CODEOLYMP: ALGORITMIK MASALA YECHISHGA BAG'ISHLANGAN MIKROSERVIS ASOSIDAGI ONLAYN HUKM TIZIMI ARXITEKTURASI VA TATBIQI

Yuldoshev Begzod R
Kattaqo'rg'on davlat pedagogika instituti talabasi,
e-mail: yuldashevbehzod925@gmail.com

Annotatsiya. Maqolada talabalar va dasturchilar uchun mo'ljallangan CodeOlymp algoritmik masala yechish platformasi taqdim etilgan. Tizim Codeforces, LeetCode va HackerRank kabi xalqaro platformalarning afzalliklarini O'zbekiston ta'lim tizimining ehtiyojlariga moslab birlashtiradi. Platforma mikroservis arxitekturasi asosida qurilgan bo'lib, foydalanuvchilar tomonidan yuborilgan dastur kodini izolyatsiya qilingan Docker konteynerlarida xavfsiz tarzda bajaradi, natijani avtomatik ravishda baholaydi va Elo reyting tizimi asosida ishtirokchilarning ko'nikmasini hisoblab boradi. Tadqiqot natijasida platformaning ishlash samaradorligi, xavfsizlik darajasi va yuklamaga chidamliligi tahlil qilingan.

Kalit so'zlar: algoritmik dasturlash, onlayn hukm, sandbox, Docker, mikroservis, MongoDB, Node.js, ICPC, dasturlash musobaqalari, ta'lim texnologiyalari.

CODEOLYMP: АРХИТЕКТУРА И РЕАЛИЗАЦИЯ ОНЛАЙН-СИСТЕМЫ ПРИНЯТИЯ РЕШЕНИЙ НА ОСНОВЕ МИКРОСЕРВИСОВ, ПРЕДНАЗНАЧЕННОЙ ДЛЯ АЛГОРИТМИЧЕСКОГО РЕШЕНИЯ ЗАДАЧ

Аннотация. В статье представлена платформа CodeOlymp для решения алгоритмических задач, предназначенная для студентов и программистов. Система объединяет преимущества международных платформ — Codeforces, LeetCode и HackerRank — адаптируя их под потребности образовательной системы Узбекистана. Платформа построена на базе микросервисной архитектуры; код, отправленный пользователями, выполняется в изолированных Docker-контейнерах, автоматически оценивается, а навыки участников рассчитываются на основе системы рейтинга Elo. Проведённые тесты показали стабильность работы системы и удовлетворительную производительность для соревнований среднего масштаба.

Ключевые слова: алгоритмическое программирование, онлайн-судья, песочница, Docker, микросервисы, MongoDB, Node.js, ICPC, программистские соревнования, образовательные технологии.

CODEOLYMP: ARCHITECTURE AND IMPLEMENTATION OF A MICROSERVICE-BASED ONLINE DECISION SYSTEM DEDICATED TO ALGORITHMIC PROBLEM SOLVING

Abstract. The article presents CodeOlymp — an algorithmic problem-solving platform for students and developers. The system integrates the advantages of well-known international platforms (Codeforces, LeetCode, HackerRank) while adapting them to the needs of Uzbekistan's educational ecosystem. Built on a microservice architecture, the platform securely executes user-submitted source code in isolated Docker containers, performs automated grading, and computes participants' skills via an Elo-based rating system. The conducted tests demonstrated the stability of the system and sufficient performance for medium-scale contests.

Keywords: algorithmic programming, online judge, sandbox, Docker, microservices, MongoDB, Node.js, ICPC, programming contests, educational technologies.

Kirish

So'nggi yillarda axborot texnologiyalari sohasidagi mutaxassislarga bo'lgan talab keskin oshib bormoqda. Bu talabni qondirish uchun algoritmik tafakkurni rivojlantiruvchi onlayn platformalar muhim

rol o'ynaydi. Codeforces, LeetCode, HackerRank va AtCoder kabi xalqaro xizmatlar dunyo bo'ylab millionlab dasturchilarning o'sishiga hissa qo'shdi [6, 10].

Biroq mahalliy ta'lim muassasalarida o'qitiladigan algoritmik fanlarni qo'llab-quvvatlaydigan, o'zbek tilida ishlaydigan va mahalliy musobaqalarni tashkil etishga moslashtirilgan platforma yetishmasligi tadqiqot dolzarbligini belgilaydi. Mazkur maqolada ushbu muammoni hal qilish maqsadida ishlab chiqilgan CodeOlymp loyihasi taqdim etiladi.

Tadqiqot maqsadi — talabalarga algoritmik masalalarni mustaqil yechish, o'z bilimini avtomatik tarzda tekshirish va ICPC (International Collegiate Programming Contest) formatidagi musobaqalarda qatnashish imkonini beruvchi zamonaviy, xavfsiz va kengaytiriladigan onlayn hukm tizimini yaratish.

Mavjud yechimlar tahlili

Avtomatik baholash tizimlari 1990-yillarning oxiridan beri rivojlanib kelayotgan yo'nalishdir [2]. 1-jadvalda mavjud xalqaro platformalarning xususiyatlari taqqoslangan.

Platforma	Yili	Asosiy fokus	Til	Manba kodi
Codeforces	2009	Sport dasturlash	Rus / Ingliz	Yopiq
LeetCode	2015	Suhbat tayyorgarligi	Ingliz	Yopiq
HackerRank	2009	Ko'nikma sinovi	Ingliz	Yopiq
AtCoder	2012	Musobaqa	Yapon / Ingliz	Yopiq
DOMjudge	2004	ICPC mahalliy	Ingliz	Ochiq
CodeOlymp	**2026**	**O'zbekiston ta'limi**	**O'zbek**	**Ochiq**

1-jadval

Mavjud onlayn hukm platformalari taqqoslanishi

Quyidagi kamchiliklar mavjud platformalarda kuzatildi:

- til to'sig'i — interfeys va masala matnlari ko'pincha rus yoki ingliz tilida bo'ladi;
- yopiq manba kodi mahalliy ehtiyojlarga moslashtirish imkonini bermaydi;
- mahalliy musobaqalar va olimpiada formatlari bilan integratsiya yo'qligi;
- talabaning shaxsiy rivojlanish trayektoriyasini kuzatish imkoni cheklanganligi;
- tijoriy paketlarning yuqori narxi (HackerRank Enterprise, LeetCode Premium);
- eski xalqaro tizimlarning (PC^2, DOMjudge) zamonaviy UX talablariga javob bermasligi.

Tizim arxitekturasi

CodeOlymp mikroservis arxitekturasi asosida loyihalashtirilgan [3]. Bu yondashuv har bir komponentni mustaqil rivojlantirish, masshtablash va xavfsizlikni ta'minlashga imkon beradi.

1. Komponentlar tarkibi

Tizim quyidagi mustaqil servislardan iborat:

- Frontend — React 18 va Vite asosidagi yagona sahifali ilova (SPA);
- Backend — Node.js / Express 5 asosidagi REST API va Socket.IO real-time qatlami;
- Executor — yuborilgan kodlarni izolyatsiyalangan muhitda ishga tushiruvchi servis;
- MongoDB 7 — asosiy ma'lumotlar omborini boshqaruvchi NoSQL bazasi [7];
- Redis 7 — keshlash, Bull navbat va sessiya boshqaruvi uchun;
- MinIO — test holatlari (ZIP arxivlar) saqlanadigan S3-mos object storage;
- NGINX — teskari proksi va statik fayllarni xizmat ko'rsatish vositasi.

2. Texnologik stek

Loyihada ishlatilgan asosiy texnologiyalar:

- Node.js 20 LTS — server logikasi uchun yagona ishlash muhiti;
- Express.js 5 — HTTP API endpointlari uchun freymvork;
- Mongoose 9 — MongoDB bilan ishlash uchun ODM kutubxonasi;
- Socket.IO 4 — real-time bidirectional aloqa [8];
- Bull 4 — Redis asosidagi job navbati;
- React 18 va React Router 6 — komponent asosidagi frontend;
- Zustand va TanStack Query — frontend holatini boshqarish;
- Monaco Editor — VS Code kod muharririga asoslangan komponent;
- Tailwind CSS — foydalanuvchi interfeysi dizayni;
- Docker va Docker Compose — konteynerizatsiya va orkestratsiya [4].

3. Ma'lumotlar oqimi

Foydalanuvchi tomonidan yuborilgan dastur kodi quyidagi bosqichlardan o'tadi:

1. Frontend'dagi Monaco Editor orqali kod kiritiladi va POST so'rovi orqali Backend'ga yuboriladi.
2. Backend kodni MongoDB'da PENDING statusi bilan saqlaydi va Submission ID ni qaytaradi.
3. Bull queue orqali submission Executor servisiga uzatiladi.
4. Executor MongoDB dan masala metadatasini va MinIO dan test holatlarini yuklab oladi.
5. Tilga mos sandbox Docker konteyner ishga tushirilib, kod kompilyatsiya va bajariladi.
6. Har bir test holati uchun chiqish solishtirilib, verdict aniqlanadi.
7. Natija MongoDB ga yoziladi va Backend Socket.IO orqali foydalanuvchiga real-time xabar yuboradi.

Sandbox xavfsizligi

Algoritmik hukm tizimining eng kritik qismi — begona kodni xavfsiz bajarish mexanizmidir [9]. CodeOlymp da har bir submission alohida Docker konteynerda bir martalik ishga tushiriladi va quyidagi cheklovlar qo'llanadi:

- --network none — konteynerda tarmoq interfeysi mavjud emas (tashqi internetga chiqish imkonsiz);
- --memory=Xm — masala uchun belgilangan RAM cheklovi (sukut bo'yicha 256 MB);
- --memory-swap=Xm — swap ham bir xil chegara bilan cheklanadi;
- --cpus=1 — bitta CPU yadrosi;
- --pids-limit=50 — yangi jarayonlar soni 50 dan oshmaydi (fork-bombasi himoyasi);
- --read-only — fayl tizimi faqat o'qish rejimida;
- --tmpfs /tmp:size=64m,exec — vaqtinchalik papka cheklangan hajm bilan;
- timeout Xs — masala vaqti tugasa, jarayon Linux signal orqali to'xtatiladi;
- --rm — konteyner bajarish tugagach avtomatik o'chiriladi (state qoldirmaydi);
- non-root user — sandbox image'lar sandbox foydalanuvchisi nomidan ishlaydi.

1. Qo'llab-quvvatlanadigan dasturlash tillari

Hozirda quyidagi to'rt til qo'llab-quvvatlanadi:

- C++17 (gcc:13-bookworm) — g++ -O2 -std=c++17 flaglari bilan;
- Python 3.12 (python:3.12-slim) — interpretator orqali bevosita;
- Java 17 (eclipse-temurin:17-jdk) — javac va java jufti bilan;
- JavaScript (node:20-slim) — Node.js asosidagi bajarish.

2. Verdict turlari

Submission ning yakuniy holati to'qqiz holatdan biri bilan ifodalanadi:

- PENDING — masala navbatda kutmoqda;
- JUDGING — hozirda baholash jarayonida;
- AC (Accepted) — barcha testlar muvaffaqiyatli;
- WA (Wrong Answer) — chiqish kutilgan natijaga mos kelmadi;
- TLE (Time Limit Exceeded) — vaqt chegarasidan o'tib ketdi;
- MLE (Memory Limit Exceeded) — xotira chegarasidan oshdi;
- RE (Runtime Error) — bajarilish vaqtida xato yuz berdi;
- CE (Compile Error) — kompilyatsiya xatosi;
- SE (System Error) — tizim ichidagi nosozlik.

Reyting tizimi (Elo)

Musobaqalar uchun Elo reyting tizimining ICPC ga moslashtirilgan varianti qo'llanadi [5]. Har bir musobaqa yakuniga yetganda barcha juftliklar uchun kutilgan natija (1)-formula bo'yicha hisoblanadi:

$$E_A = 1 / (1 + 10^{((R_B - R_A) / 400)}) \qquad (1)$$

Bu yerda R_A va R_B — mos ravishda A va B ishtirokchining joriy reytingi. K-faktor 32 ga teng. Yangi foydalanuvchining boshlang'ich reytingi 1200 ga teng. Yangi reyting (2)-formula yordamida hisoblanadi:

$$R_A' = R_A + K \cdot \Sigma (S_Ai - E_Ai) \qquad (2)$$

bu yerda S_Ai — A ning i ishtirokchiga nisbatan haqiqiy natijasi (1 g'alaba, 0 yutqazish), E_Ai — kutilgan natija. Ko'rsatkichlar RatingHistory kolleksiyasida har bir musobaqadan keyin qayd etiladi.

Autentifikatsiya va xavfsizlik

Foydalanuvchilarning ma'lumotlari va kirish huquqlari quyidagi mexanizmlar orqali himoyalangan:

- bcrypt algoritmi (12 round) bilan parol heshlash;
- JWT access token (15 daqiqa) va refresh token (7 kun) juftligi;
- email tasdiqlash — ro'yxatdan o'tganda 6 raqamli kod yuboriladi;
- parolni tiklash — vaqt cheklovi (1 soat) bilan UUID asosidagi token;
- XSS himoyasi — barcha kirish ma'lumotlari xss kutubxonasi orqali tozalanadi;
- MongoDB injection himoyasi — $ va . belgilarni o'z ichiga olgan kalitlar o'chiriladi;
- rate limiting — /auth/login uchun daqiqada 10 marta, umumiy API uchun daqiqada 100 marta;
- Helmet.js orqali HTTP javob sarlavhalari kuchaytiriladi;
- CORS — faqat FRONTEND_URL dan kelgan so'rovlar qabul qilinadi;
- cookie-based refresh — httpOnly va secure bayroqlari bilan.

Real-time xabarnoma tizimi

Foydalanuvchilarga submission natijasini darhol yetkazish uchun Socket.IO 4 ishlatiladi [8]. Foydalanuvchi tizimga kirgach JWT token bilan WebSocket aloqasi tiklanadi, watch:submission xabari orqali muayyan submission ga obuna bo'ladi, watch:contest orqali esa musobaqa scoreboard iga obuna bo'ladi. Executor natija qaytarganda Backend submission:update event ini tegishli xonaga yuboradi. Bu mexanizm sahifani qayta yuklash zaruriyatini bartaraf qiladi va foydalanuvchi tajribasini sezilarli yaxshilaydi.

Tizimni yoyish

CodeOlymp Docker Compose yordamida aaPanel boshqaruv paneliga ega serverga yoyilgan. Quyidagi tarmoq topologiyasi qo'llanilgan:

- tashqi DNS so'rov codyolymp.katdpi.uz domeniga keladi;
- aaPanel NGINX servisi 80 va 443 portlarini qabul qiladi, SSL ni Let's Encrypt orqali boshqaradi;
- / joylashuv frontend container'ga (127.0.0.1:3000) yo'naltiriladi;
- /api/ joylashuv backend container'ga (127.0.0.1:5000) yo'naltiriladi;
- /socket.io/ joylashuv backend ga WebSocket upgrade bilan yo'naltiriladi;
- MongoDB, Redis va MinIO konteynerlari host portlariga chiqarilmaydi;
- Executor container Docker socket ni mount qilib, sandbox konteynerlarini ishga tushiradi.

Natijalar va muhokama

Loyiha sinov serverida muvaffaqiyatli yoyildi va quyidagi ko'rsatkichlarga erishildi:

- o'rtacha javob vaqti (API GET so'rovlari): 45–80 ms;
- submission baholash vaqti (oddiy C++ masalasi, 10 ta test): 1.2–2.5 sekund;
- bir vaqtning o'zida bajariladigan submission soni: 2 (sozlanadi);
- MongoDB indexlari: 8 ta optimallashtirilgan indeks;
- frontend bundle hajmi: 338 KB (gzip da 102 KB);
- backend xotira sarfi (idle): 75–95 MB;
- sandbox container ko'tarilish vaqti: 280–450 ms.

Kelajakdagi ishlar

Quyidagi yo'nalishlar bo'yicha tadqiqot va rivojlantirish ishlari rejalashtirilgan:

- qo'shimcha dasturlash tillarini joriy qilish (Rust, Go, C#, Kotlin);
- plagiat aniqlash uchun MOSS yoki shunga o'xshash algoritmlarni integratsiyalash;
- avtomatik masala tasdiqlash — test holatlari konsistentligini tekshirish;
- sun'iy intellekt asosidagi tavsiyalar tizimi;
- mobil ilova versiyasini ishlab chiqish (React Native);
- o'qituvchilar uchun statistika va guruh boshqaruv vositalari;
- custom checker (interaktiv masalalar) qo'llab-quvvatlash;
- Codeforces va AtCoder formatlaridagi masala importi.

Xulosa

Ushbu maqolada algoritmik dasturlash uchun mo'ljallangan CodeOlymp onlayn hukm tizimining arxitekturasi va tatbiqi taqdim etildi. Loyiha mikroservis yondashuvi, Docker asosidagi sandbox izolyatsiyasi va Elo reyting tizimini birlashtirib, talabalarga zamonaviy va xavfsiz muhit yaratdi. Yopiq manbali tijoriy raqobatchilardan farqli o'laroq, tizim ochiq manbali va mahalliy ta'lim ekotizimiga moslashtirilgan.

Olib borilgan sinovlar tizimning ishonchli ishlashini va o'rtacha o'lchamdagi musobaqalar uchun yetarli unumdorlik berishini ko'rsatdi. Olingan natijalar shuni ko'rsatadiki, ochiq manbali yondashuv mahalliy ehtiyojlarga muvofiq holda sifatli ta'lim platformalarini yaratish imkonini beradi. Kelajakda tizimni sun'iy intellekt asosidagi tavsiya beruvchi modullar bilan kengaytirish va ko'proq oliy ta'lim muassasalari bilan integratsiya qilish maqsad qilingan.

Foydalanilgan adabiyotlar

1. Mirzaxmedov A. Algoritmik tafakkurni rivojlantirish metodologiyasi. — Toshkent: Fan, 2022. — 184 b.
2. Forišek M., Steinová M. Explaining Algorithms Using Metaphors. — SpringerBriefs in Computer Science, 2013. — 105 p.
3. Newman S. Building Microservices: Designing Fine-Grained Systems. — 2nd ed. — O'Reilly Media, 2021. — 612 p.
4. Mouat A. Using Docker: Developing and Deploying Software with Containers. — O'Reilly Media, 2016. — 354 p.
5. Elo A. E. The Rating of Chessplayers, Past and Present. — Arco Pub., 1978. — 209 p.
6. Codeforces statistics. [Elektron resurs]. URL: https://codeforces.com (kirish sanasi: 19.05.2026).
7. MongoDB Manual. [Elektron resurs]. URL: https://www.mongodb.com/docs/manual (kirish sanasi: 15.05.2026).
8. Socket.IO Documentation. [Elektron resurs]. URL: https://socket.io/docs/v4 (kirish sanasi: 12.05.2026).
9. Codeforces statistics. [Elektron resurs]. URL: https://codeforces.com (kirish sanasi: 19.05.2026).
10. MongoDB Manual. [Elektron resurs]. URL: https://www.mongodb.com/docs/manual (kirish sanasi: 15.05.2026).
11. Socket.IO Documentation. [Elektron resurs]. URL: https://socket.io/docs/v4 (kirish sanasi: 12.05.2026).
12. Docker security documentation. [Elektron resurs]. URL: https://docs.docker.com/engine/security (kirish sanasi: 18.05.2026).
13. LeetCode Engineering Blog. How LeetCode Scales. [Elektron resurs]. URL: https://leetcode.com/discuss (kirish sanasi: 19.05.2026).