A Level Computer Science Preparation Tasks

We hope you are looking forward to beginning your A Level Computer Science studies in September. In the meantime, we would like to provide you with an opportunity to prepare for the course.

You will complete a set of tasks which should be submitted via a Google Classroom assignment at the start of the course. This work will help us to assess your effort and current skill level, enabling us to better understand your strengths and areas for development.

You must complete:

- Task A: Programming Challenges (complete at least TWO problems)
- Task B: Ethics Task

Task A – Programming Challenges

The pages that follow contain three programming challenges. You should use a programming language of your choice to develop solutions for each problem. You are required to complete **at least two of the three** programming tasks.

We encourage you to create a GitHub account and upload your work there, sharing a link to your repository as part of your submission. This will provide an effective way to manage and share your code. All program code should include appropriate and detailed comments.

We are not only interested in your programming ability, but also in your problem-solving approach, testing process, and ability to document your development work in detail. These are important skills within A Level Computer Science.

For each programming task you complete, you should produce a report including the following sections:

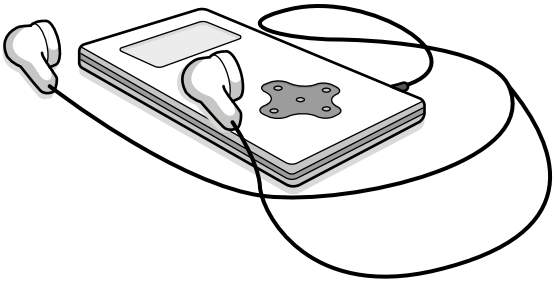
- Analysis – Define the success criteria for your program and explain how you will ensure these are met.
- Development – Include screenshots of your code during development and describe the programming techniques used, explaining how these meet your success criteria.

- Testing – Create a test plan containing at least 8 tests. Carry out the tests, include evidence (screenshots), and clearly state whether each test has passed or failed. Where tests fail, explain the corrective actions taken.
- Evaluation – Review your original success criteria and explain how each has been met. If any criteria have not been fully achieved, explain why. Be honest and reflective about your technical development.

Please use these headings to structure your submission. The final write-up should be completed in either Google Docs or Microsoft Word. A single document containing all work is preferred.

Your teacher will ask you to submit this work to a Google Classroom assignment after your first Computer Science lesson of the year. Please ensure it is ready for submission with delay.

TASK 1



Jay is creating a music quiz game.

The game stores a list of song names and their artist (e.g. the band or solo artist name). The player needs to try and guess the song name.

The game is played as follows:

- A random song name and artist are chosen.
- The artist and the first letter of each word in the song title are displayed.
- The user has two chances to guess the name of the song.
- If the user guesses the answer correctly the first time, they score 3 points. If the user guesses the answer correctly the second time they score 1 point. The game repeats.
- The game ends when a player guesses the song name incorrectly the second time.

Only authorised players are allowed to play the game.

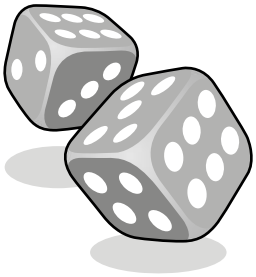
Where appropriate, input from the user should be validated.

Design, develop, test and evaluate a system that:

1. Allows a player to enter their details, which are then authenticated to ensure that they are an authorised player.
2. Stores a list of song names and artists in an external file.
3. Selects a song from the file, displaying the artist and the first letter of each word of the song title.
4. Allows the user up to two chances to guess the name of the song, stopping the game if they guess a song incorrectly on the second chance.
5. If the guess is correct, add the points to the player's score depending on the number of guesses.
6. Displays the number of points the player has when the game ends.
7. Stores the name of the player and their score in an external file.
8. Displays the score and player name of the top 5 winning scores from the external file.

Turn over

TASK 2



Ahmed is developing a two-player dice game.

The players roll two 6-sided dice each and get points depending on what they roll. There are 5 rounds in a game. In each round, each player rolls the two dice.

The rules are:

- The points rolled on each player's dice are added to their score.
- If the total is an even number, an additional 10 points are added to their score.
- If the total is an odd number, 5 points are subtracted from their score.
- If they roll a double, they get to roll one extra die and get the number of points rolled added to their score.
- The score of a player cannot go below 0 at any point.
- The person with the highest score at the end of the 5 rounds wins.
- If both players have the same score at the end of the 5 rounds, they each roll 1 die and whoever gets the highest score wins (this repeats until someone wins).

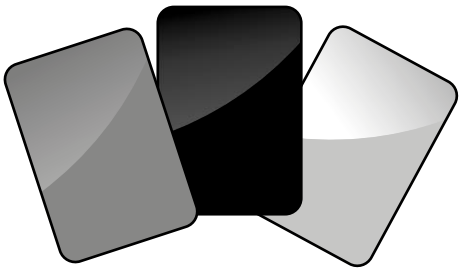
Only authorised players are allowed to play the game.

Where appropriate, input from the user should be validated.

Design, develop, test and evaluate a program that:

1. Allows two players to enter their details, which are then authenticated to ensure that they are authorised players.
2. Allows each player to roll two 6-sided dice.
3. Calculates and outputs the points for each round and each player's total score.
4. Allows the players to play 5 rounds.
5. If both players have the same score after 5 rounds, allows each player to roll 1 die each until someone wins.
6. Outputs who has won at the end of the 5 rounds.
7. Stores the winner's score, and their name, in an external file.
8. Displays the score and player name of the top 5 winning scores from the external file.

TASK 3



Joshua is creating a card game for two players.

The game uses a deck of cards. There are 30 cards in a deck. Each card has one colour (red, black or yellow). Each card has a number (1, 2, 3, 4, 5, 6, 7, 8, 9, 10) for each colour. Each card is unique.

The 30 cards are shuffled and stored in the deck.

The rules are:

- Player 1 takes the top card from the deck.
- Player 2 takes the next card from the deck.
- If both players have a card of the same colour, the player with the highest number wins.
- If both players have cards with different colours, the winning colour is shown in the table.

Card	Card	Winner
Red	Black	Red
Yellow	Red	Yellow
Black	Yellow	Black

- The winner of each round keeps both cards.
- The players keep playing until there are no cards left in the deck.

Only authorised players are allowed to play the game.

Where appropriate, input from the user should be validated.

Design, develop, test and evaluate a program that:

1. Allows two players to enter their details, which are then authenticated, to ensure that they are authorised players.
2. Shuffles the 30 cards in the deck.
3. Allows each player to take a card from the top of the deck. Play continues until there are no cards left in the deck.
4. Calculates the winner and allocates both cards to the winner.
5. Displays which player wins (the player with the most cards).

6. Lists all of the cards held by the winning player.
7. Stores the name and quantity of cards of the winning player in an external file.
8. Displays the name and quantity of cards of the 5 players with the highest quantity of cards from the external file.

Task B – Ethics Task (OCR A Level Preparation)

This task is aligned to the **OCR A Level Computer Science specification (H446 – Section 1.2: Ethical, legal, cultural and environmental impacts of digital technology).**

You should select **one topic area** from the list below:

- Data privacy and data protection
- Artificial intelligence and automation
- Cybersecurity and ethical hacking
- Social media, misinformation and digital wellbeing
- Environmental impact of computer systems

Task Requirements

You must produce a structured written response (recommended word limit: 500–700 words) which:

- Clearly explains the chosen issue within a computing context
- Describes at least two different perspectives or stakeholder viewpoints
- Uses relevant real-world examples
- Identifies ethical, legal, social and environmental (ELSE) implications
- Ends with a justified conclusion that weighs up the arguments presented

Your response should demonstrate your ability to analyse the impact of computing technologies on society and form balanced, reasoned judgements.