytes
              $= 810$ KB (approximately)
   (c) Increasing the bit depth would increase the size of the file.

7. Each picture = $(4 \times 1024) \times (6 \times 1024) \times 24$ bits
   = 603 979 776 bits
   = 75 497 472 bytes
   = 72 MB

   12 GB = 12×2^{30} bytes
   = 12 884 901 888 bytes
   = 12×2^{10} MB
   = 12 288 MB

   12 884 901 888 bytes ÷ 75 497 472 bytes = 170.6 pictures (maximum) can be stored.

   or

   12 288 MB ÷ 72 MB = 170.6 pictures (maximum) can be stored.

Chapter 15 – Software

1. An operating system is a system software program that manages all the operations and resources of a computer. It acts as the interface between the hardware, application software and the end users.

2. A single user operating system:
   - Only one person has access to the computer at any given time.
   - There are two types:
     - Single user, single application – only one program can be run at any given time.
     - Single user, multitasking application – several applications (processes) can be in operation at any given time. The user can switch between processes.
   A multi-user operating system:
   - Enables multiple users to run programs simultaneously (from a few users up to thousands of users).
   - Each user has their own console and appears to have exclusive use of the system.
   - The OS must make sure that the resources (such as memory, processing and data requests) allocated to the users are managed, so that each user has sufficient resources.
   - Time slicing can be used to share time between tasks.
   - Data security must be maintained ensuring each user's files and data are kept private from all other users.

3. (a) Multiprogramming operating system:
   - More than one program (process) can be resident in the computer's main memory.
   - Only one program can be executed at any given time.
   - Optimises CPU time.
   - From the user's perspective, the operating system is switching between processes rapidly enough to create the illusion of parallelism.
   (b) Multiprocessing operating system:
   - More than one program (process) can be resident in the computer's main memory.
   - Each task is divided between multiple processors resulting in parallel computing.
   - The main advantage is speed.
   (c) Multitasking operating system:
   - Each process consists of a number of tasks.
   - Enables the user to perform more than one task at a time.
   - The OS allocates resources accordingly.
   - Time-slicing may be used to allocate CPU time.
   (d) Multithreading operating system:
   - Ability of a process to be used by more than one user at a time, or to manage multiple requests by the same user without running multiple copies of the process.
   - Each user runs a 'thread'.
   - Multiple threads can exist within the same process, executing concurrently and sharing resources.

4. (a) Data compression program:
   - Reduces the size of the files.
   - Reduces storage space
   - Reduces file transfer times.
   (b) File archive program:
   - Combines a number of files together into one archive file.

- File contents are concatenated sequentially into archives with 'metadata' (containing filenames, lengths of files, access rights etc) stored with them.
- Often uses data compression.
- Can be used for backup and recovery.

(c) Disk defragmenting program:
- Increases access speeds by shuffling the data around until each file has contiguous clusters.
- Can take many hours as a lot of data may need to be moved around.

5. (a) Advantages of custom-built software. Two from:
- The organisation will get exactly what they need.
- The software will be optimised for the company's specific needs.
- Only those features required by the system will be provided (and hence storage space will not be taken up by other unnecessary and unused functions).
- The developers will be available to correct any errors which arise during initial use of the system.

Disadvantages of custom-built software. Two from:
- It can be very expensive to develop.
- It can take a long time to develop.
- The software may not be fully tested and there may be bugs in the source code which are only triggered by a specific event, for example the start of a new year.
- Little support will be available online via blogs or forums.

(b) Advantages of off-the-shelf software. Two from:
- The software is available immediately.
- It will usually cost less than custom-built software.
- The software will be fully tested.
- A lot of support will be available online.

Disadvantages of off-the-shelf software. Two from:
- Installation and configuration may take some time to ensure that it is aligned to the company's needs.
- There may be many features that the company does not require (and extra storage space is required for these features).
- The software may not have some features that the company requires.

6. (a) Features of propriety software:
- It is owned by an individual or a company.
- It is distributed under a license agreement, i.e. it is copyright protected.
- Has specific features and capabilities.
- Source code is generally not available to the public.

(b) Advantages of propriety software. Two from:
- Tend to be easier to use and provide greater functionality than open-source products.
- Usually includes regular upgrades and technical support.
- Product is constantly updated to detect and remove bugs.
- Developers can incorporate the latest industry standards into the programs.

(c) Disadvantages of propriety software. Two from:
- Users depend on the developers for support and upgrades.
- User may be paying for features they do not require.
- User cannot modify or customise the software.
- If the developing company goes out of business, there will no longer be support for the product, so the user may have to change products (extra cost).

(d) Features of open source software:
- Software for which the copyright to source code is in the public domain.
- No licence is required to use the code.
- Users can use, change or improve the software and redistribute it, modified or unmodified.

(e) Advantages of open source software. Two from:
- It is free and no licence is required.
- The source code is available.
- No restrictions on the use of the software. Users can use, change or improve the software and redistribute it modified or unmodified.
- Not dependent on one developer, as anyone can access the source code and continue development even if the originator is no longer involved.

(f) Disadvantages of open-source software. Two from:
- Many versions may be available as the software is developed collaboratively.
- The software may not be widely used.
- The software may not have been fully tested/documented.
- Bugs found in the system may not be fixed.

(g) The shop owner could choose the open source software, as they have considerable programming knowledge and could tailor the software to their specific needs.

7.
- The data to be processed (cheques) is grouped until a suitable processing time is reached (end of each day).
- The entire batch is processed in one go, usually at an 'off peak period'.
- All data undergoes the same processing without human intervention.
- Control languages or scripts can be used.
- Validation checks (batch/control) are performed.

8.
- Each order will be processed immediately.
- The stock level will be updated immediately, before another transaction takes place.
- The system can check if there is sufficient stock to fulfil the order.
- The appropriate amount will be immediately debited from the customer's account.
- If there are insufficient funds in the customer's account, the purchase can be cancelled immediately.

9. Input (any two from):
- Cash card is inserted.
- PIN is entered.
- Menu choice is selected.
- An amount is entered or selected.

Processing (any two from):
- The customer's data is retrieved from the bank's database.
- The PIN is validated.
- The amount is checked against the customer's balance.
- The amount is deducted from the customer's balance.

Output (any two from):
- Instructions/menu choices are displayed on the ATM screen.
- A receipt may be printed.
- A sum of money is issued.

10. (a) Input devices (any two from):
- Barcode reader.
- Touchscreen numeric keypad.
- Electronic scales.
- Card reader for electronic payment.

(b) Output devices (any two from):
- Monitors displaying product prices and running total.
- Printing device to provide a receipt.
- Auditory feedback to indicate failure of the system to recognise a product.

(c) Processing:
- The bar code of an item is read and the database is searched to get the cost of the item.
- Calculating the cost of an item which has to be weighed.
- For an electronic payment:
  - The PIN is validated.
  - The amount is checked against the customer's balance.
  - The amount is deducted from the customer's balance.

11. Input (any two from):
- Touch-sensitive screen.
- Receive data transmitted by wifi and high-speed cellular networks such as 4G and the emerging 5G.
- Embedded microphone for voice commands.
- Fingerprint sensor to secure the device against unauthorised access.
- High-resolution image sensors that enable the user to take photographs and record video.
- A gyroscope (or tilt sensor) allows the phone to automatically switch between portrait and landscape displays according to the orientation of the device in the user's hand.

Output (any two from):
- The screen.
- A near-field communication (NFC) antenna embedded in the smartphone body allows it to be used as an electronic payment device by wirelessly transferring details of the user's payment card across short distances (a few centimetres) to an EFTPOS system.
- Embedded speakers can be used to output audio. Alternatively, users can choose to connect headphones, either through a headphone socket or wirelessly via Bluetooth.
- A 'vibrate' feature allows the device to alert the user to incoming calls or other events even when the device has been switched into 'silent' mode.

Chapter 16 – Web Applications

1. (a) FTP
   (b) HTTP

2. (a) The World Wide Web is a global information system organised as a collection of linked pages, where each page has a unique address.
   (b) A Web server is a program running on a computer connected to the Internet that delivers web pages when requested by a browser. Pages are delivered using the HTTP protocol.
   (c) A website is a set of related pages that are identified by a common domain name and are delivered by a common web server.
   (d) A web page is a single hypertext document connected to the World Wide Web and accessible through a web browser.

3. An Internet Protocol address (IP address) is a numeric value assigned to each device that is guaranteed to refer only to that device. It is either a 32-bit value (IPv4) or a 128-bit value (IPv6). On the Internet, the IP address is usually represented by a domain name, a human-readable combination of words separated by full stop characters which maps onto an IP address.

4. www.ulster.ac.uk is a sub-domain of ulster.ac.uk, which is itself a sub-domain of ac.uk.

5. The stateless nature of HTTP makes it very simple to implement as each request is an independent transaction. It is also a very efficient protocol with minimal network resource required.

6. A Domain Name Server (DNS) provides a translation service between domain names and IP addresses by maintaining a dictionary that maps between them. As IP addresses may change over time, the Internet contains a number of Domain Name Servers which continually exchange information with each other to ensure that the most up-to-date IP address is always associated with the correct domain name.

7. An intranet is a private network that is contained within an organisation and can only be accessed by members of that organisation. The term is most often used to refer to a 'private' Web of online services. Intranets allow individuals to communicate and collaborate, but within a secure area that cannot be accessed by anyone outside the authorised group of users.

8. Examples of features you might expect an ISP to provide are:
   - Internet access.
   - Domain name registration.
   - Web hosting.
   - Email.
   - Email forwarding.
   - Database support.
   - Blog/message board software.
   - Photo album.

9. **Web crawler**: The aim of a Web crawler is to visit every page on the Internet. It starts from a 'seed' page and collects all of the links on that page. All of these links are then visited in turn, collecting all of the links to pages that have yet to be visited and adding these to the list of pages to visit. The process continues until every page has been visited or an upper limit has been reached.
   Index: As the Web crawler visits each page, it gathers information about the page, including the text it contains, any images or video on the page and any descriptive information provided in the page header. All this information is stored in a dictionary structure so that the content is associated with the page on which it is found.
   Query engine: This provides an interface through which the user can search for some specific text. The search engine compares the user's request against the contents of the index and returns the list of pages on which this text

is found. The list of matching pages is sorted according to some criteria so that the 'best' matches are returned at the top of the list.

10. Public key encryption is a technique that uses two key values (essentially long, random numeric sequences) that are connected by a mathematical formula such that data encrypted using one of the keys can only be decrypted by using the other key. The key used for encryption is known as the public key and is made freely available, while the key used for decryption is known as the private key and is available only to the individual to whom it belongs. For example, if user Alice wishes to send a private message to user Bob she encrypts the message using Bob's public key. The message can only be decrypted using Bob's private key, which only Bob has access to. In this way, even if the message is intercepted during transmission, it cannot be decrypted and read.

Chapter 17 – Website Development

1. An empty tag is one that does not consist of an opening and closing pair, such as `<br>` or `<img>`. A container tag is a pair where the closing element contains a / character after the opening `<`. These include the anchor tag `<a>`...`</a>`, the paragraph tag `<p>`...`</p>` and the heading tag `<h1>`...`</h1>`.

2. (a)
```
<h1>Welcome to the Model School</h1>
<p>We prepare students for the following subjects:</p>
<ul>
    <li>Technology</li>
    <li>Languages</li>
    <li>Sciences</li>
</ul>
<a href="information.html">Click here to continue to our further information page</a>
```

(b) Change the list container tag from `<ul>`...`</ul>` to `<ol>`...`</ol>`.

(c) Add the line of code:
```
<img src="modelschool.png">
```
before the opening `<h1>`...`</h1>` tag.

3. CSS is used to specify the visual characteristics of the elements on the page, in terms of their colour, font, text style, spacing and positioning. HTML describes the content of the page in terms of the elements used such as paragraphs, headings, hyperlinks and images.

4. Linked style sheets are where the CSS rules are specified in an external file and then linked to the HTML file. This allows a single style sheet to be applied to a collection of HTML files so that a single change in the CSS will be applied across all HTML files to which the CSS is linked. This greatly simplifies the application of a 'house style' across an entire website and the ongoing maintenance of that style. Linked CSS files also allow different styles to be applied to the same content depending on the device used for viewing (e.g. desktop computer, smartphone).

5. (a) Red text in the Arial font. The font is inherited from the CSS rule for the `p` element, while the red colour definition within the `<p>`...`</p>` tag overwrites the earlier CSS definition.

(b) The text would be displayed in blue, as both the `font-family` and `color` style attributes would now be inherited from the `<style>` definition.

6. Client-side processing would be the check that both the username and password had been entered. If either or both of these are missing, the user will be prompted again and no communication will be made with the server. Once the username and password are entered and the button clicked, the values are passed to the server. Now, server-side processing will be used to retrieve the details for the specified account from the database, compose the web page that displays the information and return that web page to the browser.