Lahore Grammar School
Johar Town Senior Boys Campus

LAHORE GRAMMAR SCHOOL JOHAR TOWN FOR BOYS

INNOVENTIONS '24

INDULGE. INVENT. INNOVATE.

Bit by Bit

(Study Guide)

11th-13th OCTOBER

Round 1: "game_design.docx"

Delegate Cap: 02

Duration: 3h

Teams Eliminated: 30%

Description

This round will challenge delegates to create a comprehensive Game Design Document (GDD) for an original game concept they will create at the start of the round. The theme will not be revealed until the start of the round.

Participants will need to demonstrate creativity, technical knowledge, and effective collaboration skills among each other. The GDD should detail the game's mechanics, story, art style, and overall design with a focus on technical comprehensiveness and feasibility. This includes flowcharts of the game loop, detail on the games mechanics, optional drawings of the characters and descriptions of artstyle (though quality of any drawn art itself will not necessarily be graded, the focus is on the concept and the detailing on its technical implementation).

A sample document for this round can be found on https://linktr.ee/bitbybitguide.

Judging Criteria

- Structure and Presentation (10 marks)
- Game Loop & Mechanics (Fun, Completeness, Usage) (10 marks)
- Originality (5 marks)
- Technical Comprehensiveness (5 marks)

Rules

- **Original Content:** All game concepts must be original and created during the competition.
- **Document Structure:** Teams must ensure all required sections of the GDD are completed.
- **No External Help:** Teams must work independently without external assistance, including but not limited to the use of electronic devices, the internet, and AI Large Language models.

Round 2: "code_blitz.exe"

Delegate Cap: 02

Duration: 2h 30m

Teams Eliminated: 50%

Description

Delegates will be tested on their ability to write the most efficient code to solve given problems. The goal is to solve the most number of problems within the shortest time. This round will challenge participants to optimize their coding skills under time constraints, and their performance will contribute to the second-round leaderboard as well.

Procedure

Teams are provided with a set of programming problems. Problemset will have 8-12 problems with increasing order of difficulty.

Calculation of Scores:

- Each problem will have a specific number of points. Harder problems (having a higher contest rating) will have greater points. If the solution passes all the test cases, the team will be awarded those points. The goal is to maximize the points achieved within the time limit.

Types of Problems:

- The round will have problems in increasing order of difficulty, starting from easier to harder. The contest will consist of problems inspired by Leetcode and Codeforces problemsets.
- For 5-6 easier problems (LeetCode Easy and Codeforces 800-900 Rated, a fundamental understanding of algorithms and data structures is expected. Contestants should be familiar with certain types of problems, including:
  - Implementation
  - Greedy Approach
  - Brute Force
  - Math
- For the rest of 5-6 problems (LeetCode Easy/Mediums and Codeforces 1000 - 1200), contestants will require a higher degree of problem solving skills and problems might consist of previously mentioned problems solving paradigms as well as:
  - Data Structures (Arrays, stacks, and queues)
  - Two Pointers
  - Strings
  - Bitmasks
  - Number Theory

Sample Problems: A list of problems for this round can be found on https://linktr.ee/bitbybitguide. It has a curated list of problems from codeforces.com and instructions on how to participate.
Additional Resources: Practicing codeforces problemsets (800 - 1200 rating) and Leetcode (Easy and Medium Level) can be helpful for preparing for this round.

Rules

1. **IDEs:** Contestants will only be allowed to use online IDEs, Visual Studio Code, and offline text editors / IDEs (including Sublime text, Jupyter, Pycharm, Visual Studio)
2. **Programming Languages:** All standard set of languages that are allowed on Codeforces will be allowed in the contest. These include languages like Python, C++, Java, etc. Pseudo Code would be written on a sheet of paper and the time of submission will be considered as the time the participant raised their hand, or the time they made a submission request through an online form. One of the event managers will receive the paper, and check the code against a key consisting of pseudocode for the optimal solution. 9618 Pseudocode guide and rules will be followed during evaluation. If the code does not satisfy the model answer, the verdict of Wrong Answer or Syntax Error will be delivered and the paper will be returned. Since evaluation of pseudocode will take a longer time than virtual judges, slight leniency of spelling mistakes and declarations will be allowed in order to compensate for such disadvantage.
3. **Cheating:** Contestants will be surveilled and will not be allowed to use LLMs or resources on the internet during the competition. A specific set of documentations for programming languages will only be allowed.

Round 3: "capture_the_flag.txt"

Delegate Cap: 02

Duration: 3h

Teams Eliminated: Standings Determined; All Other Teams Eliminated

Description

This round combines elements of traditional Cyber Security Capture The Flags with problem-solving tasks to create an engaging challenge, even if the delegates don't specifically have programming or cybersecurity experience. Delegates will tackle a series of puzzles and mini-challenges, collecting virtual "flags" for each completed task. After each flag has been collected, these flags will have to be validated by approaching the “Validator” who will keep count of the number of flags collected by each team. These tasks will test their knowledge in various subjects, teamwork, and especially critical thinking. Success in this round will significantly impact the final team standings.

Process

Teams are given a set of challenges spread across different categories, such as logic puzzles, simpler coding problems to extract flags, and deciphering encoded messages. Some problems only unlock once a certain number of flags have been collected. There will be three secret problems that can be discovered for bonus flags. Each challenge completed successfully awards the team a virtual flag. The challenges will be of varying difficulty, allowing teams to strategize and prioritize their efforts. The teams with the most flags at the end of the round gain the most marks, and these marks are tallied with the scores from previous rounds to decide the winning teams. During the duration of the round, delegates would be allowed to access the internet.

Example

One of the challenges might be to solve a coding problem where delegates must realize they have to write a function to reverse a string to uncover a flag (the string cannot be manually reversed as the Validator will ensure the code functions). Another challenge could involve decrypting a substitution cipher or scanning a code to find a hidden message leading to a flag. Teams might also face logic puzzles like building a digital logic circuit to acquire a flag or a series of hidden secrets that lead to a secret flag.