Bioinformatics - Code of Gene-iuses

Event Objective

The objective of this competition is to engage participants in a dynamic, head-to-head bioinformatics challenge that combines coding, biological knowledge, and strategic reasoning.

Participants will go through a quiz round which will test their biological, mathematical, computer science and statistical knowledge.

Qualifying teams will progress to a head-on challenges round, where the teams will face other teams in various challenges where they can both attack and defend their points to stay in the top teams.

Finally, the top teams will work on various Machine Learning problem statements related to Bioinformatics and will try to get the best models on the table for the judges to judge their work.

The event will:

- Promote Bioinformatics + ML among the participants who are new to this field.
- Help build coding skills and analytical skills.
- Make learning fun and competitive with our full duplex round which allows the participants to attack and defend their points.

Event Flow & Rounds

Round 1: Online Quiz (1 week before fest)

- **Format**: MCQs
- **Topics**: Biology, Mathematics, Statistics, and Computer Science basics.

- **Duration**: 30–45 minutes
- **Platform**: Unstop.
- **Selection**: Top 10–12 teams qualify for Round 2.

Round 2:

- **Task**: Teams compete live on organizer questions and peer-broadcasted questions on a shared platform. A real-time leaderboard ranks teams; top teams at the end advance to Round 3.
- **Deliverables**:
    - Answers submitted on platform (auto-graded where possible).
    - For each team-broadcasted question: problem statement + expected answer (and brief solution/verification notes) uploaded when broadcasting.
    - System logs of submissions and timestamps (used for tie-breaks/appeals).
- **Time**: 2–3 hours live match (+5 min grace for appeals). Broadcast windows open at every 30-minute interval.
- **Judging Criteria**:
    - **Initial Points (S_0)**: 100 points for every team.
    - **Floor**: Scores never drop below 0 (no negative scores).
    - **Organizer Questions**: +10 points per correct solve.
    - **Broadcasted Questions (from other teams)**: +25 points per correct solve (higher reward).
    - **Unsolved Broadcast Penalty**:
        - For each broadcasted question not solved by T = 3:00 (and not authored by your team): −5 points at the end.
        - If solved, the −5 penalty is waived (and +25 is awarded).
    - **Broadcast Limits/Quality**:
        - Up to 6 broadcasts per team (one per 30-min window).

        - Each broadcast must include an answer key/verifier.
        - Invalid or ambiguous broadcasts are voided (no penalties, no points).
    - **Anti-spam Attempts**: Only graded submissions count.
        - (Optional) +1 activity bonus per distinct near-correct attempt on organizer questions (max +5).
    - **Fair-Play Safeguards**: If a broadcast is later ruled invalid/ambiguous, all related penalties and points are rolled back.
- **Advancement**: At T = 2:00, after applying penalties, caps, and tie-breaks, the Top 4–5 teams on the leaderboard qualify for Finals.

Round 3:

- **Task**: Implement Machine Learning on a Bioinformatics problem to get best results.
- **Deliverables**:
    - Code Notebook/Script
    - Trained Model/Predictions File
- **Time**: 12 hours
- **Presentation**: Each team gets 10–15 minutes to present their work to judges.
- **Judging Criteria**:
    - Model Performance (40%)
    - Methodology & Innovation (25%)
    - Reproducibility & Code Quality (15%)
    - Presentation & Communication (20%)

Basic Example Problem Statements For Round 2

1. Create a synthetic DNA sequence dataset with missing base-pairs and noise. Teams must clean and reconstruct the dataset to restore sequence accuracy.

2. Provide adversarial protein sequence data with swapped labels. Teams must detect anomalies and relabel correctly using rules/ML techniques.

Basic Example Problem Statements For Round 3

1. Data Clustering on Gene Expression Data

Problem: High-dimensional gene expression datasets are complex and noisy, making it difficult to identify meaningful groups of genes or patients. Traditional analysis struggles to uncover hidden biological patterns.

Challenge: Perform data clustering on a given gene expression dataset to identify natural clusters among the patients, visualise the results and interpret the biological significance of clusters.

Constraints:

- Dataset size will be moderately high-dimensional (hundreds of genes, multiple patients).

- Teams must use Python (NumPy, Pandas, scikit-learn, matplotlib/seaborn) only – no pre-trained ML models or AutoML tools.

- Maximum system runtime allowed: 5 minutes per algorithm (ensure efficiency).

2. Protein Classification Challenge

Problem: Understanding protein function from sequence and physicochemical properties is a central challenge in bioinformatics. Given a synthetic protein dataset with amino acid sequences, calculated properties, and functional classes, teams must build a machine learning model to classify proteins into their respective functional categories.

Challenge:

- Perform exploratory data analysis (EDA) on protein sequences and their physicochemical features.

- Train at least two machine learning models (e.g., Random Forest, SVM, Logistic Regression, Neural Networks).

- Compare their performance on the test set.
- Use feature importance/analysis to explain which properties drive classification.
- Present a confusion matrix and classification metrics (accuracy, precision, recall, F1-score).

Constraints:

- Teams must use Python with libraries like scikit-learn, pandas, matplotlib, seaborn.
- No deep pretrained protein models (e.g., AlphaFold, ProtBERT) – the focus is on classical ML pipelines.
- Maximum model training runtime: 5 minutes per algorithm.
- At least two distinct ML approaches must be implemented.
- Each team must justify feature selection (not just feed everything blindly).

3. Cancer Prediction with Gene Expression

Problem: Early cancer detection is one of the most crucial applications of bioinformatics and machine learning. Given gene expression levels of two genes and their correlation with cancer presence, teams must build models that can predict whether a patient sample indicates cancer or not.

Challenge:

- Perform exploratory data analysis (EDA) to understand the distribution and correlation of the two gene expression levels.
- Train at least two classification models (e.g., K-Nearest Neighbors, SVM, Neural Networks, Logistic Regression).
- Visualize the decision boundaries of models.
- Evaluate performance using metrics like accuracy, precision, recall, F1-score, ROC-AUC.
- Compare models and provide insights into which gene(s) play a stronger role in cancer classification.

Constraints:

- Must use at least two distinct ML algorithms.
- Models should be trained on 80% training data and tested on 20% unseen data.
- Maximum runtime for model training: 3 minutes per algorithm.
- Teams must visualize the scatter plot of gene expressions colored by cancer status.
- Neural networks (if used) should be lightweight (1–2 hidden layers).