

Підтримка Таланту в ІТ: Олімпіади та Інновації

Ляпота Віталій
кафедра Програмної інженерії
ХНУРЕ
Харків, Україна
vitaliy.lyapota@nure.ua

Вечур Олександр
кафедра Програмної інженерії
ХНУРЕ
Харків, Україна
alexander.vechur@nure.ua

Саманцов Олександр
кафедра Програмної інженерії
ХНУРЕ
Харків, Україна
oleksandr.samantsov@nure.ua

Supporting Talent in IT: Olympiads and Innovations

Vitaliy Lyapota
Department of Software Engineering
Kharkiv National University of
Radio Electronics
Kharkiv, Ukraine
vitaliy.lyapota@nure.ua

Olekander Vechur
Department of Software Engineering
Kharkiv National University of
Radio Electronics
Kharkiv, Ukraine
alexander.vechur@nure.ua

Oleksandr Samantsov
Department of Software Engineering
Kharkiv National University of
Radio Electronics
Kharkiv, Ukraine
oleksandr.samantsov@nure.ua

Анотація— у роботі розглядається підтримка ІТ-освіти через Кубок України з програмування, який включає онлайн-змагання для студентів та професіоналів. Змагання сприяють розвитку навичок, командної роботи та конкурентного духу. Інноваційною є ідея освіти через змагання, через конкуренцію не обмежену рамками навчального потоку до університету.

Abstract— the work considers the support of IT education through the Ukrainian Programming Cup, which includes an online competition for students and professionals. Competitions promote skills, teamwork and competitive spirit. The idea of education through competition, through competition not limited by the framework of the educational stream to the university, is innovative.

Ключові слова — Кубок України; спортивне програмування; дивізіон; плагіат.

Keywords — Ukrainian Cup; sports programming; division; plagiarism.

I. ВСТУП

На даний час є вкрай важливим забезпечувати підтримку освітніх заходів не тільки для усіх студентів, але й зокрема для тих, хто виявляє високий потенціал. Ускладнені умови в країні вимагають від освітніх закладів не просто підтримувати існуючий рівень навчання та кількість студентів, а й приділяти особливу увагу талановитій молоді. Мова йде про необхідність глибокого вивчення алгоритмів і структур у межах ІТ-освіти, де вже було розроблено та впроваджено чіткі технологічні рішення. Є впевненість у тому, що наявний досвід та сучасні технології можуть бути застосовані та адаптовані в інших сферах.

Серед учнів існують ті, хто з ранніх років проявляє інтерес до олімпіад, і до моменту вступу до університетів уже мають певний рівень знань, який часто перевищує

базовий рівень, на якому проводиться вивчення фундаментальних алгоритмічних курсів в університетах.

Якщо у цих студентів зберігається бажання розвивати свою майстерність у алгоритмах, слід створити необхідні умови для цього. Часто таким студентам може бути недостатньо цікаво в рамках стандартних університетських курсів, тому потрібно використовувати інноваційні підходи, щоб організувати навчання, що відповідає їхнім потребам.

Не так багато студентів у середньому вищому навчальному закладі готові витратити час на вивчення алгоритмів; деякі університети мають більше таких студентів, деякі — менше, але ясно, що рамки традиційної освіти є обмеженнями для них.

Олімпіади мають велике значення, оскільки створюють сильне конкурентне середовище, служать стимулом для самостійної роботи, дозволяють швидко отримувати зворотний зв'язок та спостерігати особистий прогрес. Вони допомагають перевершити обмеження університетських програм і створюють умови для змагань на національному рівні, де колишні школярі, які потрапили до різних університетів, можуть конкурувати між собою.

Різноманітність олімпіад велика, але олімпіадам з програмування особливо пощастило, бо в них використовуються прогресивні ІТ-технології для підготовки та проведення змагань. Ці технології впроваджувались давно, тому вже накопичено достатньо досвіду для організації таких подій, і цей досвід можна перенести на інші освітні програми.

Спостереження за прогресом студентів доводить, що сильна конкуренція є важливою мотивацією для професійного зростання. Технології допомагають підтримувати конкуренцію за допомогою автоматизації систем оцінювання, мінімізації людського фактору та швидкого набуття балів. Важлива також команда

взаємодія, адже більшість університетських олімпіад із програмування проводяться у командному форматі.

II. КУБОК УКРАЇНИ З ПРОГРАМУВАННЯ

Для того, щоб сприяти просунутій алгоритмічній освіті, ось уже втретє проводяться паралельно з академічним навчальним роком етапи Кубка України з програмування (КУ) [1]. Це може розглядатися як додаткова альтернативна освіта для просунутих здобувачів освіти.

Крім того, КУ позиціонується як захід для трьох категорій учасників: школярі, студенти, професійні розробники, які поділяються при реєстрації на 2 дивізіони. Це дає можливість об'єднати три покоління хлопців та дати можливість комунікації у сфері передачі знань та досвіду. КУ складається з етапів, що являють собою онлайн олімпіади, які мають жорсткий дедлайн і вимагають якнайшвидше реалізувати і здати в систему набір завдань.

Для молодих хлопців дуже важливим є професійний досвід та знання від старших, раніше такий процес проходив насамперед через зміну покоління всередині команди, коли на заміну випускника до старшокурсників приходив першокурсник. Але додатковим показником професійного зростання є можливість вигадувати завдання, не у всіх це виходить, але ми вітаємо таку активність.

КУ складається з декількох етапів (не менш 10 етапів), кожен з яких є звичайним констестом на 5 годин. Кожний етап проводиться раз у два тижні у неділю. Між етапами заплановані дорожв'язування задач, які учасники не змогли розв'язати за 5 годин.

Загальний залік. Рейтинг підсумовується за усі етапи без урахування двох найгірших етапів. Рейтинг за етап вираховується згідно з формулою.

$$R_{team} = K_{div} * 100 * \frac{Solved_{team}}{Solved_{winner}} * \frac{2 * N_{team} - 2}{N_{team} + Place_{team} - 2}$$

де

K_{div} – 1 для команд першого дивізіону та 0,5 для команд другого дивізіону;

$Solved_{team}$ – кількість завдань, розв'язаних даною командою;

$Solved_{winner}$ – кількість завдань, розв'язаних переможцем;

N_{team} – кількість команд, які брали участь у даному етапі;

$Place_{team}$ – місце, що посіла дана команда.

Під час кубку між етапами проводиться консультації для роз'яснення нового матеріалу та допомоги у розв'язку складних задач. Кубок є настільки масштабним заходом, що можна його назвати альтернативною елітною освітою. Крім того елітність освітнього процесу під час Кубку забезпечуються за допомогою конкуренції між найсильнішими та найдібнішими школярами та студентами України.

III. ПЕРЕВІРКА ПРОГРАМНОГО КОДУ НА ПЛАГІАТ

Важливим етапом будь-яких змагань зі спортивного програмування є перевірка наданого командами коду на плагіат [2].

Основна мета перевірки на плагіат — забезпечити рівні умови для всіх учасників. Це допомагає гарантувати, що всі рішення, представлені учасниками, є їхньою власною роботою і не були скопійовані або взяті з інших джерел.

Для перевірки необхідно виконати декілька етапів.

1) Підготовка вхідних даних:

a) *Збір даних:* Агрегація всіх рішень учасників з системи, включаючи рішення з усіх раундів та ліг.

b) *Отримання авторських рішень:* Колекціонування рішень, поданих авторами задач, для подальшого порівняння з рішеннями учасників.

c) *Інтеграція зовнішніх джерел:* Додавання рішень, отриманих з інтернету, які відповідають задачам, що раніше використовувалися у змаганнях або навчальних цілях.

2) *Обробка вхідних даних, а саме розробка скрипта, який виконує наступні функції:*

a) *Екстракція метаданих:* Вилучення інформації з рішень учасників, включно з даними про команди та задачі. Структуризація даних з організацією в папки, що класифіковані за раундами та задачами.

b) *Форматування файлів:* Збереження інформації про команди в окремі файли, перевірка на допустимість імен файлів у файлової системі з автоматичною заміною спеціальних символів на символ "_".

c) *Групування рішень:* Аналогічне групування рішень з інтернету та авторських рішень за задачами, зі збереженням у структуровані папки для кожної задачі.

d) *Централізація даних:* Компіляція всіх рішень у спеціалізовані папки, щоб усі варіанти рішення для кожної задачі були доступні у єдиному місці для подальшого аналізу.

3) *Запуск алгоритму пошуку плагіату, який використовує наступні алгоритми:*

a) *Система неперетених множин.*

b) *Редакційна відстань.*

c) *Найдовша спільна підслідовність.*

Перевірка на плагіат дозволяє організаторам змагань виявити, чи справді учасники самостійно розв'язують задачі та розробляють рішення.

ЛІТЕРАТУРА REFERENCES

- [1] Кубок України з програмування - Регламент. Кубок України з програмування - Регламент. URL: <https://cup.org.ua/> (дата звернення: 10.05.2024).
- [2] Зайченко І.В. Аналіз методів та дієвих систем виявлення плагіату навчальних кодових програм. Інформатика, обчислювальна техніка та автоматизація. 2021. Т. 32(71), № 6. С. 85–90.