Challenge: Participate in the Open - Make Your Team OSS-Contribution!

Motivation: Open Source is everywhere: More than 90 percent of all software projects and largest companies use Open Source software. Contributing to Open Source projects is a large advantage for software developers, students, researchers and any IT enthusiasts as it does not only significantly improve technical skills but also soft skills like collaboration, communication, marketing, conflict management resulting in much higher chances in the job market. However, the contribution barriers for larger Open Source projects have shifted significantly as well. Most core contributors are paid, long tenured software professionals and automated quality checks and contribution guidelines can look very intimidating for beginners. The goal of this years' InformatiCup is to encourage students to get over this initial contribution hurdle and unlock the many rewards of a potential career involving Open Source related development.

Task Description:

- *Project Selection:* Select a project that aligns with your interests from millions of Open Source projects. We would like to restrict the selection to existing Open Source projects with an active community. [1]
- *Task Identification:* Identify a specific algorithm or feature to work on. This includes adding a novel or improving an existing functionality of the project. [2]
- *Code Contribution:* Fork the repository, create a new branch, and implement your solution.
- *Pull Request*: Once the code is complete, create a pull request to the original repository. Pull requests must be started within the processing period to be considered. [3]
- *Code Review and Feedback*: Be open to feedback and suggestions from the project maintainers and other contributors.
- *Elaboration*: Finally, document your project in an accompanying document ranging between 8 and 12 pages. [4]

Additional Requirements:

- *Documentation*: The code should be well-documented, including clear docstrings explaining its purpose, parameters, and return value.
- *Demonstration*: Create a demo or tutorial showcasing the implemented feature. Provide clear instructions and examples for users to understand and utilize the implemented feature.
- *Community Engagement*: We encourage you to interact with the Open Source community, ask questions, as well as offer and accept help.
- *Testing*: Your project should rely on unit and integration tests to ensure the correctness of the implementation.

[1] Choose your project at GitHub, GitLab, Codeberg or a similar hosting platform. A good indicator of an active community is the number of stars, an actively managed issue log as well as regular merges and releases.
[2] Great inspiration for potential features and complex bug fixes would be repositories and issues labelled with hacktoberfest or good first issue.
[3] The pull request has to follow the code contribution guidelines of the forked repository and satisfy all automated checks that are part of the projects' test suites. Your pull request does not have to be merged for a valid submission.
[4] excluding references, single or double column allowed, main text font size 10-12pt

Coding Challenge Evaluation:

- *Code Quality*: Assess the quality of the code, including readability, efficiency, and adherence to coding standards.
- *Problem-Solving Skills*: Evaluate the participant's ability to understand and solve the problem. Consider the difficulty level as well as additional implications of the problem.
- *Collaboration and Community Engagement*: Assess the participant's willingness to collaborate with others and their ability to work in a team. Evaluate the participant's level of engagement with the Open Source community.
- *Documentation*: Evaluate the written project documentation. Consider a clear writing style, a good layout as well as a well researched background following academic standards. Assess the comprehensibility of the demo.

By focusing on Open Source contributions, we can not only test coding skills but also foster a sense of community and encourage collaborative practices in software development.